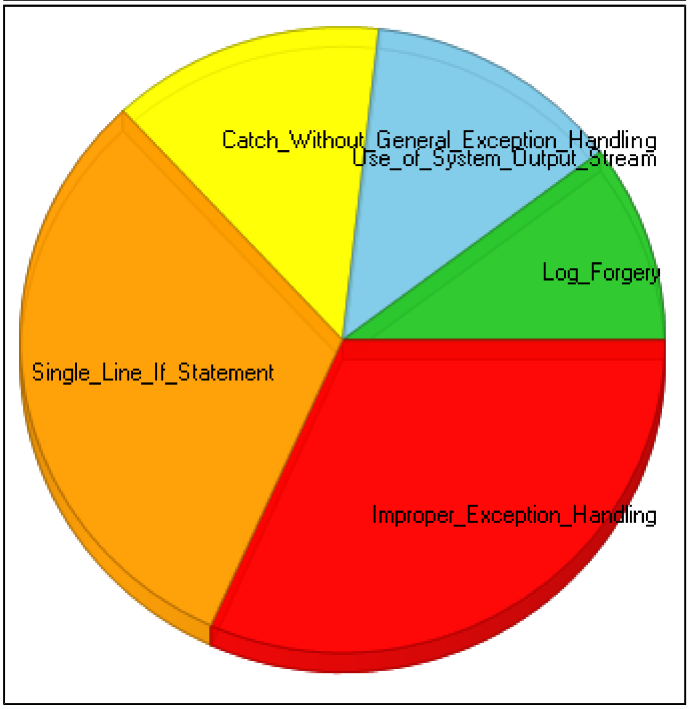
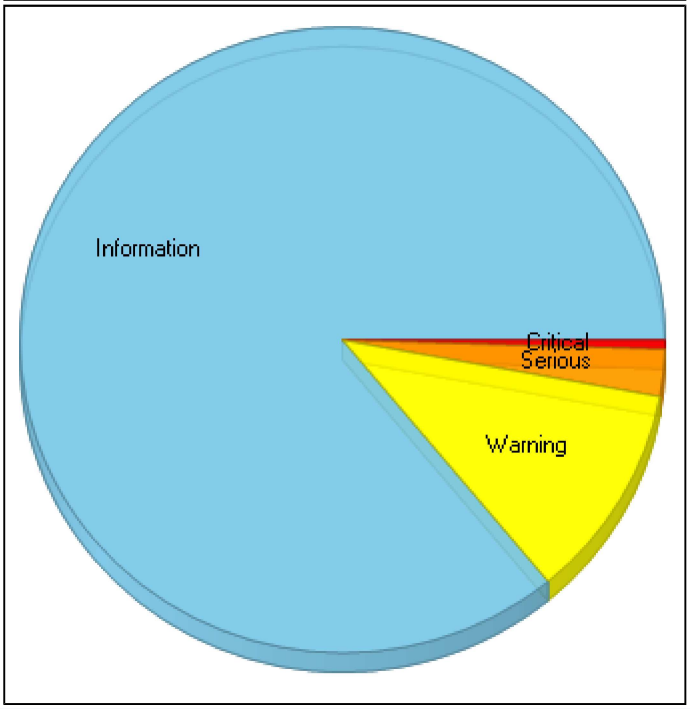
Checkmarx Report

Project Name:	**Dspace**
Load Time:	**27/02/2008 10:49:00**
CxSuite Version:	**2.4.3**

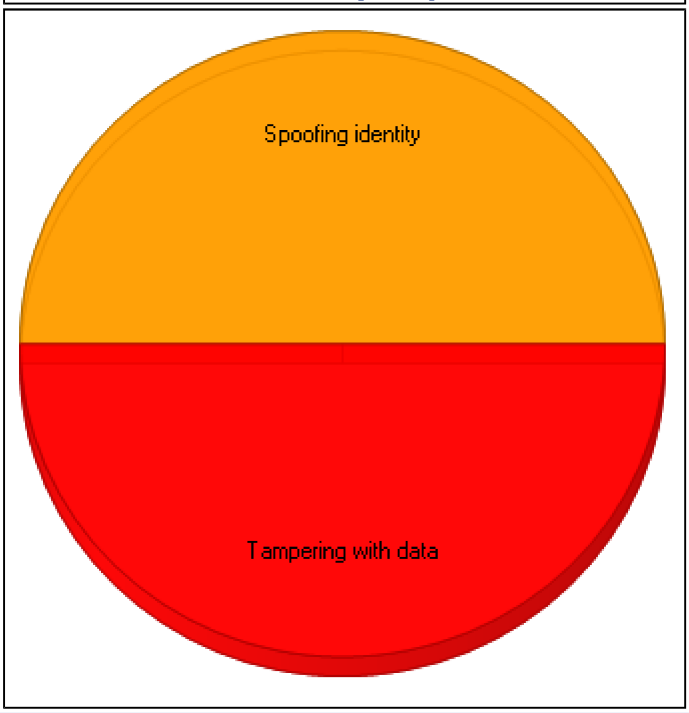
Problems by Queries



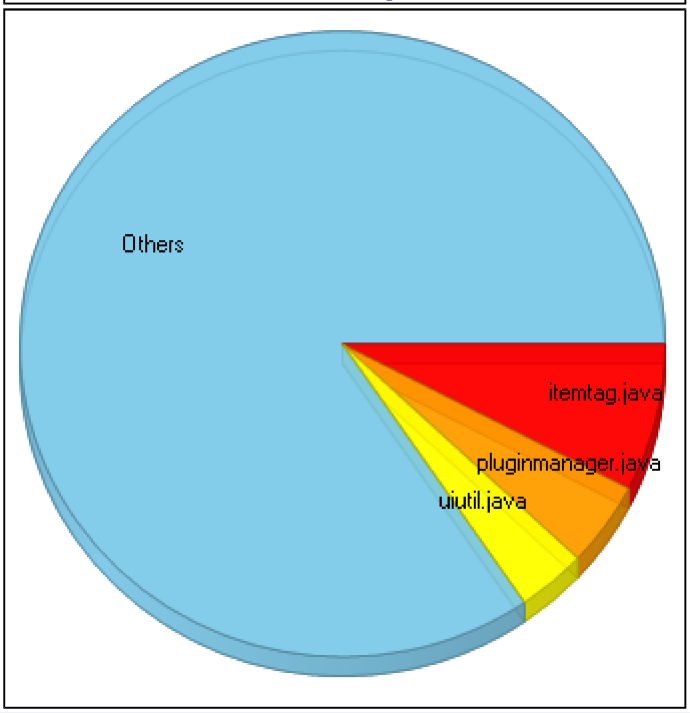
Problems by Severity



Problems by Impacts



Problems by Files



Summary

Query	Group	Problems found	Severity
Improper Exception Handling	Best Coding Practice	515	
Unclosed Objects	Best Coding Practice	24	
Catch Without General Exception Handling	Best Coding Practice	223	
Single Line If Statement	Best Coding Practice	510	
Empty Catch	Best Coding Practice	29	
No Default Case	Best Coding Practice	16	
Erroneous String Compare	Best Coding Practice	4	
Magic Numbers	Best Coding Practice	7	
Confusing Naming	Best Coding Practice	6	
Leftover Debug Code	Best Coding Practice	26	
Missing Catch Block	Best Coding Practice	4	
Missing XML Validation	Best Coding Practice	2	
Not Static Final Logger	Best Coding Practice	1	
Overly Broad Catch	Best Coding Practice	46	
Overly Broad Throws	Best Coding Practice	46	
Use of System Output Stream	Best Coding Practice	210	
Second Order SQL Injection	High Risk	1	
Reflected XSS All Clients	High Risk	2	
Stored XSS	High Risk	3	
LDAP Injection	High Risk	3	
Log Forgery	Low Visibility	167	
DoS by Unreleased Resources	Low Visibility	13	
URL Redirection Attack	Low Visibility	19	
Equals without GetHashCode	Low Visibility	5	
Improper Transaction Handling	Low Visibility	12	
Verbose Error Reporting	Medium Threat	34	
Files Manipulation	Medium Threat	4	
Parameter Tampering	Medium Threat	9	

Metrics

Metric Name	Value
Number of types	403
Number of Classes	393
Number of Interfaces	10
Avg methods per class	5
Number of Constructors	193
Avg statements per Method	9
Number of Control Statements	3580
Number of Methods	2082
Lines of Code	105078

Top 10 Files

File name	Problems found
dspace-1.4.2-source\src\org\dspace\app\webui\jsptag\itemtag.java	148
dspace-1.4.2-source\src\org\dspace\core\pluginmanager.java	85
dspace-1.4.2-source\src\org\dspace\app\webui\util\uiutil.java	67
dspace-1.4.2-source\src\org\dspace\checker\simplereporterimpl.java	58
dspace-1.4.2-source\src\org\dspace\content\packager\metsmanifest.java	54
dspace-1.4.2-source\src\org\dspace\storage\rdbms\databasemanager.java	45
dspace-1.4.2-source\src\org\dspace\core\configurationmanager.java	41
dspace-1.4.2-source\src\org\dspace\app\webui\servlet\registerservlet.java	37
dspace-1.4.2-source\jsp\submit\edit-metadata.jsp	35
dspace-1.4.2-source\src\org\dspace\content\crosswalk\qdccrosswalk.java	33

	Often Misused: Exception Handling
CWE ID	248
Description	Failing to catch an exception thrown from a dangerous function, can potentially cause the program to crash.
Demonstrative Examples	The _alloca() function allocates memory on the stack. If an allocation request is too large for the available stack space, _alloca() throws an exception. If the exception is not caught, the program will crash, potentially enabling a denial of service attack. _alloca() has been deprecated as of Microsoft Visual Studio 2005®. It has been replaced with the more secure _alloca_s().
	EnterCriticalSection() can raise an exception, potentially causing the program to crash. Under operating systems prior to Windows 2000, the EnterCriticalSection() function can raise an exception in low memory situations. If the exception is not caught, the program will crash, potentially enabling a denial of service attack.
Node Relationships	ChildOf - API Abuse (227) Peer - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	7 Pernicious Kingdoms - Often Misused: Exception Handling
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
89. out.println( "<li class=\"communityLink\">" );
```

[Table of Contents] [Description]

Path 2:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
```

```
...
90. out.println( "<strong><a href=\"" + request.getContextPath() + "/handle/"
+ c.getHandle() + "\">" + c.getMetadata("name") + "</a></strong>");
```

[Table of Contents] [Description]

Path 3:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
...
96. out.println("<ul>");
```

[Table of Contents] [Description]

Path 4:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
...
99. out.println("<li class=\"collectionListItem\">");
```

[Table of Contents] [Description]

Path 5:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
...
100. out.println("<a href=\"" + request.getContextPath() + "/handle/" + cols
[j].getHandle() + "\">" + cols[j].getMetadata("name") +"</a>");
```

[Table of Contents] [Description]

Path 6:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
...
103. out.println(" [" + cols[j].countItems() + "]");
```

[Table of Contents] [Description]

Path 7:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      |...
      106. out.println("</li>");
```

[Table of Contents] [Description]

Path 8:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      |...
      108. out.println("</ul>");
```

[Table of Contents] [Description]

Path 9:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      |...
      115. out.println("<ul>");
```

[Table of Contents] [Description]

Path 10:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      |...
      120. out.println("</ul>");
```

[Table of Contents] [Description]

Path 11:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      |...
      122. out.println("<br />");
```

[Table of Contents] [Description]

Path 12:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      ...
123. out.println("</li>");
```

[Table of Contents] [Description]

Path 13:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
247. private static boolean isFromMITCommunity(HttpServletRequest
request) //mitspecialgroup.java
      ...
251. log.debug("checking MIT membership of IP addr="+addr);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public static void loadBitstreamFormats(Context context, String
filename) //registryloader.java
      ...
172. log.info(LogManager.getHeader(context, "load_bitstream_formats",
```

[Table of Contents] [Description]

Path 15:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
226. public static void loadDublinCoreTypes(Context context, String
filename) //registryloader.java
      ...
244. log.info(LogManager.getHeader(context, "load_dublin_core_types",
```

[Table of Contents] [Description]

Path 16:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
371. private static void writeMetadata(Context c, String schema, Item i, File
```

```
destDir) //itemexport.java
    ...
395. out.write(utf8, 0, utf8.length);
```

[Table of Contents] [Description]

Path 17:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
371. private static void writeMetadata(Context c, String schema, Item i, File
destDir) //itemexport.java
    ...
399. out.write(utf8, 0, utf8.length);
```

[Table of Contents] [Description]

Path 18:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
371. private static void writeMetadata(Context c, String schema, Item i, File
destDir) //itemexport.java
    ...
415. out.write(utf8, 0, utf8.length);
```

[Table of Contents] [Description]

Path 19:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
371. private static void writeMetadata(Context c, String schema, Item i, File
destDir) //itemexport.java
    ...
419. out.write(utf8, 0, utf8.length);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
430. private static void writeHandle(Context c, Item i, File
destDir) //itemexport.java
    ...
441. out.println(i.getHandle());
```

[Table of Contents] [Description]

Path 21:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
465. private static void writeBitstreams(Context c, Item i, File
destDir) //itemexport.java
      |...
      |510. out.println("-r -s " + b.getStoreNumber()
```

[Table of Contents] [Description]

Path 22:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
465. private static void writeBitstreams(Context c, Item i, File
destDir) //itemexport.java
      |...
      |514. out.println(myName + "\tbundle:" + bundleName);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
512. private void replaceItems(Context c, Collection[]
mycollections, //itemimport.java
      |...
      |572. handleOut.println(oldHandle);
```

[Table of Contents] [Description]

Path 24:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
616. private Item addItem(Context c, Collection[] mycollections, String
path, //itemimport.java
      |...
      |687. mapOut.println(mapOutput);
```

[Table of Contents] [Description]

Path 25:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
118. public Vector getSchemaLocations(String identifier) //dspaceoaicatalog.java
      |...
```

```
122. log.info(LogManager.getHeader(null, "oai_request",
```

[Table of Contents] [Description]

Path 26:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
208. public Map listIdentifiers(String from, String until, String
set, //dspaceoaicatalog.java
     ...
214. .info(LogManager.getHeader(null, "oai_request",
```

[Table of Contents] [Description]

Path 27:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
328. public String getRecord(String identifier, String
metadataPrefix) //dspaceoaicatalog.java
     ...
333. .info(LogManager.getHeader(null, "oai_request",
```

[Table of Contents] [Description]

Path 28:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
426. public Map listRecords(String from, String until, String
set, //dspaceoaicatalog.java
     ...
432. .info(LogManager.getHeader(null, "oai_request",
```

[Table of Contents] [Description]

Path 29:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
426. public Map listRecords(String from, String until, String
set, //dspaceoaicatalog.java
     ...
448. log.info(LogManager.getHeader(null, "oai_error",
```

[Table of Contents] [Description]

Path 30:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
426. public Map listRecords(String from, String until, String
set, //dspaceoaicatalog.java
     |...
     458. log.info(LogManager.getHeader(null, "oai_error", "no_items_match"));
```

[Table of Contents] [Description]

Path 31:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
480. public Map listRecords(String resumptionToken) //dspaceoaicatalog.java
     |...
     483. log.info(LogManager.getHeader(null, "oai_request",
```

[Table of Contents] [Description]

Path 32:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
480. public Map listRecords(String resumptionToken) //dspaceoaicatalog.java
     |...
     514. log.info(LogManager.getHeader(null, "oai_error",
```

[Table of Contents] [Description]

Path 33:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
654. public Map listSets() throws NoSetHierarchyException, //dspaceoaicatalog.java
     |...
     657. log.info(LogManager.getHeader(null, "oai_request", "verb=listSets"));
```

[Table of Contents] [Description]

Path 34:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. private static String makeSchemaLocation(String
schemaLabel) //plugincrosswalk.java
     |...
     132. log.debug("Initialized schemaLabel="+schemaLabel+" with schemaLocation =
     \""+schemaLoc+"\"");
```

[Table of Contents] [Description]

Path 35:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. private static String makeSchemaLocation(String
schemaLabel) //plugincrosswalk.java
        ...
135. log.error("makeSchemaLocation: crosswalk cannot provide schemaLocation,
label="+schemaLabel);
```

[Table of Contents] [Description]

Path 36:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. private static String makeSchemaLocation(String
schemaLabel) //plugincrosswalk.java
        ...
138. log.error("No crosswalk found, makeSchemaLocation giving up,
label="+schemaLabel);
```

[Table of Contents] [Description]

Path 37:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
343. public static void processLogs(Context context, String
myLogDir, //loganalyser.java
        ...
409. while ((line = br.readLine()) != null)
```

[Table of Contents] [Description]

Path 38:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
898. public static void readConfig(String configFile) //loganalyser.java
        ...
918. while ((record = br.readLine()) != null)
```

[Table of Contents] [Description]

Path 39:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
547. public static void readMap(String map) //reportgenerator.java
```

```
...
567. while ((record = br.readLine()) != null)
```

[Table of Contents] [Description]

Path 40:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
624. public static void readInput(String input) //reportgenerator.java
...
646. while ((record = br.readLine()) != null)
```

[Table of Contents] [Description]

Path 41:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
297. private static Hashtable loadControlledVocabularies(String
directory) //controlledvocabularytag.java
...
332. log.warn("Could not find any vocabularies...");
```

[Table of Contents] [Description]

Path 42:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
...
120. out.println("<br/><p align=\"center\">");
```

[Table of Contents] [Description]

Path 43:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
...
121. out.println("<img src=\""
```

[Table of Contents] [Description]

Path 44:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
     ...
147. out.println("<br/>" + dcValue[0].value);
```

[Table of Contents] [Description]

Path 45:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
     ...
151. out.println("</p>");
```

[Table of Contents] [Description]

Path 46:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
290. out.println("<center><table class=\"itemDisplayTable\">");
```

[Table of Contents] [Description]

Path 47:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
339. out.print("<tr><td class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 48:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
341. out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 49:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
344. out.print(": </td><td class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 50:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
350. out.print("<br />");
```

[Table of Contents] [Description]

Path 51:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
355. out.print("<a href=\"" + values[j].value + "\">"
```

[Table of Contents] [Description]

Path 52:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
443. out.print(Utils.addEntities(values[i].value));
```

[Table of Contents] [Description]

Path 53:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
367. out.print(Utils.addEntities(values[j].value));
```

[Table of Contents] [Description]

Path 54:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
       ...
371. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 55:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
       ...
377. out.println("</table></center><br/>");
```

[Table of Contents] [Description]

Path 56:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
       ...
385. out.println("<br/><br/>");
```

[Table of Contents] [Description]

Path 57:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
       ...
400. out.println("<p align=\"center\">"
```

[Table of Contents] [Description]

Path 58:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
       ...
405. out.println("<center><table class=\"itemDisplayTable\">");
```

[Table of Contents] [Description]

Path 59:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
406. out.println("<tr><th id=\"s1\" class=\"standard\">"
```

[Table of Contents] [Description]

Path 60:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
432. .print("<tr><td headers=\"s1\" class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 61:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
433. out.print(values[i].schema);
```

[Table of Contents] [Description]

Path 62:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
434. out.print("." + values[i].element);
```

[Table of Contents] [Description]

Path 63:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
438. out.print("." + values[i].qualifier);
```

[Table of Contents] [Description]

Path 64:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
442. .print("</td><td headers=\"s2\" class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 65:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
443. out.print(Utils.addEntities(values[i].value));
```

[Table of Contents] [Description]

Path 66:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
445. .print("</td><td headers=\"s3\" class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 67:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
509. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 68:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
453. out.print(values[i].language);
```

[Table of Contents] [Description]

Path 69:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
456. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 70:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
462. out.println("</table></center><br/>");
```

[Table of Contents] [Description]

Path 71:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
     ...
469. out.println("<br/><br/>");
```

[Table of Contents] [Description]

Path 72:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
     ...
485. out.print("<tr><td class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 73:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
     ...
488. out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 74:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
493. out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 75:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
496. out.print("</td><td class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 76:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
500. out.print("<a href=\"");
```

[Table of Contents] [Description]

Path 77:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
501. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 78:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
502. out.print("/handle/");
```

[Table of Contents] [Description]

Path 79:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
       ...
503. out.print(collections[i].getHandle());
```

[Table of Contents] [Description]

Path 80:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
       ...
504. out.print("\">");
```

[Table of Contents] [Description]

Path 81:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
       ...
505. out.print(collections[i].getMetadata("name"));
```

[Table of Contents] [Description]

Path 82:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
       ...
506. out.print("</a><br/>");
```

[Table of Contents] [Description]

Path 83:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
       ...
509. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 84:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
       ...
522. out.print("<table align=\"center\" class=\"miscTable\"><tr>");
```

[Table of Contents] [Description]

Path 85:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
       ...
523. out.println("<td class=\"evenRowEvenCol\"><p><strong>"
```

[Table of Contents] [Description]

Path 86:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
       ...
757. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 87:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
       ...
785. out.println("<table align=\"center\" class=\"attentionTable\"><tr>");
```

[Table of Contents] [Description]

Path 88:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
       ...
787. out.println("<td class=\"attentionCell\"><p><strong>"
```

[Table of Contents] [Description]

Path 89:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
     |...
798. out.print("<div align=\"center\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 90:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
     |...
799. out.print("<strong><a target=\"_blank\" href=\"");
```

[Table of Contents] [Description]

Path 91:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
     |...
800. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 92:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
     |...
801. out.print("/retrieve/");
```

[Table of Contents] [Description]

Path 93:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
     |...
802. out.print(bitstreams[k].getID() + "/");
```

[Table of Contents] [Description]

Path 94:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
       ...
803. out.print(UIUtil.encodeBitstreamName(bitstreams[k].getName(),
```

[Table of Contents] [Description]

Path 95:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
       ...
806. .print("\">"
```

[Table of Contents] [Description]

Path 96:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
       ...
814. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 97:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
820. private void getStyleFor(Collection c) //itemtag.java
       ...
840. .warn("dspace.cfg specifies undefined item metadata display style '"
```

[Table of Contents] [Description]

Path 98:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
       ...
147. log.info(LogManager.getHeader(context, "view_bitstream",
```

[Table of Contents] [Description]

Path 99:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
      ...
176. log.info(LogManager.getHeader(context, "invalid_id", "path="
```

[Table of Contents] [Description]

Path 100:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
      ...
184. log.info(LogManager.getHeader(context, "view_bitstream",
```

[Table of Contents] [Description]

Path 101:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
387. log.info(LogManager.getHeader(context, "browse_" + what, logInfo
```

[Table of Contents] [Description]

Path 102:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
71. protected void doDSGet(Context context, HttpServletRequest
request, //communitylistservlet.java
      ...
75. log.info(LogManager.getHeader(context, "view_community_list", ""));
```

[Table of Contents] [Description]

Path 103:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. private void doSearch(Context context, HttpServletRequest
```

```
request, //controlledvocabularysearchservlet.java
    ...
321. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Path 104:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
67. protected void doDSGet(Context context, HttpServletRequest
request, //editprofileservlet.java
    ...
73. log.info(LogManager.getHeader(context, "view_profile", ""));
```

[Table of Contents] [Description]

Path 105:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
80. protected void doDSPost(Context context, HttpServletRequest
request, //editprofileservlet.java
    ...
121. log.info(LogManager.getHeader(context, "edit_profile",
```

[Table of Contents] [Description]

Path 106:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
80. protected void doDSPost(Context context, HttpServletRequest
request, //editprofileservlet.java
    ...
135. log.info(LogManager.getHeader(context, "view_profile",
```

[Table of Contents] [Description]

Path 107:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
    ...
128. log.info(LogManager.getHeader(context, "show_feedback_form",
```

[Table of Contents] [Description]

Path 108:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
    ...
173. log.info(LogManager.getHeader(context, "show_feedback_form",
```

[Table of Contents] [Description]

Path 109:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
198. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```

[Table of Contents] [Description]

Path 110:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
206. log.info(LogManager.getHeader(context, "invalid_syndformat", "path=" +
path));
```

[Table of Contents] [Description]

Path 111:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
550. private static void cache(String feedKey, CacheFeed newFeed) //feedservlet.java
    ...
590. log.info(logMsg);
```

[Table of Contents] [Description]

Path 112:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    |
```

```
...
141. log.info(LogManager
```

[Table of Contents] [Description]

Path 113:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
...
233. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 114:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
256. private void displayItem(Context context, HttpServletRequest
request, //handleservlet.java
...
273. .info(LogManager.getHeader(context, "view_item", "handle="
```

[Table of Contents] [Description]

Path 115:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
351. private void communityHome(Context context, HttpServletRequest
request, //handleservlet.java
...
359. log.info(LogManager.getHeader(context, "view_community",
```

[Table of Contents] [Description]

Path 116:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
426. private void collectionHome(Context context, HttpServletRequest
request, //handleservlet.java
...
462. log.info(LogManager.getHeader(context, "view_collection",
```

[Table of Contents] [Description]

Path 117:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
     ...
238. log.info(LogManager.getHeader(context, "view_html", "handle="
```

[Table of Contents] [Description]

Path 118:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
     ...
258. log.info(LogManager.getHeader(context, "view_html",
```

[Table of Contents] [Description]

Path 119:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
69. protected void doGet(HttpServletRequest request, //internalerrorservlet.java
     ...
83. log.warn(logMessage, t);
```

[Table of Contents] [Description]

Path 120:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbyauthorservlet.java
     ...
126. log.info(LogManager.getHeader(context, "items_by_author", logInfo
```

[Table of Contents] [Description]

Path 121:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
71. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbysubjectservlet.java
     ...
125. log.info(LogManager.getHeader(context, "items_by_subject", logInfo
```

[Table of Contents] [Description]

Path 122:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
140. log.info(LogManager
```

[Table of Contents] [Description]

Path 123:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
173. log.info(LogManager
```

[Table of Contents] [Description]

Path 124:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
194. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 125:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
202. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 126:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
225. log.info(LogManager.getHeader(context, "login",
```

[Table of Contents] [Description]

Path 127:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
233. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 128:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
244. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 129:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
67. protected void doDSGet(Context context, HttpServletRequest
request, //logoutservlet.java
       ...
71. log.info(LogManager.getHeader(context, "logout", ""));
```

[Table of Contents] [Description]

Path 130:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
102. protected void doDSPost(Context context, HttpServletRequest
request, //mydspaceservlet.java
       ...
137. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 131:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
```

```
...
211. log.info(LogManager.getHeader(context, "confirm_removal",
```

[Table of Contents] [Description]

Path 132:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
220. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 133:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
232. log.info(LogManager.getHeader(context, "view_workflow",
```

[Table of Contents] [Description]

Path 134:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
246. log.info(LogManager.getHeader(context, "view_workflow",
```

[Table of Contents] [Description]

Path 135:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
260. log.info(LogManager.getHeader(context, "unclaim_workflow",
```

[Table of Contents] [Description]

Path 136:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
274. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 137:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
311. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 138:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
322. log.info(LogManager.getHeader(context, "remove_submission",
```

[Table of Contents] [Description]

Path 139:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
    ...
368. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 140:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
    ...
425. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 141:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
       ...
465. log.info(LogManager.getHeader(context, "get_reject_reason",
```

[Table of Contents] [Description]

Path 142:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
       ...
476. log.info(LogManager.getHeader(context, "edit_workflow_item",
```

[Table of Contents] [Description]

Path 143:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
512. private void processRejectReason(Context context, //mydspaceservlet.java
       ...
534. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 144:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
580. private void showMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
       ...
584. log.info(LogManager.getHeader(context, "view_mydspace", ""));
```

[Table of Contents] [Description]

Path 145:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
636. private void showPreviousSubmissions(Context context, //mydspaceservlet.java
       ...
659. log.info(LogManager.getHeader(context, "view_own_submissions", "count="
```

[Table of Contents] [Description]

Path 146:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
97. log.info(LogManager.getHeader(context, "openURL", logInfo
```

[Table of Contents] [Description]

Path 147:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
    ...
101. log.info(LogManager.getHeader(context, "login", "type=explicit"));
```

[Table of Contents] [Description]

Path 148:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
    ...
114. log.info(LogManager.getHeader(context, "failed_login",
```

[Table of Contents] [Description]

Path 149:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
193. protected void doDSPost(Context context, HttpServletRequest
request, //registerservlet.java
    ...
224. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 150:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
        ...
470.    log.info(LogManager.getHeader(context, "invalid_token", "token="
```

[Table of Contents] [Description]

Path 151:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
        ...
527.    log.info(LogManager.getHeader(context, "usedtoken_register",
```

[Table of Contents] [Description]

Path 152:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
571. private void processNewPassword(Context context, //registerservlet.java
        ...
585.    log.info(LogManager.getHeader(context, "invalid_token", "token="
```

[Table of Contents] [Description]

Path 153:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
571. private void processNewPassword(Context context, //registerservlet.java
        ...
606.    log.info(LogManager.getHeader(context, "usedtoken_forgotpw",
```

[Table of Contents] [Description]

Path 154:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
        ...
113.    log.info(LogManager.getHeader(context, "view_bitstream",
```

[Table of Contents] [Description]

Path 155:

Query Name - **Improper_Exception_Handling**

Severity

- **Information**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
    ...
133. log.info(LogManager.getHeader(context, "view_bitstream",
```

[Table of Contents] [Description]

Path 156:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
    ...
292. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Path 157:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
    ...
250. while ((line = br.readLine()) != null)
```

[Table of Contents] [Description]

Path 158:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
    ...
312. log.info(LogManager.getHeader(context, "select_collection", ""));
```

[Table of Contents] [Description]

Path 159:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
320. protected void doDSPost(Context context, HttpServletRequest
request, //submitservlet.java
    ...
439. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 160:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
787. private void processEditMetadata(Context context, //submitservlet.java
       ...
933. log.info(" inner "+schema);
```

[Table of Contents] [Description]

Path 161:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
       ...
1098. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 162:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1178. private void processGetFileFormat(Context context, //submitservlet.java
       ...
1217. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 163:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
       ...
1314. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 164:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
       ...
```

```
1345. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 165:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1389. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 166:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1448. private void processChangeFileDescription(Context
context, //submitservlet.java
    ...
1472. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 167:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1598. private void processLicense(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1608. log.info(LogManager.getHeader(context, "accept_license",
```

[Table of Contents] [Description]

Path 168:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1598. private void processLicense(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1630. log.info(LogManager.getHeader(context, "reject_license",
```

[Table of Contents] [Description]

Path 169:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
1783. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 170:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
      ...
1883. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 171:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
      ...
105. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```

[Table of Contents] [Description]

Path 172:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
      ...
136. log.info(LogManager.getHeader(context, "show_suggest_form",
```

[Table of Contents] [Description]

Path 173:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
      ...
220. log.info(LogManager.getHeader(context, "show_suggest_form",
"problem=false"));
```

[Table of Contents] [Description]

Path 174:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
108. private void showMainPage(Context c, //viewworkspaceitemservlet.java
       ...
124. log.info(LogManager.getHeader(c,
```

[Table of Contents] [Description]

Path 175:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. private void showErrorPage(Context context, //viewworkspaceitemservlet.java
       ...
161. log.error(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 176:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
113. private void showErrorPage(Context context, //workspaceservlet.java
       ...
119. log.error(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 177:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. private void showMainPage(Context context, //workspaceservlet.java
       ...
163. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 178:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
       ...
```

```
167. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 179:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
200. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 180:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
229. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 181:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
461. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 182:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
588. private void showNextPage(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
602. log.debug(LogManager.getHeader(context, "nextpage", "stage=" + stage));
```

[Table of Contents] [Description]

Path 183:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      288. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 184:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
171. protected void doDSPost(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      275. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 185:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
229. private void showSchemas(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
      ...
      235. log.info("Showing Schemas");
```

[Table of Contents] [Description]

Path 186:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
232. void addSupervisionOrder(Context context, //superviseservlet.java
      ...
      244. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 187:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
277. void removeSupervisionOrder(Context context, //superviseservlet.java
      ...
      288. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 188:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
177. public static boolean startAuthentication(Context context, //authenticate.java
       ...
194. log.info(LogManager.getHeader(context, "login", "type=implicit"));
```

[Table of Contents] [Description]

Path 189:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
       ...
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 190:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
88. public static Context obtainContext(HttpServletRequest request) //uiutil.java
       ...
130. log.warn("POSSIBLE HIJACKED SESSION: request from "+
```

[Table of Contents] [Description]

Path 191:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
88. public static Context obtainContext(HttpServletRequest request) //uiutil.java
       ...
142. log.debug("Adding Special Group id="+String.valueOf(groupIDs[i]));
```

[Table of Contents] [Description]

Path 192:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
587. public static String encodeBitstreamName(String stringIn, String
encoding) //uiutil.java
       ...
648. log.debug("encoded \"" + stringIn + "\" to \"" + out.toString() + "\"");
```

[Table of Contents] [Description]

Path 193:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
316. public static List getLastSubmitted(BrowseScope scope) throws
SQLException //browse.java
     |...
     323. log.debug("SQL for last submitted is \"" + sql + "\"");
```

[Table of Contents] [Description]

Path 194:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
     |...
     722. log.debug("Number of matches " + matches);
```

[Table of Contents] [Description]

Path 195:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1066. private static List getResults(BrowseScope scope, List results, int
max) //browse.java
     |...
     1107. log.debug("Adding result " + theValue);
```

[Table of Contents] [Description]

Path 196:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1132. private static PreparedStatement createSql(BrowseScope scope, //browse.java
     |...
     1160. log.debug("Created SQL \"" + sql + "\"");
```

[Table of Contents] [Description]

Path 197:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1175. private static String createSqlInternal(BrowseScope scope, //browse.java
     |...
```

```
1231. log.debug("Subquery is \"" + subquery + "\"");
```

[Table of Contents] [Description]

Path 198:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1386. private static void logInfo(BrowseInfo info) //browse.java
      ...
1393. log.debug("Number of Results: " + info.getResultCount()
```

[Table of Contents] [Description]

Path 199:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1386. private static void logInfo(BrowseInfo info) //browse.java
      ...
1403. log.debug("Got no results");
```

[Table of Contents] [Description]

Path 200:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1386. private static void logInfo(BrowseInfo info) //browse.java
      ...
1406. log.debug("Got results: " + info.getOverallPosition() + " to "
```

[Table of Contents] [Description]

Path 201:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1753. public static BrowseInfo get(BrowseScope key) //browse.java
      ...
1758. .debug("Checking browse cache with " + cache.size()
```

[Table of Contents] [Description]

Path 202:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1753. public static BrowseInfo get(BrowseScope key) //browse.java
```

```
...
1814. log.debug("Using cached browse");
```

[Table of Contents] [Description]

Path 203:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1937. public static TableRow countAndMax(Context context, BrowseScope
scope) //browse.java
...
1958. log.debug("Running SQL to check whether index has changed: \""
```

[Table of Contents] [Description]

Path 204:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
125. public void process() //checkercommand.java
...
127. LOG.debug("Begin Checker Processing");
```

[Table of Contents] [Description]

Path 205:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
125. public void process() //checkercommand.java
...
147. LOG.debug("Processing bitstream id = " + id);
```

[Table of Contents] [Description]

Path 206:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
221. private String digestStream(InputStream stream, String
algorithm) //checkercommand.java
...
231. while (dStream.read(bytes, 0, BYTE_ARRAY_SIZE) != -1)
```

[Table of Contents] [Description]

Path 207:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
91. public ResultsLogger(Date startDt) //resultslogger.java
      ...
93. this.resultDAO = new ChecksumResultDAO();
```

[Table of Contents] [Description]

Path 208:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
91. public ResultsLogger(Date startDt) //resultslogger.java
      ...
95. LOG.info(msg("run-start-time") + ": " + DATE_FORMAT.format(startDt));
```

[Table of Contents] [Description]

Path 209:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
119. LOG.info("*******************************************************");
```

[Table of Contents] [Description]

Path 210:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
120. LOG.info(msg("bitstream-id") + ": " + info.getBitstreamId());
```

[Table of Contents] [Description]

Path 211:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
121. LOG.info(msg("bitstream-info-found") + ": " + info.getInfoFound());
```

[Table of Contents] [Description]

Path 212:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
122. LOG.info(msg("bitstream-marked-deleted") + ": " + info.getDeleted());
```

[Table of Contents] [Description]

Path 213:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
123. LOG.info(msg("bitstream-found") + ": " + info.getBitstreamFound());
```

[Table of Contents] [Description]

Path 214:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
124. LOG.info(msg("to-be-processed") + ": " + info.getToBeProcessed());
```

[Table of Contents] [Description]

Path 215:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
125. LOG.info(msg("internal-id") + ": " + info.getInternalId());
```

[Table of Contents] [Description]

Path 216:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
126. LOG.info(msg("name") + ": " + info.getName());
```

[Table of Contents] [Description]

Path 217:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
127. LOG.info(msg("store-number") + ": " + info.getStoreNumber());
```

[Table of Contents] [Description]

Path 218:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
128. LOG.info(msg("size") + ": " + info.getSize());
```

[Table of Contents] [Description]

Path 219:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
129. LOG.info(msg("bitstream-format") + ": " + info.getBitstreamFormatId());
```

[Table of Contents] [Description]

Path 220:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
130. LOG.info(msg("user-format-description") + ": "
```

[Table of Contents] [Description]

Path 221:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
      ...
132. LOG.info(msg("source") + ": " + info.getSource());
```

[Table of Contents] [Description]

Path 222:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
     ...
134. .info(msg("checksum-algorithm") + ": "
```

[Table of Contents] [Description]

Path 223:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
     ...
136. LOG.info(msg("previous-checksum") + ": " + info.getStoredChecksum());
```

[Table of Contents] [Description]

Path 224:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
     ...
137. LOG.info(msg("previous-checksum-date")
```

[Table of Contents] [Description]

Path 225:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
     ...
141. LOG.info(msg("new-checksum") + ": " + info.getCalculatedChecksum());
```

[Table of Contents] [Description]

Path 226:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
     ...
142. LOG.info(msg("checksum-comparison-result") + ": "
```

[Table of Contents] [Description]

Path 227:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
117. public void collect(BitstreamInfo info) //resultslogger.java
       ...
144. LOG.info("\n\n");
```

[Table of Contents] [Description]

Path 228:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
       ...
111. osw.write("\n");
```

[Table of Contents] [Description]

Path 229:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
       ...
112. osw.write(msg("deleted-bitstream-intro"));
```

[Table of Contents] [Description]

Path 230:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
       ...
113. osw.write(DATE_FORMAT_MIN.format(startDate));
```

[Table of Contents] [Description]

Path 231:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
       ...
114. osw.write(" ");
```

[Table of Contents] [Description]

Path 232:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |115. osw.write(msg("date-range-to"));
```

[Table of Contents] [Description]

Path 233:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |116. osw.write(" ");
```

[Table of Contents] [Description]

Path 234:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |117. osw.write(DATE_FORMAT_MIN.format(endDate));
```

[Table of Contents] [Description]

Path 235:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |118. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 236:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |
```

```
...
122. osw.write("\n\n");
```

[Table of Contents] [Description]

Path 237:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
123. osw.write(msg("no-bitstreams-to-delete"));
```

[Table of Contents] [Description]

Path 238:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. public int getDeletedBitstreamReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
124. osw.write("\n");
```

[Table of Contents] [Description]

Path 239:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
156. osw.write("\n");
```

[Table of Contents] [Description]

Path 240:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
157. osw.write(msg("checksum-did-not-match"));
```

[Table of Contents] [Description]

Path 241:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
158. osw.write(" ");
```

[Table of Contents] [Description]

Path 242:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
159. osw.write("\n");
```

[Table of Contents] [Description]

Path 243:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
160. osw.write(DATE_FORMAT_MIN.format(startDate));
```

[Table of Contents] [Description]

Path 244:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
161. osw.write(" ");
```

[Table of Contents] [Description]

Path 245:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
```

```
162. osw.write(msg("date-range-to"));
```

[Table of Contents] [Description]

Path 246:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
163. osw.write(" ");
```

[Table of Contents] [Description]

Path 247:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
164. osw.write(DATE_FORMAT_MIN.format(endDate));
```

[Table of Contents] [Description]

Path 248:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
165. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 249:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
...
169. osw.write("\n\n");
```

[Table of Contents] [Description]

Path 250:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
170. osw.write(msg("no-changed-bitstreams"));
```

[Table of Contents] [Description]

Path 251:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public int getChangedChecksumReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
171. osw.write("\n");
```

[Table of Contents] [Description]

Path 252:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
203. osw.write("\n");
```

[Table of Contents] [Description]

Path 253:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
204. osw.write(msg("bitstream-not-found-report"));
```

[Table of Contents] [Description]

Path 254:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
205. osw.write(DATE_FORMAT_MIN.format(startDate));
```

[Table of Contents] [Description]

Path 255:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
206. osw.write(" ");
```

[Table of Contents] [Description]

Path 256:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
207. osw.write(msg("date-range-to"));
```

[Table of Contents] [Description]

Path 257:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
208. osw.write(" ");
```

[Table of Contents] [Description]

Path 258:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
      ...
209. osw.write(DATE_FORMAT_MIN.format(endDate));
```

[Table of Contents] [Description]

Path 259:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
     ...
210. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 260:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
     ...
214. osw.write("\n\n");
```

[Table of Contents] [Description]

Path 261:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
     ...
215. osw.write(msg("no-bitstreams-changed"));
```

[Table of Contents] [Description]

Path 262:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
196. public int getBitstreamNotFoundReport(Date startDate, Date
endDate, //simplereporterimpl.java
     ...
216. osw.write("\n");
```

[Table of Contents] [Description]

Path 263:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
     ...
249. osw.write("\n");
```

[Table of Contents] [Description]

Path 264:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |250. osw.write(msg("bitstream-will-no-longer-be-processed"));
```

[Table of Contents] [Description]

Path 265:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |251. osw.write(" ");
```

[Table of Contents] [Description]

Path 266:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |252. osw.write(DATE_FORMAT_MIN.format(startDate));
```

[Table of Contents] [Description]

Path 267:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
     |...
     |253. osw.write(" ");
```

[Table of Contents] [Description]

Path 268:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
```

```
endDate, //simplereporterimpl.java
    ...
254. osw.write(msg("date-range-to"));
```

[Table of Contents] [Description]

Path 269:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
    ...
255. osw.write(" ");
```

[Table of Contents] [Description]

Path 270:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
    ...
256. osw.write(DATE_FORMAT_MIN.format(endDate));
```

[Table of Contents] [Description]

Path 271:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
    ...
257. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 272:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
    ...
261. osw.write("\n\n");
```

[Table of Contents] [Description]

Path 273:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
      |...
      |262. osw.write(msg("no-bitstreams-to-no-longer-be-processed"));
```

[Table of Contents] [Description]

Path 274:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
242. public int getNotToBeProcessedReport(Date startDate, Date
endDate, //simplereporterimpl.java
      |...
      |263. osw.write("\n");
```

[Table of Contents] [Description]

Path 275:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
      |...
      |290. osw.write("\n");
```

[Table of Contents] [Description]

Path 276:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
      |...
      |291. osw.write(msg("unchecked-bitstream-report"));
```

[Table of Contents] [Description]

Path 277:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
      |
```

```
...
292. osw.write(DATE_FORMAT_MIN.format(new Date()));
```

[Table of Contents] [Description]

Path 278:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
...
293. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 279:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
...
297. osw.write("\n\n");
```

[Table of Contents] [Description]

Path 280:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
...
298. osw.write(msg("no-unchecked-bitstreams"));
```

[Table of Contents] [Description]

Path 281:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
...
299. osw.write("\n");
```

[Table of Contents] [Description]

Path 282:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
      ...
      303. osw.write(msg("howto-add-unchecked-bitstreams"));
```

[Table of Contents] [Description]

Path 283:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
284. public int getUncheckedBitstreamsReport(OutputStreamWriter
osw) //simplereporterimpl.java
      ...
      304. osw.write("\n\n\n");
```

[Table of Contents] [Description]

Path 284:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
322. private void printHistoryRecords(List history, OutputStreamWriter
osw) //simplereporterimpl.java
      ...
      346. osw.write(buf.toString());
```

[Table of Contents] [Description]

Path 285:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
361. private void printDSpaceInfoRecords(List bitstreams, OutputStreamWriter
osw) //simplereporterimpl.java
      ...
      394. osw.write(buf.toString());
```

[Table of Contents] [Description]

Path 286:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
131. public static Bitstream find(Context context, int id) throws
SQLException //bitstream.java
      ...
```

```
148. log.debug(LogManager.getHeader(context, "find_bitstream",
```

[Table of Contents] [Description]

Path 287:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
131. public static Bitstream find(Context context, int id) throws
SQLException //bitstream.java
     ...
158. log.debug(LogManager.getHeader(context, "find_bitstream",
```

[Table of Contents] [Description]

Path 288:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. static Bitstream create(Context context, InputStream is) //bitstream.java
     ...
186. log.info(LogManager.getHeader(context, "create_bitstream",
```

[Table of Contents] [Description]

Path 289:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
210. static Bitstream register(Context context, //bitstream.java
     ...
218. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 290:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
463. public void update() throws SQLException, AuthorizeException //bitstream.java
     ...
468. log.info(LogManager.getHeader(bContext, "update_bitstream",
```

[Table of Contents] [Description]

Path 291:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
479. void delete() throws SQLException //bitstream.java
```

```
...
490. log.info(LogManager.getHeader(bContext, "delete_bitstream",
```

[Table of Contents] [Description]

Path 292:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
139. public static BitstreamFormat find(Context context, int
id) //bitstreamformat.java
...
158. log.debug(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 293:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
139. public static BitstreamFormat find(Context context, int
id) //bitstreamformat.java
...
169. log.debug(LogManager.getHeader(context, "find_bitstream_format",
```

[Table of Contents] [Description]

Path 294:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
233. private static BitstreamFormat findByFinish(Context
context, //bitstreamformat.java
...
240. log.debug(LogManager.getHeader(context, "find_bitstream",
```

[Table of Contents] [Description]

Path 295:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
389. public static BitstreamFormat create(Context context) throws
SQLException, //bitstreamformat.java
...
403. log.info(LogManager.getHeader(context, "create_bitstream_format",
```

[Table of Contents] [Description]

Path 296:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
561. public void update() throws SQLException,
AuthorizeException //bitstreamformat.java
      ...
570. log.info(LogManager.getHeader(bfContext, "update_bitstream_format",
```

[Table of Contents] [Description]

Path 297:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
598. public void delete() throws SQLException,
AuthorizeException //bitstreamformat.java
      ...
630. log.info(LogManager.getHeader(bfContext, "delete_bitstream_format",
```

[Table of Contents] [Description]

Path 298:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public static Bundle find(Context context, int id) throws
SQLException //bundle.java
      ...
158. log.debug(LogManager.getHeader(context, "find_bundle",
```

[Table of Contents] [Description]

Path 299:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public static Bundle find(Context context, int id) throws
SQLException //bundle.java
      ...
168. log.debug(LogManager.getHeader(context, "find_bundle",
```

[Table of Contents] [Description]

Path 300:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
187. static Bundle create(Context context) throws SQLException //bundle.java
      ...
192. log.info(LogManager.getHeader(context, "create_bundle", "bundle_id="
```

[Table of Contents] [Description]

Path 301:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
400. public void addBitstream(Bitstream b) throws SQLException, //bundle.java
      |
      ...
      406. log.info(LogManager.getHeader(ourContext, "add_bitstream", "bundle_id="
```

[Table of Contents] [Description]

Path 302:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
448. public void removeBitstream(Bitstream b) throws
AuthorizeException, //bundle.java
      |
      ...
      454. log.info(LogManager.getHeader(ourContext, "remove_bitstream",
```

[Table of Contents] [Description]

Path 303:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
501. public void update() throws SQLException, AuthorizeException //bundle.java
      |
      ...
      505. log.info(LogManager.getHeader(ourContext, "update_bundle", "bundle_id="
```

[Table of Contents] [Description]

Path 304:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
516. void delete() throws SQLException, AuthorizeException,
IOException //bundle.java
      |
      ...
      518. log.info(LogManager.getHeader(ourContext, "delete_bundle", "bundle_id="
```

[Table of Contents] [Description]

Path 305:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
178. public static Collection find(Context context, int id) throws
SQLException //collection.java
```

```
...
195. log.debug(LogManager.getHeader(context, "find_collection",
```

[Table of Contents] [Description]

Path 306:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
178. public static Collection find(Context context, int id) throws
SQLException //collection.java
...
205. log.debug(LogManager.getHeader(context, "find_collection",
```

[Table of Contents] [Description]

Path 307:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
223. static Collection create(Context context) throws
SQLException, //collection.java
...
256. log.info(LogManager.getHeader(context, "create_collection",
```

[Table of Contents] [Description]

Path 308:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. public Bitstream setLogo(InputStream is) throws
AuthorizeException, //collection.java
...
452. log.info(LogManager.getHeader(ourContext, "remove_logo",
```

[Table of Contents] [Description]

Path 309:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. public Bitstream setLogo(InputStream is) throws
AuthorizeException, //collection.java
...
467. log.info(LogManager.getHeader(ourContext, "set_logo",
```

[Table of Contents] [Description]

Path 310:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
719. public void createTemplateItem() throws SQLException,
AuthorizeException //collection.java
      |...
      |729. log.info(LogManager.getHeader(ourContext, "create_template_item",
```

[Table of Contents] [Description]

Path 311:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
746. public void removeTemplateItem() throws SQLException,
AuthorizeException, //collection.java
      |...
      |757. log.info(LogManager.getHeader(ourContext, "remove_template_item",
```

[Table of Contents] [Description]

Path 312:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
777. public void addItem(Item item) throws SQLException,
AuthorizeException //collection.java
      |...
      |782. log.info(LogManager.getHeader(ourContext, "add_item", "collection_id="
```

[Table of Contents] [Description]

Path 313:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
803. public void removeItem(Item item) throws SQLException,
AuthorizeException, //collection.java
      |...
      |809. log.info(LogManager.getHeader(ourContext, "remove_item",
```

[Table of Contents] [Description]

Path 314:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
850. public void update() throws SQLException, IOException,
AuthorizeException //collection.java
      |...
```

```
858. log.info(LogManager.getHeader(ourContext, "update_collection",
```

[Table of Contents] [Description]

Path 315:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
914. void delete() throws SQLException, AuthorizeException,
IOException //collection.java
...
916. log.info(LogManager.getHeader(ourContext, "delete_collection",
```

[Table of Contents] [Description]

Path 316:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1187. public int countItems() //collection.java
...
1198. ResultSet rs = statement.executeQuery();
```

[Table of Contents] [Description]

Path 317:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1187. public int countItems() //collection.java
...
1201. int itemcount = rs.getInt(1);
```

[Table of Contents] [Description]

Path 318:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
133. public static Community find(Context context, int id) throws
SQLException //community.java
...
150. log.debug(LogManager.getHeader(context, "find_community",
```

[Table of Contents] [Description]

Path 319:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
133. public static Community find(Context context, int id) throws
SQLException //community.java
     ...
160. log.debug(LogManager.getHeader(context, "find_community",
```

[Table of Contents] [Description]

Path 320:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
176. public static Community create(Community parent, Context
context) //community.java
     ...
204. log.info(LogManager.getHeader(context, "create_community",
```

[Table of Contents] [Description]

Path 321:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
382. public Bitstream setLogo(InputStream is) throws
AuthorizeException, //community.java
     ...
397. log.info(LogManager.getHeader(ourContext, "remove_logo",
```

[Table of Contents] [Description]

Path 322:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
382. public Bitstream setLogo(InputStream is) throws
AuthorizeException, //community.java
     ...
416. log.info(LogManager.getHeader(ourContext, "set_logo",
```

[Table of Contents] [Description]

Path 323:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
427. public void update() throws SQLException, IOException,
AuthorizeException //community.java
     ...
435. log.info(LogManager.getHeader(ourContext, "update_community",
```

[Table of Contents] [Description]

Path 324:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
628. public void addCollection(Collection c) throws SQLException, //community.java
     |...
     |634. log.info(LogManager.getHeader(ourContext, "add_collection",
```

[Table of Contents] [Description]

Path 325:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
681. public void addSubcommunity(Community c) throws SQLException, //community.java
     |...
     |687. log.info(LogManager.getHeader(ourContext, "add_subcommunity",
```

[Table of Contents] [Description]

Path 326:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
717. public void removeCollection(Collection c) throws
SQLException, //community.java
     |...
     |723. log.info(LogManager.getHeader(ourContext, "remove_collection",
```

[Table of Contents] [Description]

Path 327:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
763. public void removeSubcommunity(Community c) throws
SQLException, //community.java
     |...
     |769. log.info(LogManager.getHeader(ourContext, "remove_subcommunity",
```

[Table of Contents] [Description]

Path 328:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
807. public void delete() throws SQLException, AuthorizeException,
IOException //community.java
     |...
```

```
834. log.info(LogManager.getHeader(ourContext, "delete_community",
```

[Table of Contents] [Description]

Path 329:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
130. Item(Context context, TableRow row) throws SQLException //item.java
     ...
153. log.error("Loading item - cannot found metadata field "
```

[Table of Contents] [Description]

Path 330:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
195. public static Item find(Context context, int id) throws
SQLException //item.java
     ...
211. log.debug(LogManager.getHeader(context, "find_item",
```

[Table of Contents] [Description]

Path 331:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
195. public static Item find(Context context, int id) throws
SQLException //item.java
     ...
221. log.debug(LogManager.getHeader(context, "find_item", "item_id="
```

[Table of Contents] [Description]

Path 332:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
239. static Item create(Context context) throws SQLException,
AuthorizeException //item.java
     ...
253. log.info(LogManager.getHeader(context, "create_item", "item_id="
```

[Table of Contents] [Description]

Path 333:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1089. public void addBundle(Bundle b) throws SQLException,
AuthorizeException //item.java
      |...
      1094. log.info(LogManager.getHeader(ourContext, "add_bundle", "item_id="
```

[Table of Contents] [Description]

Path 334:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1132. public void removeBundle(Bundle b) throws SQLException,
AuthorizeException, //item.java
      |...
      1138. log.info(LogManager.getHeader(ourContext, "remove_bundle", "item_id="
```

[Table of Contents] [Description]

Path 335:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1349. public void update() throws SQLException, AuthorizeException //item.java
      |...
      1361. log.info(LogManager.getHeader(ourContext, "update_item", "item_id="
```

[Table of Contents] [Description]

Path 336:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1349. public void update() throws SQLException, AuthorizeException //item.java
      |...
      1450. log.warn(LogManager
```

[Table of Contents] [Description]

Path 337:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1511. public void withdraw() throws SQLException, AuthorizeException,
IOException //item.java
      |...
      1572. log.info(LogManager.getHeader(ourContext, "withdraw_item", "user="
```

[Table of Contents] [Description]

Path 338:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1583. public void reinstate() throws SQLException, AuthorizeException, //item.java
      ...
1640. log.info(LogManager.getHeader(ourContext, "reinstate_item", "user="
```

[Table of Contents] [Description]

Path 339:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1653. void delete() throws SQLException, AuthorizeException, IOException //item.java
      ...
1658. log.info(LogManager.getHeader(ourContext, "delete_item", "item_id="
```

[Table of Contents] [Description]

Path 340:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
258. public void create(Context context) throws IOException,
AuthorizeException, //metadatafield.java
      ...
287. log.info(LogManager.getHeader(context, "create_metadata_field",
```

[Table of Contents] [Description]

Path 341:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
411. public void update(Context context) throws SQLException, //metadatafield.java
      ...
446. log.info(LogManager.getHeader(context, "update_metadatafieldregistry",
```

[Table of Contents] [Description]

Path 342:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
478. public void delete(Context context) throws SQLException,
AuthorizeException //metadatafield.java
      ...
487. log.info(LogManager.getHeader(context, "delete_metadata_field",
```

[Table of Contents] [Description]

Path 343:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
508. private boolean unique(Context context, int schemaID, String
element, //metadatafield.java
     |...
     |526. ResultSet rs = statement.executeQuery();
```

[Table of Contents] [Description]

Path 344:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
508. private boolean unique(Context context, int schemaID, String
element, //metadatafield.java
     |...
     |531. count = rs.getInt(1);
```

[Table of Contents] [Description]

Path 345:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
588. private static void initCache(Context context) throws
SQLException //metadatafield.java
     |...
     |593. log.info("Loading MetadataField elements into cache.");
```

[Table of Contents] [Description]

Path 346:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
205. public void create(Context context) throws SQLException, //metadataschema.java
     |...
     |243. .info(LogManager.getHeader(context, "create_metadata_schema",
```

[Table of Contents] [Description]

Path 347:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
291. public void update(Context context) throws SQLException, //metadataschema.java
     |...
     |321. log.info(LogManager.getHeader(context, "update_metadata_schema",
```

[Table of Contents] [Description]

Path 348:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
333. public void delete(Context context) throws SQLException,
AuthorizeException //metadataschema.java
      |...
      342. log.info(LogManager.getHeader(context, "delete_metadata_schema",
```

[Table of Contents] [Description]

Path 349:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
385. private boolean uniqueNamespace(Context context, String
namespace) //metadataschema.java
      |...
      398. ResultSet rs = statement.executeQuery();
```

[Table of Contents] [Description]

Path 350:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
385. private boolean uniqueNamespace(Context context, String
namespace) //metadataschema.java
      |...
      403. count = rs.getInt(1);
```

[Table of Contents] [Description]

Path 351:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
417. private boolean uniqueShortName(Context context, String
name) //metadataschema.java
      |...
      431. ResultSet rs = statement.executeQuery();
```

[Table of Contents] [Description]

Path 352:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
417. private boolean uniqueShortName(Context context, String
name) //metadataschema.java
      |...
      |436. count = rs.getInt(1);
```

[Table of Contents] [Description]

Path 353:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
499. private static void initCache(Context context) throws
SQLException //metadataschema.java
      |...
      |504. log.info("Loading schema cache for fast finds");
```

[Table of Contents] [Description]

Path 354:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
338. public void update(Context context) throws SQLException,
AuthorizeException //metadatavalue.java
      |...
      |347. log.info(LogManager.getHeader(context, "update_metadatavalue",
```

[Table of Contents] [Description]

Path 355:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
358. public void delete(Context context) throws SQLException,
AuthorizeException //metadatavalue.java
      |...
      |360. log.info(LogManager.getHeader(context, "delete_metadata_value",
```

[Table of Contents] [Description]

Path 356:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
115. public static WorkspaceItem find(Context context, int id) //workspaceitem.java
      |...
      |133. log.debug(LogManager.getHeader(context, "find_workspace_item",
```

[Table of Contents] [Description]

Path 357:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
115. public static WorkspaceItem find(Context context, int id) //workspaceitem.java
       ...
143. log.debug(LogManager.getHeader(context, "find_workspace_item",
```

[Table of Contents] [Description]

Path 358:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
165. public static WorkspaceItem create(Context c, Collection
coll, //workspaceitem.java
       ...
280. log.info(LogManager.getHeader(c, "create_workspace_item",
```

[Table of Contents] [Description]

Path 359:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
463. public void update() throws SQLException, AuthorizeException,
IOException //workspaceitem.java
       ...
469. log.info(LogManager.getHeader(ourContext, "update_workspace_item",
```

[Table of Contents] [Description]

Path 360:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
484. public void deleteAll() throws SQLException,
AuthorizeException, //workspaceitem.java
       ...
505. log.info(LogManager.getHeader(ourContext, "delete_workspace_item",
```

[Table of Contents] [Description]

Path 361:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
533. public void deleteWrapper() throws SQLException,
AuthorizeException, //workspaceitem.java
       ...
```

```
542. log.info(LogManager.getHeader(ourContext, "delete_workspace_item",
```

[Table of Contents] [Description]

Path 362:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
     ...
248. log.error("Must use PluginManager to instantiate
MODSDisseminationCrosswalk so the class knows its name.");
```

[Table of Contents] [Description]

Path 363:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
     ...
257. log.error(msg);
```

[Table of Contents] [Description]

Path 364:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
     ...
284. log.warn("Illegal MODS mapping in "+propsFile.toString()+", line = "+
```

[Table of Contents] [Description]

Path 365:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
347. log.warn("WARNING: "+getPluginInstanceName()+": No MODS mapping for \"" +
qdc+"\"");
```

[Table of Contents] [Description]

Path 366:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
87. public void ingest(Context context, DSpaceObject dso, Element
root) //nullingestioncrosswalk.java
      ...
91. log.debug("Null crosswalk is ignoring this metadata Element: \n"+
```

[Table of Contents] [Description]

Path 367:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
95. public void ingest(Context context, DSpaceObject dso, List
ml) //nullingestioncrosswalk.java
      ...
99. log.debug("Null crosswalk is ignoring this List of metadata: \n"+
```

[Table of Contents] [Description]

Path 368:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      ...
170. log.debug("Bitstream checksum agrees with PREMIS: "+bitstream.getName());
```

[Table of Contents] [Description]

Path 369:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      ...
175. log.warn("Cannot test checksum on bitstream="+bitstream.getName()+
```

[Table of Contents] [Description]

Path 370:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      ...
194. log.debug("Changing bitstream id="+String.valueOf(bitstream.getID())
+"name and source to: "+bsName);
```

[Table of Contents] [Description]

Path 371:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      ...
208. log.debug("Skipping element: "+me.toString());
```

[Table of Contents] [Description]

Path 372:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
      ...
381. log.warn("WARNING: "+myName+": No QDC mapping for \"" + qdc+"\"");
```

[Table of Contents] [Description]

Path 373:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      ...
466. log.warn("WARNING: "+myName+": No mapping for Element=\"" + key+"\" to
qdc.");
```

[Table of Contents] [Description]

Path 374:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
160. protected static String[] makeAliases(String direction) //xsltcrosswalk.java
      ...
168. log.debug("XSLTCrosswalk: Looking for config prefix = "+prefix);
```

[Table of Contents] [Description]

Path 375:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
160. protected static String[] makeAliases(String direction) //xsltcrosswalk.java
```

```
...
174. log.debug("Getting XSLT plugin name from config line: "+key);
```

[Table of Contents] [Description]

Path 376:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
191. protected XSLTransformer getTransformer(String direction) //xsltcrosswalk.java
...
198. log.error("Must use PluginManager to instantiate XSLTCrosswalk so the
class knows its name.");
```

[Table of Contents] [Description]

Path 377:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
191. protected XSLTransformer getTransformer(String direction) //xsltcrosswalk.java
...
205. log.error("Missing configuration filename for XSLT-based crosswalk: no "+
```

[Table of Contents] [Description]

Path 378:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
...
143. log.error("Must use PluginManager to instantiate
XSLTDisseminationCrosswalk so the class knows its name.");
```

[Table of Contents] [Description]

Path 379:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
...
154. log.warn("No schemaLocation for crosswalk="+myAlias+",
key="+prefix+"schemaLocation");
```

[Table of Contents] [Description]

Path 380:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
       ...
158. log.warn("Possible INVALID schemaLocation (no space found) for
crosswalk="+
```

[Table of Contents] [Description]

Path 381:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
108. private void applyDim(List dimList, Item item) //xsltingestioncrosswalk.java
       ...
124. log.error("Got unexpected element in DIM list: "+elt.toString());
```

[Table of Contents] [Description]

Path 382:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
       ...
615. if (licenseID != null)
```

[Table of Contents] [Description]

Path 383:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
       ...
257. log.warn("Adding zero-length file for Bitstream, SID="+String.valueOf
(bitstreams[k].getSequenceID())+", not authorized for READ.");
```

[Table of Contents] [Description]

Path 384:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
       ...
263. log.warn("Skipping Bitstream, SID="+String.valueOf(bitstreams
[k].getSequenceID())+", not authorized for READ.");
```

[Table of Contents] [Description]

Path 385:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
168. public void addLicense(Context context, Collection
collection, //dspacemetsingester.java
      |...
      |184. log.debug("Got Creative Commons license in rightsMD");
```

[Table of Contents] [Description]

Path 386:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      |...
      |221. log.warn("Schema file not found for config entry=\""+spec+"\"");
```

[Table of Contents] [Description]

Path 387:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      |...
      |238. log.warn("Schema config entry has wrong format, entry=\""+spec+"\"");
```

[Table of Contents] [Description]

Path 388:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      |...
      |242. log.debug("Got local schemas = \""+localSchemas+"\"");
```

[Table of Contents] [Description]

Path 389:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
682. private Element getFirstDiv() //metsmanifest.java
      |...
```

```
693. log.debug("Got firstDiv result="+result.toString());
```

[Table of Contents] [Description]

Path 390:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
764. public Element[] getItemRightsMD() //metsmanifest.java
     ...
772. log.debug("getItemRightsMD: No ADMID references found.");
```

[Table of Contents] [Description]

Path 391:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
811. public void crosswalkBitstream(Context context, Bitstream
bitstream, //metsmanifest.java
     ...
825. log.warn("Got no bitstream ADMID, file@ID="+fileId);
```

[Table of Contents] [Description]

Path 392:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
811. public void crosswalkBitstream(Context context, Bitstream
bitstream, //metsmanifest.java
     ...
841. log.debug("Got bitstream techMD of type="+type+", for file ID="+fileId);
```

[Table of Contents] [Description]

Path 393:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
     ...
614. log.warn("Interpolation failed in value of property \""+key+
```

[Table of Contents] [Description]

Path 394:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
673. private static void installConfigurationFile(String
template) //configurationmanager.java
     ...
706. while ((line = in.readLine()) != null)
```

[Table of Contents] [Description]

Path 395:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
673. private static void installConfigurationFile(String
template) //configurationmanager.java
     ...
760. out.println(line);
```

[Table of Contents] [Description]

Path 396:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
820. private static void info(String string) //configurationmanager.java
     ...
828. log.info(string);
```

[Table of Contents] [Description]

Path 397:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
832. private static void warn(String string, Exception
e) //configurationmanager.java
     ...
841. log.warn(string, e);
```

[Table of Contents] [Description]

Path 398:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
845. private static void warn(String string) //configurationmanager.java
     ...
853. log.warn(string);
```

[Table of Contents] [Description]

Path 399:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. private static void error(String string) //configurationmanager.java
       ...
865. log.error(string);
```

[Table of Contents] [Description]

Path 400:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
869. private static void fatal(String string, Exception
e) //configurationmanager.java
       ...
878. log.fatal(string, e);
```

[Table of Contents] [Description]

Path 401:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
882. private static void fatal(String string) //configurationmanager.java
       ...
890. log.fatal(string);
```

[Table of Contents] [Description]

Path 402:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. public static Object[] getPluginSequence(Class intfc) //pluginmanager.java
       ...
192. log.warn("No Configuration entry found for Sequence Plugin
interface="+iname);
```

[Table of Contents] [Description]

Path 403:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
180. public static Object[] getPluginSequence(Class intfc) //pluginmanager.java
       ...
204. log.debug("Adding Sequence plugin for interface= "+iname+",
class="+classname[i]);
```

[Table of Contents] [Description]

Path 404:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
277. log.debug("Got Named configuration for interface="+iname+",
config="+namedVal);
```

[Table of Contents] [Description]

Path 405:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
328. log.error("No named plugins found for interface="+iname);
```

[Table of Contents] [Description]

Path 406:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
333. private static int installNamedConfigs(String iname, String classname, String
names[]) //pluginmanager.java
     ...
341. log.error("Name collision in named plugin, implementation
class=\""+classname+
```

[Table of Contents] [Description]

Path 407:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
333. private static int installNamedConfigs(String iname, String classname, String
names[]) //pluginmanager.java
     ...
345. log.debug("Got Named Plugin, intfc="+iname+", name="+names[i]+",
class="+classname);
```

[Table of Contents] [Description]

Path 408:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
```

```
...
587. while ((line = cr.readLine()) != null)
```

[Table of Contents] [Description]

Path 409:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
...
601. log.error("Duplicate key \""+key+"\" in DSpace configuration
file="+config.toString());
```

[Table of Contents] [Description]

Path 410:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
...
616. log.error("Key with unknown prefix \""+key+"\" in DSpace configuration
file="+config.toString());
```

[Table of Contents] [Description]

Path 411:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
...
633. log.error("Key is in ConfigurationManager.propertyNames() but NOT text
crawl: \""+nk+"\"");
```

[Table of Contents] [Description]

Path 412:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
...
641. log.error("Key is in text crawl but NOT
ConfigurationManager.propertyNames(): \""+key+"\"");
```

[Table of Contents] [Description]

Path 413:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
670. log.error("Single plugin config not found for: "+SINGLE_PREFIX+key);
```

[Table of Contents] [Description]

Path 414:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
686. log.error("Sequence plugin config not found for: "+SEQUENCE_PREFIX+key);
```

[Table of Contents] [Description]

Path 415:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
705. log.error("Selfnamed plugin config not found for:
"+SELFNAMED_PREFIX+key);
```

[Table of Contents] [Description]

Path 416:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
729. log.error("Named plugin config not found for: "+NAMED_PREFIX+key);
```

[Table of Contents] [Description]

Path 417:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
751. log.error("In plugin.reusable configuration, class \""+rk+"\" is NOT a
plugin implementation class.");
```

[Table of Contents] [Description]

Path 418:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
218. public static void copy(final InputStream input, final OutputStream
output) //utils.java
      |...
      |226. final int count = input.read(buffer, 0, BUFFER_SIZE);
```

[Table of Contents] [Description]

Path 419:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
218. public static void copy(final InputStream input, final OutputStream
output) //utils.java
      |...
      |234. output.write(buffer, 0, count);
```

[Table of Contents] [Description]

Path 420:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
221. protected static TableRow sendInfo(Context context, String
email, //accountmanager.java
      |...
      |245. log.debug("Created callback "
```

[Table of Contents] [Description]

Path 421:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
278. private static void sendEmail(String email, boolean isRegister, TableRow
rd) //accountmanager.java
      |...
      |299. log.info("Sent " + (isRegister ? "registration" : "account")
```

[Table of Contents] [Description]

Path 422:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
405. public static EPerson create(Context context) throws
SQLException, //eperson.java
      |...
```

```
420. log.info(LogManager.getHeader(context, "create_eperson", "eperson_id="
```

[Table of Contents] [Description]

Path 423:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
433. public void delete() throws SQLException, AuthorizeException, //eperson.java
     ...
473. log.info(LogManager.getHeader(myContext, "delete_eperson",
```

[Table of Contents] [Description]

Path 424:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
738. public void update() throws SQLException, AuthorizeException //eperson.java
     ...
751. log.info(LogManager.getHeader(myContext, "update_eperson",
```

[Table of Contents] [Description]

Path 425:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
198. public static Group create(Context context) throws SQLException, //group.java
     ...
213. log.info(LogManager.getHeader(context, "create_group", "group_id="
```

[Table of Contents] [Description]

Path 426:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
868. public void delete() throws SQLException //group.java
     ...
900. log.info(LogManager.getHeader(myContext, "delete_group", "group_id="
```

[Table of Contents] [Description]

Path 427:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
962. public void update() throws SQLException, AuthorizeException //group.java
```

```
...
1020. log.info(LogManager.getHeader(myContext, "update_group", "group_id="
```

[Table of Contents] [Description]

Path 428:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
...
149. log.info(LogManager.getHeader(context, "auth", "attempting trivial auth
of user="+netid));
```

[Table of Contents] [Description]

Path 429:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
...
174. log.info(LogManager
```

[Table of Contents] [Description]

Path 430:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
...
190. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 431:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
...
178. log.info(LogManager.getHeader(context, "authenticate", "attempting
password auth of user="+username));
```

[Table of Contents] [Description]

Path 432:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
     ...
199. log.warn(LogManager.getHeader(context, "authenticate", "rejecting
PasswordAuthentication because "+username+" requires certificate."));
```

[Table of Contents] [Description]

Path 433:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
     ...
207. log.info(LogManager.getHeader(context, "authenticate",
"type=PasswordAuthentication"));
```

[Table of Contents] [Description]

Path 434:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
91. public static void subscribe(Context context, EPerson eperson, //subscribe.java
     ...
113. log.info(LogManager.getHeader(context, "subscribe",
```

[Table of Contents] [Description]

Path 435:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
139. public static void unsubscribe(Context context, EPerson
eperson, //subscribe.java
     ...
161. log.info(LogManager.getHeader(context, "unsubscribe",
```

[Table of Contents] [Description]

Path 436:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
322. public static void sendEmail(Context context, EPerson eperson, //subscribe.java
     ...
428. log.info(LogManager.getHeader(context, "sent_subscription",
```

[Table of Contents] [Description]

Path 437:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
95. public static String resolveToURL(Context context, String
handle) //handlemanager.java
      |...
      |110. log.debug("Resolved " + handle + " to " + url);
```

[Table of Contents] [Description]

Path 438:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
156. public static String createHandle(Context context, DSpaceObject
dso) //handlemanager.java
      |...
      |169. log.debug("Created new handle for "
```

[Table of Contents] [Description]

Path 439:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
188. public static String createHandle(Context context, DSpaceObject
dso, //handlemanager.java
      |...
      |201. log.debug("Created new handle for "
```

[Table of Contents] [Description]

Path 440:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      |...
      |247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 441:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      |...
```

```
259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 442:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
...
271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 443:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public void init(StreamTable st) throws Exception //handleplugin.java
...
102. log.info("Called init (not implemented)");
```

[Table of Contents] [Description]

Path 444:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
109. public void setHaveNA(byte[] theHandle, boolean haveit) //handleplugin.java
...
115. log.info("Called setHaveNA (not implemented)");
```

[Table of Contents] [Description]

Path 445:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
122. public void createHandle(byte[] theHandle, HandleValue[]
values) //handleplugin.java
...
128. log.info("Called createHandle (not implemented)");
```

[Table of Contents] [Description]

Path 446:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
135. public boolean deleteHandle(byte[] theHandle) throws
HandleException //handleplugin.java
      |...
      |140. log.info("Called deleteHandle (not implemented)");
```

[Table of Contents] [Description]

Path 447:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
149. public void updateValue(byte[] theHandle, HandleValue[]
values) //handleplugin.java
      |...
      |155. log.info("Called updateValue (not implemented)");
```

[Table of Contents] [Description]

Path 448:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
162. public void deleteAllRecords() throws HandleException //handleplugin.java
      |...
      |167. log.info("Called deleteAllRecords (not implemented)");
```

[Table of Contents] [Description]

Path 449:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
174. public void checkpointDatabase() throws HandleException //handleplugin.java
      |...
      |179. log.info("Called checkpointDatabase (not implemented)");
```

[Table of Contents] [Description]

Path 450:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
186. public void shutdown() //handleplugin.java
      |...
      |191. log.info("Called shutdown (not implemented)");
```

[Table of Contents] [Description]

Path 451:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
198. public void scanHandles(ScanCallback callback) throws
HandleException //handleplugin.java
      ...
203. log.info("Called scanHandles (not implemented)");
```

[Table of Contents] [Description]

Path 452:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
210. public void scanNAs(ScanCallback callback) throws
HandleException //handleplugin.java
      ...
215. log.info("Called scanNAs (not implemented)");
```

[Table of Contents] [Description]

Path 453:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
238. public byte[][] getRawHandleValues(byte[] theHandle, int[]
indexList, //handleplugin.java
      ...
243. log.info("Called getRawHandleValues");
```

[Table of Contents] [Description]

Path 454:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
333. public boolean haveNA(byte[] theHandle) throws
HandleException //handleplugin.java
      ...
337. log.info("Called haveNA");
```

[Table of Contents] [Description]

Path 455:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
374. public Enumeration getHandlesForNA(byte[] theNAHandle) //handleplugin.java
      ...
381. log.info("Called getHandlesForNA for NA " + naHandle);
```

[Table of Contents] [Description]

Path 456:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
181. private static void createHarmonyData(Context context, Object
historyObj, //historymanager.java
      |...
      |324. fw.write(data);
```

[Table of Contents] [Description]

Path 457:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
598. private static void store(Context context, String
serialization) //historymanager.java
      |...
      |617. fw.write(serialization);
```

[Table of Contents] [Description]

Path 458:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
987. private static String id2Filename(int id) throws
IOException //historymanager.java
      |...
      |1016. log.debug("Filename for " + id + " is " + theName);
```

[Table of Contents] [Description]

Path 459:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
99. public class DSIndexer //dsindexer.java
      |...
      |170. log.warn("Malformed configuration line: search.index." + i);
```

[Table of Contents] [Description]

Path 460:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
      |
```

```
...
493. log.info("Removing " + line.getOptionValue("r") + " from Index");
```

[Table of Contents] [Description]

Path 461:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
...
498. log.info("Optimizing Index");
```

[Table of Contents] [Description]

Path 462:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
...
503. log.info("Cleaning Index");
```

[Table of Contents] [Description]

Path 463:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
...
508. log.info("Updating Index");
```

[Table of Contents] [Description]

Path 464:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
...
513. log.info("(Re)building index from scratch.");
```

[Table of Contents] [Description]

Path 465:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
      ...
      520. log.info("Done with indexing");
```

[Table of Contents] [Description]

Path 466:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
592. public static void cleanIndex(Context context) throws IOException,
SQLException //dsindexer.java
      ...
      607. log.info("Deleting: " + handle);
```

[Table of Contents] [Description]

Path 467:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
592. public static void cleanIndex(Context context) throws IOException,
SQLException //dsindexer.java
      ...
      614. log.debug("Keeping: " + handle);
```

[Table of Contents] [Description]

Path 468:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
592. public static void cleanIndex(Context context) throws IOException,
SQLException //dsindexer.java
      ...
      619. log.debug("Encountered deleted doc: " + i);
```

[Table of Contents] [Description]

Path 469:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      ...
      883. log.debug("Building Item: " + handle);
```

[Table of Contents] [Description]

Path 470:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      |...
      1030. log.debug(" Added Metadata");
```

[Table of Contents] [Description]

Path 471:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
118. public static List harvest(Context context, DSpaceObject scope, //harvest.java
      |...
      222. log.debug(LogManager.getHeader(context, "harvest SQL", query));
```

[Table of Contents] [Description]

Path 472:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      |...
      167. log.error("No default assetstore");
```

[Table of Contents] [Description]

Path 473:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      |...
      205. log.error("Problem with assetstore " + i);
```

[Table of Contents] [Description]

Path 474:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      |...
      220. log.error("SRB FileSystem is null for assetstore " + i);
```

[Table of Contents] [Description]

Path 475:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      |
      ...
      |
231. log.error("srb.parentdir is undefined for assetstore " + i);
```

[Table of Contents] [Description]

Path 476:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      |
      ...
      |
235. log.error("Unexpected " + o.getClass().toString()
```

[Table of Contents] [Description]

Path 477:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
281. public static int store(Context context, InputStream
is) //bitstreamstoragemanager.java
      |
      ...
      |
365. log.debug("Stored bitstream " + bitstream_id + " in file "
```

[Table of Contents] [Description]

Path 478:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
assetstore, //bitstreamstoragemanager.java
      |
      ...
      |
460. final int count = dis.read(buffer, 0, BUFFER_SIZE);
```

[Table of Contents] [Description]

Path 479:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
assetstore, //bitstreamstoragemanager.java
```

```
...
474. log.error("File: " + file.getAbsolutePath()
```

[Table of Contents] [Description]

Path 480:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
assetstore, //bitstreamstoragemanager.java
...
509. log.debug("Stored bitstream " + bitstream_id + " in file "
```

[Table of Contents] [Description]

Path 481:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
767. private static GeneralFile getFile(TableRow bitstream) throws
IOException //bitstreamstoragemanager.java
...
821. log.debug("Local filename for " + sInternalId + " is "
```

[Table of Contents] [Description]

Path 482:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
767. private static GeneralFile getFile(TableRow bitstream) throws
IOException //bitstreamstoragemanager.java
...
830. log.debug("SRB filename for " + sInternalId + " is "
```

[Table of Contents] [Description]

Path 483:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. public static TableRowIterator queryTable(Context context, String
table, //databasemanager.java
...
153. log.debug("Running query \"" + query + "\"");
```

[Table of Contents] [Description]

Path 484:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
148. public static TableRowIterator queryTable(Context context, String
table, //databasemanager.java
     |...
     |159. TableRowIterator retTRI = new TableRowIterator(statement.executeQuery(),
```

[Table of Contents] [Description]

Path 485:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
248. public static TableRowIterator query(Context context, String query, Object[]
parameters) //databasemanager.java
     |...
     |253. log.debug("Running query \"" + query + "\"");
```

[Table of Contents] [Description]

Path 486:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
248. public static TableRowIterator query(Context context, String query, Object[]
parameters) //databasemanager.java
     |...
     |259. TableRowIterator retTRI = new TableRowIterator(statement.executeQuery());
```

[Table of Contents] [Description]

Path 487:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
331. public static TableRowIterator queryPreparedTable(String
table, //databasemanager.java
     |...
     |334. TableRowIterator retTRI = new TableRowIterator(statement.executeQuery(),
```

[Table of Contents] [Description]

Path 488:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
351. public static TableRowIterator queryPrepared(PreparedStatement
statement) //databasemanager.java
     |...
```

```
354. TableRowIterator retTRI = new TableRowIterator(statement.executeQuery());
```

[Table of Contents] [Description]

Path 489:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
504. public static int updateQuery(Context context, String query, Object[]
parameters) //databasemanager.java
...
511. log.debug("Running query \"" + query + "\"");
```

[Table of Contents] [Description]

Path 490:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
767. public static void insert(Context context, TableRow row) //databasemanager.java
...
782. ResultSet rs = statement.executeQuery(myQuery);
```

[Table of Contents] [Description]

Path 491:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
767. public static void insert(Context context, TableRow row) //databasemanager.java
...
786. int newID = rs.getInt(1);
```

[Table of Contents] [Description]

Path 492:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
...
1221. ResultSetMetaData meta = results.getMetaData();
```

[Table of Contents] [Description]

Path 493:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
1239. row.setColumn(name, results.getBoolean(i));
```

[Table of Contents] [Description]

Path 494:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
1249. long longValue = results.getLong(i);
```

[Table of Contents] [Description]

Path 495:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
1256. row.setColumn(name, results.getInt(i));
```

[Table of Contents] [Description]

Path 496:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
1260. row.setColumn(name, results.getLong(i));
```

[Table of Contents] [Description]

Path 497:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
1265. row.setColumn(name, results.getString(i));
```

[Table of Contents] [Description]

Path 498:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      |...
      1291. row.setColumn(name, results.getDate(i));
```

[Table of Contents] [Description]

Path 499:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      |...
      1295. row.setColumn(name, results.getTime(i));
```

[Table of Contents] [Description]

Path 500:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      |...
      1299. row.setColumn(name, results.getTimestamp(i));
```

[Table of Contents] [Description]

Path 501:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
      |...
      1405. log.debug("Running query \"" + sql + "\"");
```

[Table of Contents] [Description]

Path 502:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
58. public static void main(String[] argv) //initializedatabase.java
      |
```

```
...
63. log.warn("Schema file not specified");
```

[Table of Contents] [Description]

Path 503:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
58. public static void main(String[] argv) //initializedatabase.java
...
67. log.info("Initializing Database");
```

[Table of Contents] [Description]

Path 504:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
127. public static WorkflowItem find(Context context, int id) //workflowitem.java
...
145. log.debug(LogManager.getHeader(context, "find_workflow_item",
```

[Table of Contents] [Description]

Path 505:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
127. public static WorkflowItem find(Context context, int id) //workflowitem.java
...
155. log.debug(LogManager.getHeader(context, "find_workflow_item",
```

[Table of Contents] [Description]

Path 506:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
348. public void update() throws SQLException, IOException,
AuthorizeException //workflowitem.java
...
351. log.info(LogManager.getHeader(ourContext, "update_workflow_item",
```

[Table of Contents] [Description]

Path 507:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
170. public static WorkflowItem start(Context c, WorkspaceItem
wsi) //workflowmanager.java
      |...
      177. log.info(LogManager.getHeader(c, "start_workflow", "workspace_item_id="
```

[Table of Contents] [Description]

Path 508:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
288. public static void claim(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      |...
      320. log.info(LogManager.getHeader(c, "claim_task", "workflow_item_id="
```

[Table of Contents] [Description]

Path 509:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
341. public static void advance(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      |...
      378. log.info(LogManager.getHeader(c, "advance_workflow",
```

[Table of Contents] [Description]

Path 510:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
395. public static void unclaim(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      |...
      427. log.info(LogManager.getHeader(c, "unclaim_workflow",
```

[Table of Contents] [Description]

Path 511:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
446. public static void abort(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      |...
      459. log.info(LogManager.getHeader(c, "abort_workflow", "workflow_item_id="
```

[Table of Contents] [Description]

Path 512:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
641. private static Item archive(Context c, WorkflowItem wfi) //workflowmanager.java
       |
       ...
       |
648. log.info(LogManager.getHeader(c, "archive_item", "workflow_item_id="
```

[Table of Contents] [Description]

Path 513:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
641. private static Item archive(Context c, WorkflowItem wfi) //workflowmanager.java
       |
       ...
       |
655. log.info(LogManager.getHeader(c, "install_item", "workflow_id="
```

[Table of Contents] [Description]

Path 514:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
711. private static WorkspaceItem returnToWorkspace(Context c, WorkflowItem
wfi) //workflowmanager.java
       |
       ...
       |
737. log.info(LogManager.getHeader(c, "return_to_workspace",
```

[Table of Contents] [Description]

Path 515:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
762. public static WorkspaceItem reject(Context c, WorkflowItem wi, EPerson
e, //workflowmanager.java
       |
       ...
       |
793. log.info(LogManager.getHeader(c, "reject_workflow", "workflow_item_id="
```

[Table of Contents] [Description]

Unclosed objects

CWE ID	10031
Description	Closing objects inside a try clause, without a matching Close statement in the finally section may lead to resouce leaking in case of exception.

Likelihood of Exploit Low
Common Consequences In case of exception, open objects may not be closed. Make sure that the marked Close statement, which is within a try clause, has a matching Close statement in the finally clause.
Potential Mitigations Add a matching close statement to the finally clause.
Applicable Platforms All

Path 1:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
83. tri.close();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
103. tri.close();
```

[Table of Contents] [Description]

Path 3:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
629. public static void createOutput() //loganalyser.java
      ...
770. out.close();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
      ...
458. out.close();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
465. private String getScalingAttr(HttpServletRequest hrq, Bitstream
bitstream) //itemlisttag.java
      ...
480. is.close();
```

[Table of Contents] [Description]

Path 6:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
      ...
331. ctx.close();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
      ...
344. ctx.close();
```

[Table of Contents] [Description]

Path 8:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
51. protected void cleanup(Statement stmt, Connection conn, ResultSet
rs) //daosupport.java
      ...
57. rs.close();
```

[Table of Contents] [Description]

Path 9:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
72. protected void cleanup(Statement stmt) //daosupport.java
      ...
78. stmt.close();
```

[Table of Contents] [Description]

Path 10:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
88. protected void cleanup(Connection conn) //daosupport.java
      ...
94. conn.close();
```

[Table of Contents] [Description]

Path 11:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
247. zip.close();
```

[Table of Contents] [Description]

Path 12:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
325. public static String readNewsFile(int position) //configurationmanager.java
      ...
354. br.close();
```

[Table of Contents] [Description]

Path 13:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
374. public static String writeNewsFile(int position, String
news) //configurationmanager.java
      ...
394. out.close();
```

[Table of Contents] [Description]

Path 14:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
410. public static void writeLicenseFile(String
newLicense) //configurationmanager.java
```

```
...
422. out.close();
```

[Table of Contents] [Description]

Path 15:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
...
526. is.close();
```

[Table of Contents] [Description]

Path 16:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
119. public void loadData() //group.java
...
154. tri.close();
```

[Table of Contents] [Description]

Path 17:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
119. public void loadData() //group.java
...
181. tri.close();
```

[Table of Contents] [Description]

Path 18:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
469. private static String serialize(Context context, Object
obj) //historymanager.java
...
489. data.close();
```

[Table of Contents] [Description]

Path 19:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
99. public class DSIndexer //dsindexer.java
      ...
190. openIndex(null,true).close();
```

[Table of Contents] [Description]

Path 20:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
415. protected static synchronized IndexSearcher getSearcher(Context
c) //dsquery.java
      ...
426. searcher.close();
```

[Table of Contents] [Description]

Path 21:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
727. public static void freeConnection(Connection c) //databasemanager.java
      ...
733. c.close();
```

[Table of Contents] [Description]

Path 22:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
      ...
217. results.close();
```

[Table of Contents] [Description]

Path 23:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
      ...
219. results.close();
```

[Table of Contents] [Description]

Path 24:

Query Name - **Unclosed_Objects**

Severity

- Information

```
213. public void close() //tablerowiterator.java
      |
      ...
      229. statemt.close();
```

[Table of Contents] [Description]

Catch Without General Exception Handling

CWE ID	10009
Description	Not handling general exceptions may cause the program to crash.
Likelihood of Exploit	Low
Common Consequences	Program crashing.
Potential Mitigations	Handle the case of general exception.

Demonstrative Examples	The following code catches specific exception but not the general one.
	
	...
	try
	{
	File.Copy(path, path2);
	}
	catch(IOException e)
	{
	Console.WriteLine(e.Message);
	}
	...

Applicable Platforms	All

Path 1:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - Information

```
1. <%-- //index.jsp
      |
      ...
      70. try
```

[Table of Contents] [Description]

Path 2:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - Information

```
1. <%-- //index.jsp
      |
      ...
      71. try
```

[Table of Contents] [Description]

Path 3:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <%-- //formats.jsp
      ...
      69. try
```

[Table of Contents] [Description]

Path 4:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public int[] getSpecialGroups(Context context, HttpServletRequest
request) //mitspecialgroup.java
      ...
      148. try {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //communityfiliator.java
      ...
      153. try
```

[Table of Contents] [Description]

Path 6:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
      ...
      141. try
```

[Table of Contents] [Description]

Path 7:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
      ...
      161. try
```

[Table of Contents] [Description]

Path 8:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
      ...
199. try
```

[Table of Contents] [Description]

Path 9:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
897. private String processHandleFile(Context c, Item i, String
path, //itemimport.java
      ...
926. try
```

[Table of Contents] [Description]

Path 10:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
945. private void processContentsFile(Context c, Item i, String
path, //itemimport.java
      ...
992. try
```

[Table of Contents] [Description]

Path 11:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
323. public static void writeMETS(Context context, Item item, OutputStream os,
boolean fullURL) //metsexport.java
      ...
326. try
```

[Table of Contents] [Description]

Path 12:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
```

```
...
179. try
```

[Table of Contents] [Description]

Path 13:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
118. public Vector getSchemaLocations(String identifier) //dspaceoaicatalog.java
...
130. try
```

[Table of Contents] [Description]

Path 14:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
208. public Map listIdentifiers(String from, String until, String
set, //dspaceoaicatalog.java
...
232. try
```

[Table of Contents] [Description]

Path 15:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
328. public String getRecord(String identifier, String
metadataPrefix) //dspaceoaicatalog.java
...
345. try
```

[Table of Contents] [Description]

Path 16:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
480. public Map listRecords(String resumptionToken) //dspaceoaicatalog.java
...
501. try
```

[Table of Contents] [Description]

Path 17:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
539. private Map doRecordHarvest(String from, String until, String
set, //dspaceoaicatalog.java
      ...
555. try
```

[Table of Contents] [Description]

Path 18:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
539. private Map doRecordHarvest(String from, String until, String
set, //dspaceoaicatalog.java
      ...
573. try
```

[Table of Contents] [Description]

Path 19:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
654. public Map listSets() throws NoSetHierarchyException, //dspaceoaicatalog.java
      ...
664. try
```

[Table of Contents] [Description]

Path 20:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
850. private Object[] decodeResumptionToken(String token) //dspaceoaicatalog.java
      ...
856. try
```

[Table of Contents] [Description]

Path 21:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
343. public static void processLogs(Context context, String
myLogDir, //loganalyser.java
      ...
393. try
```

[Table of Contents] [Description]

Path 22:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
343. public static void processLogs(Context context, String
myLogDir, //loganalyser.java
      ...
539. try
```

[Table of Contents] [Description]

Path 23:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
629. public static void createOutput() //loganalyser.java
      ...
766. try
```

[Table of Contents] [Description]

Path 24:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
898. public static void readConfig(String configFile) //loganalyser.java
      ...
906. try
```

[Table of Contents] [Description]

Path 25:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1031. public static Date parseDate(String date) //loganalyser.java
      ...
1036. try
```

[Table of Contents] [Description]

Path 26:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
      ...
452. try
```

[Table of Contents] [Description]

Path 27:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
547. public static void readMap(String map) //reportgenerator.java
       ...
555. try
```

[Table of Contents] [Description]

Path 28:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
624. public static void readInput(String input) //reportgenerator.java
       ...
633. try
```

[Table of Contents] [Description]

Path 29:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
624. public static void readInput(String input) //reportgenerator.java
       ...
749. try
```

[Table of Contents] [Description]

Path 30:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public void doFilter(ServletRequest request, ServletResponse
response, //adminonlyfilter.java
       ...
89. try
```

[Table of Contents] [Description]

Path 31:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
78. public void doFilter(ServletRequest request, ServletResponse
response, //registeredonlyfilter.java
       ...
87. try
```

[Table of Contents] [Description]

Path 32:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
72. try
```

[Table of Contents] [Description]

Path 33:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
72. try
```

[Table of Contents] [Description]

Path 34:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. public int doStartTag() throws JspException //datetag.java
      ...
112. try
```

[Table of Contents] [Description]

Path 35:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public int doStartTag() throws JspException //includetag.java
      ...
82. try
```

[Table of Contents] [Description]

Path 36:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
159. try
```

[Table of Contents] [Description]

Path 37:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. public void setHighlightrow(String highlightRowIn) //itemlisttag.java
       ...
380. try
```

[Table of Contents] [Description]

Path 38:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
465. private String getScalingAttr(HttpServletRequest hrq, Bitstream
bitstream) //itemlisttag.java
       ...
470. try
```

[Table of Contents] [Description]

Path 39:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
524. private String getThumbMarkup(HttpServletRequest hrq, Item
item) //itemlisttag.java
       ...
529. try
```

[Table of Contents] [Description]

Path 40:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
524. private String getThumbMarkup(HttpServletRequest hrq, Item
item) //itemlisttag.java
       ...
561. try
```

[Table of Contents] [Description]

Path 41:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public int doStartTag() throws JspException //itempreviewtag.java
```

```
...
86. try
```

[Table of Contents] [Description]

Path 42:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
167. public int doStartTag() throws JspException //itemtag.java
...
169. try
```

[Table of Contents] [Description]

Path 43:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
528. try
```

[Table of Contents] [Description]

Path 44:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
776. try
```

[Table of Contents] [Description]

Path 45:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public int doStartTag() throws JspException //layouttag.java
...
351. try
```

[Table of Contents] [Description]

Path 46:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
388. public int doEndTag() throws JspException //layouttag.java
```

```
...
399. try
```

[Table of Contents] [Description]

Path 47:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
95. public int doAfterBody() throws JspException //popuptag.java
...
128. try
```

[Table of Contents] [Description]

Path 48:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
...
134. try
```

[Table of Contents] [Description]

Path 49:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
...
132. try
```

[Table of Contents] [Description]

Path 50:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
71. public int doStartTag() throws JspException //sfxlinktag.java
...
73. try
```

[Table of Contents] [Description]

Path 51:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
84. protected void doDSGet(Context context, HttpServletRequest
```

```
request, //bitstreamservlet.java
    ...
128. try
```

[Table of Contents] [Description]

Path 52:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
344. try
```

[Table of Contents] [Description]

Path 53:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
    ...
128. try
```

[Table of Contents] [Description]

Path 54:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
    ...
137. try
```

[Table of Contents] [Description]

Path 55:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
252. try
```

[Table of Contents] [Description]

Path 56:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
263. private boolean itemsChanged(Context context, DSpaceObject dso, long
timeStamp) //feedservlet.java
      ...
275. try {
```

[Table of Contents] [Description]

Path 57:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
      ...
440. try
```

[Table of Contents] [Description]

Path 58:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
      ...
492. try
```

[Table of Contents] [Description]

Path 59:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
      ...
522. try
```

[Table of Contents] [Description]

Path 60:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
      ...
```

```
109. try
```

[Table of Contents] [Description]

Path 61:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
...
195. try
```

[Table of Contents] [Description]

Path 62:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
...
273. try
```

[Table of Contents] [Description]

Path 63:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
...
289. try
```

[Table of Contents] [Description]

Path 64:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
157. try
```

[Table of Contents] [Description]

Path 65:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
     ...
170. try
```

[Table of Contents] [Description]

Path 66:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
     ...
299. try
```

[Table of Contents] [Description]

Path 67:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
     ...
356. try
```

[Table of Contents] [Description]

Path 68:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
     ...
413. try
```

[Table of Contents] [Description]

Path 69:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
512. private void processRejectReason(Context context, //mydspaceservlet.java
     ...
522. try
```

[Table of Contents] [Description]

Path 70:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     |...
     |263. try
```

[Table of Contents] [Description]

Path 71:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     |...
     |292. try
```

[Table of Contents] [Description]

Path 72:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     |...
     |339. try {
```

[Table of Contents] [Description]

Path 73:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
     |...
     |99. try
```

[Table of Contents] [Description]

Path 74:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
     |...
```

```
163. try
```

[Table of Contents] [Description]

Path 75:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
     ...
135. FileInputStream fir = null;
```

[Table of Contents] [Description]

Path 76:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
     ...
235. try
```

[Table of Contents] [Description]

Path 77:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
     ...
236. try
```

[Table of Contents] [Description]

Path 78:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
     ...
260. try
```

[Table of Contents] [Description]

Path 79:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1015. try
```

[Table of Contents] [Description]

Path 80:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1301. try
```

[Table of Contents] [Description]

Path 81:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1332. try
```

[Table of Contents] [Description]

Path 82:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1376. try
```

[Table of Contents] [Description]

Path 83:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
    ...
1755. try
```

[Table of Contents] [Description]

Path 84:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
    ...
176. try
```

[Table of Contents] [Description]

Path 85:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
77. protected void doDSPost(Context context, HttpServletRequest
request, //epersonadminservlet.java
    ...
260. try
```

[Table of Contents] [Description]

Path 86:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
    ...
114. try
```

[Table of Contents] [Description]

Path 87:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
    ...
152. try
```

[Table of Contents] [Description]

Path 88:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. protected void doDSPost(Context context, HttpServletRequest
```

```
request, //metadatafieldregistryservlet.java
    ...
206. try
```

[Table of Contents] [Description]

Path 89:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
81. protected void doDSPost(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
    ...
101. try
```

[Table of Contents] [Description]

Path 90:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
250. public static void loggedIn(Context context, HttpServletRequest
request, //authenticate.java
    ...
259. try
```

[Table of Contents] [Description]

Path 91:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. public static void authorizeAnyOf(Context c, DSpaceObject o, int[]
actions) //authorizemanager.java
    ...
98. try
```

[Table of Contents] [Description]

Path 92:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
217. public static boolean authorizeActionBoolean(Context c, DSpaceObject
o, //authorizemanager.java
    ...
227. try
```

[Table of Contents] [Description]

Path 93:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
984. protected static int countMatches(BrowseScope scope, String
itemValue, //browse.java
     |...
     1009. try {
```

[Table of Contents] [Description]

Path 94:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1753. public static BrowseInfo get(BrowseScope key) //browse.java
     |...
     1764. try
```

[Table of Contents] [Description]

Path 95:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1829. public static boolean indexHasChanged(BrowseScope key) throws
SQLException //browse.java
     |...
     1833. try
```

[Table of Contents] [Description]

Path 96:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
65. public static void main(String[] argv) //initializebrowse.java
     |...
     69. try
```

[Table of Contents] [Description]

Path 97:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
177. public void update(BitstreamInfo info) //bitstreaminfodao.java
     |...
     188. try
```

[Table of Contents] [Description]

Path 98:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
228. public BitstreamInfo findByBitstreamId(int id) //bitstreaminfodao.java
       ...
234. try
```

[Table of Contents] [Description]

Path 99:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
278. public void updateMissingBitstreams() //bitstreaminfodao.java
       ...
283. try
```

[Table of Contents] [Description]

Path 100:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
317. protected int deleteBitstreamInfo(int id, Connection
conn) //bitstreaminfodao.java
       ...
323. try
```

[Table of Contents] [Description]

Path 101:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
354. public int deleteBitstreamInfoWithHistory(int id) //bitstreaminfodao.java
       ...
359. try
```

[Table of Contents] [Description]

Path 102:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
388. public int getOldestBitstream() //bitstreaminfodao.java
       ...
394. try
```

[Table of Contents] [Description]

Path 103:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
430. public int getOldestBitstream(Timestamp lessThanDate) //bitstreaminfodao.java
       ...
436. try
```

[Table of Contents] [Description]

Path 104:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
472. public List getItemBitstreams(int itemId) //bitstreaminfodao.java
       ...
480. try
```

[Table of Contents] [Description]

Path 105:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
515. public List getCollectionBitstreams(int collectionId) //bitstreaminfodao.java
       ...
523. try
```

[Table of Contents] [Description]

Path 106:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
558. public List getCommunityBitstreams(int communityId) //bitstreaminfodao.java
       ...
566. try
```

[Table of Contents] [Description]

Path 107:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
320. private void processBitstream(BitstreamInfo info) //checkercommand.java
       ...
329. try
```

[Table of Contents] [Description]

Path 108:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
75. public void insertHistory(BitstreamInfo info) //checksumhistorydao.java
      ...
86. try
```

[Table of Contents] [Description]

Path 109:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. protected int deleteHistoryForBitstreamInfo(int id, Connection
conn) //checksumhistorydao.java
      ...
127. try
```

[Table of Contents] [Description]

Path 110:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
154. protected void updateMissingBitstreams(Connection conn) throws
SQLException //checksumhistorydao.java
      ...
157. try
```

[Table of Contents] [Description]

Path 111:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
216. public int prune(Map interests) //checksumhistorydao.java
      ...
219. try
```

[Table of Contents] [Description]

Path 112:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
85. public String getChecksumCheckStr(String code) //checksumresultdao.java
```

```
...
92. try
```

[Table of Contents] [Description]

Path 113:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public List listAllCodes() //checksumresultdao.java
...
131. try
```

[Table of Contents] [Description]

Path 114:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
167. public static void main(String[] args) //dailyreportemailer.java
...
198. try
```

[Table of Contents] [Description]

Path 115:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
51. protected void cleanup(Statement stmt, Connection conn, ResultSet
rs) //daosupport.java
...
55. try
```

[Table of Contents] [Description]

Path 116:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. protected void cleanup(Statement stmt) //daosupport.java
...
76. try
```

[Table of Contents] [Description]

Path 117:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
88. protected void cleanup(Connection conn) //daosupport.java
      ...
92. try
```

[Table of Contents] [Description]

Path 118:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
101. private void init() //handledispatcher.java
      ...
107. try
```

[Table of Contents] [Description]

Path 119:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
79. public synchronized int next() //listdispatcher.java
      ...
81. try
```

[Table of Contents] [Description]

Path 120:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
128. public List getBitstreamResultTypeReport(Date startDate, Date
endDate, //reporterdao.java
      ...
137. try
```

[Table of Contents] [Description]

Path 121:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
185. public List getNotProcessedBitstreamsReport(Date startDate, Date
endDate) //reporterdao.java
      ...
193. try
```

[Table of Contents] [Description]

Path 122:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
236. public List getUnknownBitstreams() //reporterdao.java
       ...
244. try
```

[Table of Contents] [Description]

Path 123:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
79. public static ResultsPruner getDefaultPruner() //resultspruner.java
       ...
81. try
```

[Table of Contents] [Description]

Path 124:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public static ResultsPruner getPruner(String propsFile) //resultspruner.java
       ...
108. try
```

[Table of Contents] [Description]

Path 125:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public static ResultsPruner getPruner(String propsFile) //resultspruner.java
       ...
123. try
```

[Table of Contents] [Description]

Path 126:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
436. public void setShortDescription(String s) //bitstreamformat.java
       ...
441. try {
```

[Table of Contents] [Description]

Path 127:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
     ...
390. try
```

[Table of Contents] [Description]

Path 128:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
868. public boolean canEditBoolean() throws java.sql.SQLException //collection.java
     ...
870. try
```

[Table of Contents] [Description]

Path 129:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
     ...
346. try
```

[Table of Contents] [Description]

Path 130:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
905. public boolean canEditBoolean() throws java.sql.SQLException //community.java
     ...
907. try
```

[Table of Contents] [Description]

Path 131:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public DCDate(String fromDC) //dcdate.java
     ...
133. try
```

[Table of Contents] [Description]

Path 132:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public Element disseminateElement(DSpaceObject
dso) //metsdisseminationcrosswalk.java
      |...
      |156. try
```

[Table of Contents] [Description]

Path 133:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public Element disseminateElement(DSpaceObject
dso) //metsdisseminationcrosswalk.java
      |...
      |168. try
```

[Table of Contents] [Description]

Path 134:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
183. public static modsTriple create(String qdc, String xml, String
xpath) //modsdisseminationcrosswalk.java
      |...
      |190. try
```

[Table of Contents] [Description]

Path 135:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |266. try
```

[Table of Contents] [Description]

Path 136:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
      |...
```

```
350. try
```

[Table of Contents] [Description]

Path 137:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
...
145. try
```

[Table of Contents] [Description]

Path 138:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
...
307. try
```

[Table of Contents] [Description]

Path 139:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. protected XSLTransformer getTransformer(String direction) //xsltcrosswalk.java
...
221. try
```

[Table of Contents] [Description]

Path 140:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public Namespace[] getNamespaces() //xsltdisseminationcrosswalk.java
...
181. try
```

[Table of Contents] [Description]

Path 141:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
192. public String getSchemaLocation() //xsltdisseminationcrosswalk.java
```

```
...
194. try
```

[Table of Contents] [Description]

Path 142:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public Element disseminateElement(DSpaceObject
dso) //xsltdisseminationcrosswalk.java
...
218. try
```

[Table of Contents] [Description]

Path 143:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
231. public List disseminateList(DSpaceObject dso) //xsltdisseminationcrosswalk.java
...
244. try
```

[Table of Contents] [Description]

Path 144:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public boolean preferList() //xsltdisseminationcrosswalk.java
...
284. try
```

[Table of Contents] [Description]

Path 145:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
147. public void ingest(Context context, DSpaceObject dso, List
metadata) //xsltingestioncrosswalk.java
...
158. try
```

[Table of Contents] [Description]

Path 146:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
175. public void ingest(Context context, DSpaceObject dso, Element
root) //xsltingestioncrosswalk.java
      ...
185. try
```

[Table of Contents] [Description]

Path 147:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
295. private void setMdType(MdWrap mdWrap, String
mdtype) //abstractmetsdisseminator.java
      ...
297. try
```

[Table of Contents] [Description]

Path 148:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
317. try
```

[Table of Contents] [Description]

Path 149:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
389. try
```

[Table of Contents] [Description]

Path 150:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
558. try
```

[Table of Contents] [Description]

Path 151:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
193. try
```

[Table of Contents] [Description]

Path 152:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
224. try
```

[Table of Contents] [Description]

Path 153:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
267. public static METSManifest create(InputStream is, boolean
validate) //metsmanifest.java
      ...
288. try
```

[Table of Contents] [Description]

Path 154:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
353. public List getMdFiles() //metsmanifest.java
      ...
358. try
```

[Table of Contents] [Description]

Path 155:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
386. public Element getOriginalFile(Element file) //metsmanifest.java
      ...
392. try
```

[Table of Contents] [Description]

Path 156:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
549. public List getMdContentAsXml(Element mdSec, Mdref
callback) //metsmanifest.java
     |...
     |552. try
```

[Table of Contents] [Description]

Path 157:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
662. private void crosswalkMdContent(Element mdSec, Mdref
callback, //metsmanifest.java
     |...
     |669. try
```

[Table of Contents] [Description]

Path 158:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
699. private Element getElementByXPath(String path, boolean
nullOk) //metsmanifest.java
     |...
     |702. try
```

[Table of Contents] [Description]

Path 159:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     |...
     |197. try
```

[Table of Contents] [Description]

Path 160:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. public static int getIntProperty(String property) //configurationmanager.java
      ...
141. try
```

[Table of Contents] [Description]

Path 161:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
325. public static String readNewsFile(int position) //configurationmanager.java
      ...
340. try
```

[Table of Contents] [Description]

Path 162:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
374. public static String writeNewsFile(int position, String
news) //configurationmanager.java
      ...
387. try
```

[Table of Contents] [Description]

Path 163:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
410. public static void writeLicenseFile(String
newLicense) //configurationmanager.java
      ...
415. try
```

[Table of Contents] [Description]

Path 164:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
      ...
468. try
```

[Table of Contents] [Description]

Path 165:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
       ...
544. try
```

[Table of Contents] [Description]

Path 166:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
781. public static void main(String[] argv) //configurationmanager.java
       ...
785. try
```

[Table of Contents] [Description]

Path 167:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
247. public void abort() //context.java
       ...
249. try
```

[Table of Contents] [Description]

Path 168:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
215. private static Object getAnonymousPlugin(String classname) //pluginmanager.java
       ...
218. try
```

[Table of Contents] [Description]

Path 169:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
       ...
364. try
```

[Table of Contents] [Description]

Path 170:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
434. public static String[] getAllPluginNames(Class intfc) //pluginmanager.java
      ...
436. try
```

[Table of Contents] [Description]

Path 171:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
494. private static boolean checkClassname(String iname, String
msg) //pluginmanager.java
      ...
496. try
```

[Table of Contents] [Description]

Path 172:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
509. private static boolean checkSelfNamed(String iname) //pluginmanager.java
      ...
511. try
```

[Table of Contents] [Description]

Path 173:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
537. private static void checkNames(String iname) //pluginmanager.java
      ...
539. try
```

[Table of Contents] [Description]

Path 174:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public static byte[] getMD5Bytes(byte[] data) //utils.java
      ...
123. try
```

[Table of Contents] [Description]

Path 175:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
     |...
     |198. try
```

[Table of Contents] [Description]

Path 176:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
241. public static EPerson[] search(Context context, String query, int offset, int
limit) //eperson.java
     |...
     |254. try {
```

[Table of Contents] [Description]

Path 177:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
299. public static int searchResultCount(Context context, String
query) //eperson.java
     |...
     |307. try {
```

[Table of Contents] [Description]

Path 178:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
775. public static Group[] search(Context context, String query, int offset, int
limit) //group.java
     |...
     |787. try {
```

[Table of Contents] [Description]

Path 179:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
831. public static int searchResultCount(Context context, String query) //group.java
     |...
```

```
839. try {
```

[Table of Contents] [Description]

Path 180:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
153. try
```

[Table of Contents] [Description]

Path 181:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
195. try
```

[Table of Contents] [Description]

Path 182:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
215. try
```

[Table of Contents] [Description]

Path 183:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
     ...
310. try
```

[Table of Contents] [Description]

Path 184:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
```

```
context) //ldapauthentication.java
    ...
326. try
```

[Table of Contents] [Description]

Path 185:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
    ...
377. try
```

[Table of Contents] [Description]

Path 186:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
    ...
179. try
```

[Table of Contents] [Description]

Path 187:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
247. public static void processDaily(Context context) throws
SQLException, //subscribe.java
    ...
271. try
```

[Table of Contents] [Description]

Path 188:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
247. public static void processDaily(Context context) throws
SQLException, //subscribe.java
    ...
297. try
```

[Table of Contents] [Description]

Path 189:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
322. public static void sendEmail(Context context, EPerson eperson, //subscribe.java
       ...
347. try {
```

[Table of Contents] [Description]

Path 190:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
110. public class X509Authentication //x509authentication.java
       ...
143. try {
```

[Table of Contents] [Description]

Path 191:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
110. public class X509Authentication //x509authentication.java
       ...
164. try
```

[Table of Contents] [Description]

Path 192:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
       ...
248. try
```

[Table of Contents] [Description]

Path 193:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
       ...
262. try
```

[Table of Contents] [Description]

Path 194:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
      |...
      |277. try
```

[Table of Contents] [Description]

Path 195:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
      |...
      |287. try {
```

[Table of Contents] [Description]

Path 196:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
380. public int authenticate(Context context, //x509authentication.java
      |...
      |398. try
```

[Table of Contents] [Description]

Path 197:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public byte[][] getRawHandleValues(byte[] theHandle, int[]
indexList, //handleplugin.java
      |...
      |313. try
```

[Table of Contents] [Description]

Path 198:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
374. public Enumeration getHandlesForNA(byte[] theNAHandle) //handleplugin.java
      |...
      |386. try
```

[Table of Contents] [Description]

Path 199:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
374. public Enumeration getHandlesForNA(byte[] theNAHandle) //handleplugin.java
      ...
416. try
```

[Table of Contents] [Description]

Path 200:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
469. private static String serialize(Context context, Object
obj) //historymanager.java
      ...
487. try
```

[Table of Contents] [Description]

Path 201:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
204. public static boolean hasLicense(Context context, Item
item) //creativecommons.java
      ...
216. try
```

[Table of Contents] [Description]

Path 202:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
99. public class DSIndexer //dsindexer.java
      ...
188. try {
```

[Table of Contents] [Description]

Path 203:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public static QueryResults doQuery(Context c, QueryArgs args) //dsquery.java
      ...
144. try
```

[Table of Contents] [Description]

Path 204:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
415. protected static synchronized IndexSearcher getSearcher(Context
c) //dsquery.java
      ...
423. try
```

[Table of Contents] [Description]

Path 205:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. public class BitstreamStorageManager //bitstreamstoragemanager.java
      ...
211. try {
```

[Table of Contents] [Description]

Path 206:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
281. public static int store(Context context, InputStream
is) //bitstreamstoragemanager.java
      ...
291. try
```

[Table of Contents] [Description]

Path 207:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
281. public static int store(Context context, InputStream
is) //bitstreamstoragemanager.java
      ...
339. try
```

[Table of Contents] [Description]

Path 208:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
```

```
assetstore, //bitstreamstoragemanager.java
    ...
396. try {
```

[Table of Contents] [Description]

Path 209:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
assetstore, //bitstreamstoragemanager.java
    ...
439. try
```

[Table of Contents] [Description]

Path 210:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
386. public static int register(Context context, int
assetstore, //bitstreamstoragemanager.java
    ...
483. try
```

[Table of Contents] [Description]

Path 211:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
594. public static void cleanup(boolean deleteDbRecords) throws SQLException,
IOException //bitstreamstoragemanager.java
    ...
600. try
```

[Table of Contents] [Description]

Path 212:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public static void main(String[] argv) //cleanup.java
    ...
84. try
```

[Table of Contents] [Description]

Path 213:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
504. public static int updateQuery(Context context, String query, Object[]
parameters) //databasemanager.java
        ...
525. try
```

[Table of Contents] [Description]

Path 214:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
727. public static void freeConnection(Connection c) //databasemanager.java
        ...
729. try
```

[Table of Contents] [Description]

Path 215:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1028. public static void loadSql(String sql) throws
SQLException //databasemanager.java
        ...
1030. try
```

[Table of Contents] [Description]

Path 216:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1049. public static void loadSql(Reader r) throws SQLException,
IOException //databasemanager.java
        ...
1132. try
```

[Table of Contents] [Description]

Path 217:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
        ...
1269. try
```

[Table of Contents] [Description]

Path 218:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
      |...
      |1498. try
```

[Table of Contents] [Description]

Path 219:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
      |...
      |215. try
```

[Table of Contents] [Description]

Path 220:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
      |...
      |226. try
```

[Table of Contents] [Description]

Path 221:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
469. private static boolean doState(Context c, WorkflowItem wi, int
newstate, //workflowmanager.java
      |...
      |599. try
```

[Table of Contents] [Description]

Path 222:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
664. private static void notifyOfArchive(Context c, Item i, Collection
coll) //workflowmanager.java
```

```
...
667. try
```

[Table of Contents] [Description]

Path 223:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
826. private static void notifyGroupOfTask(Context c, WorkflowItem
wi, //workflowmanager.java
...
841. try
```

[Table of Contents] [Description]

Incorrect Block Delimitation

CWE ID	483
Description	In some languages, forgetting to explicitly delimit a block can result in a logic error that can, in turn, have security implications.
Likelihood of Exploit	Low
Common Consequences	This is a general logic error -- with all the potential consequences that this entails.
Potential Mitigations	Implementation: Always use explicit block delimitation and use static-analysis technologies to enforce this practice.
Demonstrative Examples	In this example, when the condition is true, the intention may be that both x and y run. if (condition==true) x; y;
Context Notes	In many languages, braces are optional for blocks, and -- in a case where braces are omitted -- it is possible to insert a logic error where a statement is thought to be in a block but is not. This is a common and well known reliability error.
Node Relationships	ChildOf - Code Quality (398)
Source Taxonomies	CLASP - Incorrect block delimitation
Applicable Platforms	All

Path 1:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //community-home.jsp
...
100. if(sidebar == null)
```

[Table of Contents] [Description]

Path 2:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //formats.jsp
       ...
150. if ( formats[i].isInternal() )
```

[Table of Contents] [Description]

Path 3:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //profile-form.jsp
       ...
71. if (lastName == null) lastName = "";
```

[Table of Contents] [Description]

Path 4:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //profile-form.jsp
       ...
74. if (firstName == null) firstName = "";
```

[Table of Contents] [Description]

Path 5:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //profile-form.jsp
       ...
77. if (phone == null) phone = "";
```

[Table of Contents] [Description]

Path 6:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //cancel.jsp
       ...
74. if (displayStep==null) displayStep = step;
```

[Table of Contents] [Description]

Path 7:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
126. void doPersonalName(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
141. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 8:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
126. void doPersonalName(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
163. if (repeatable)
```

[Table of Contents] [Description]

Path 9:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
126. void doPersonalName(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
168. if (repeatable)
```

[Table of Contents] [Description]

Path 10:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
126. void doPersonalName(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
171. if (i == 0)
```

[Table of Contents] [Description]

Path 11:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
126. void doPersonalName(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
```

```
178. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 12:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
...
239. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 13:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
...
244. if (i == 0)
```

[Table of Contents] [Description]

Path 14:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
...
251. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 15:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
...
262. if (repeatable)
```

[Table of Contents] [Description]

Path 16:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
```

```
...
287. if (repeatable)
```

[Table of Contents] [Description]

Path 17:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. void doDate(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
...
298. if (repeatable)
```

[Table of Contents] [Description]

Path 18:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. void doSeriesNumber(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
...
348. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 19:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. void doSeriesNumber(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
...
353. if (i == 0)
```

[Table of Contents] [Description]

Path 20:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. void doSeriesNumber(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
...
360. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 21:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. void doSeriesNumber(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
368. if (repeatable)
```

[Table of Contents] [Description]

Path 22:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. void doSeriesNumber(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
376. if (repeatable)
```

[Table of Contents] [Description]

Path 23:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
414. void doTextArea(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
     ...
425. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 24:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
414. void doTextArea(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
     ...
430. if (i == 0)
```

[Table of Contents] [Description]

Path 25:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
414. void doTextArea(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
     ...
437. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 26:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
414. void doTextArea(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
444. if (repeatable)
```

[Table of Contents] [Description]

Path 27:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. void doOneBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
497. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 28:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. void doOneBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
502. if (i == 0)
```

[Table of Contents] [Description]

Path 29:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. void doOneBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
509. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 30:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. void doOneBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
516. if (repeatable)
```

[Table of Contents] [Description]

Path 31:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
559. void doTwoBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
586. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 32:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
559. void doTwoBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
591. if (i == 0)
```

[Table of Contents] [Description]

Path 33:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
559. void doTwoBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
598. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 34:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
559. void doTwoBox(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
       ...
627. if (i < defaults.length)
```

[Table of Contents] [Description]

Path 35:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
679. void doQualdropValue(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
       ...
701. if (fieldCount == 0)
```

[Table of Contents] [Description]

Path 36:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
679. void doQualdropValue(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
718. if (j == 0)
```

[Table of Contents] [Description]

Path 37:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
679. void doQualdropValue(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
729. if (repeatable)
```

[Table of Contents] [Description]

Path 38:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
679. void doQualdropValue(javax.servlet.jsp.JspWriter out, Item item, //edit-
metadata.jsp
     ...
749. if (repeatable)
```

[Table of Contents] [Description]

Path 39:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
787. void doDropDown(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
     ...
806. if (repeatable)
```

[Table of Contents] [Description]

Path 40:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
787. void doDropDown(javax.servlet.jsp.JspWriter out, Item item, //edit-metadata.jsp
     ...
816. if (value.equals(defaults[j].value))
```

[Table of Contents] [Description]

Path 41:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //edit-metadata.jsp
      ...
914. if (dcQualifier != null && !dcQualifier.equals("*"))
```

[Table of Contents] [Description]

Path 42:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
73. if (last >= epeople.length) last = epeople.length - 1;
```

[Table of Contents] [Description]

Path 43:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
80. if (jumpFiveBack < 0) jumpFiveBack = 0;
```

[Table of Contents] [Description]

Path 44:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
83. if (jumpOneBack < 0) jumpOneBack = 0;
```

[Table of Contents] [Description]

Path 45:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
86. if (jumpOneForward > epeople.length) jumpOneForward = first;
```

[Table of Contents] [Description]

Path 46:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
89. if (jumpFiveForward > epeople.length) jumpFiveForward = jumpEnd;
```

[Table of Contents] [Description]

Path 47:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
93. if (sortBy == EPerson.EMAIL) sortByParam = "email";
```

[Table of Contents] [Description]

Path 48:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
94. if (sortBy == EPerson.ID) sortByParam = "id";
```

[Table of Contents] [Description]

Path 49:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
      ...
73. if (last >= groups.length) last = groups.length - 1;
```

[Table of Contents] [Description]

Path 50:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
      ...
80. if (jumpFiveBack < 0) jumpFiveBack = 0;
```

[Table of Contents] [Description]

Path 51:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
       ...
83. if (jumpOneBack < 0) jumpOneBack = 0;
```

[Table of Contents] [Description]

Path 52:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
       ...
86. if (jumpOneForward > groups.length) jumpOneForward = first;
```

[Table of Contents] [Description]

Path 53:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
       ...
89. if (jumpFiveForward > groups.length) jumpFiveForward = jumpEnd;
```

[Table of Contents] [Description]

Path 54:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
       ...
93. if (sortBy == Group.ID) sortByParam = "id";
```

[Table of Contents] [Description]

Path 55:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. public boolean equals(Object obj) //uuid.java
       ...
99. if(this == obj) // comparing to myself
```

[Table of Contents] [Description]

Path 56:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. public boolean equals(Object obj) //uuid.java
      ...
101. if(obj instanceof UUID)
```

[Table of Contents] [Description]

Path 57:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
835. private void addDCValue(Item i, String schema, Node n) throws
TransformerException //itemimport.java
      ...
839. if (value == null)
```

[Table of Contents] [Description]

Path 58:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public String createMetadata(Object nativeItem) //plugincrosswalk.java
      ...
181. if (xwalk.preferList())
```

[Table of Contents] [Description]

Path 59:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      ...
181. if (line.hasOption('w'))
```

[Table of Contents] [Description]

Path 60:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      ...
183. if (line.hasOption('e'))
```

[Table of Contents] [Description]

Path 61:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
       ...
185. if (line.hasOption('c'))
```

[Table of Contents] [Description]

Path 62:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
       ...
187. if (line.hasOption('t'))
```

[Table of Contents] [Description]

Path 63:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
       ...
189. if (line.hasOption('i'))
```

[Table of Contents] [Description]

Path 64:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
       ...
192. if (files.length > 0)
```

[Table of Contents] [Description]

Path 65:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
       ...
194. if (line.hasOption('d'))
```

[Table of Contents] [Description]

Path 66:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
202. if (pair.length == 2)
```

[Table of Contents] [Description]

Path 67:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
202. if (pair.length == 2)
```

[Table of Contents] [Description]

Path 68:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
204. else if (pair.length == 1)
```

[Table of Contents] [Description]

Path 69:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
204. else if (pair.length == 1)
```

[Table of Contents] [Description]

Path 70:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
204. else if (pair.length == 1)
```

[Table of Contents] [Description]

Path 71:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
229. if (myEPerson == null)
```

[Table of Contents] [Description]

Path 72:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
229. if (myEPerson == null)
```

[Table of Contents] [Description]

Path 73:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
241. if (sip == null)
```

[Table of Contents] [Description]

Path 74:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
241. if (sip == null)
```

[Table of Contents] [Description]

Path 75:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
256. if (dso == null)
```

[Table of Contents] [Description]

Path 76:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
261. else if (dso.getType() != Constants.COLLECTION)
```

[Table of Contents] [Description]

Path 77:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
289. if (handle == null)
```

[Table of Contents] [Description]

Path 78:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
289. if (handle == null)
```

[Table of Contents] [Description]

Path 79:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
289. if (handle == null)
```

[Table of Contents] [Description]

Path 80:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     ...
320. if (dip == null)
```

[Table of Contents] [Description]

Path 81:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      |
      ...
320. if (dip == null)
```

[Table of Contents] [Description]

Path 82:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      |
      ...
325. if (dso == null)
```

[Table of Contents] [Description]

Path 83:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public String renderVocabularyAsHTML(Document
vocabulary, //controlledvocabularytag.java
      |
      ...
233. if (vocabulary == null)
```

[Table of Contents] [Description]

Path 84:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
266. public Document filterVocabulary(Document
vocabulary, //controlledvocabularytag.java
      |
      ...
269. if (vocabulary == null)
```

[Table of Contents] [Description]

Path 85:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
      |
```

```
...
287. if (configLine == null)
```

[Table of Contents] [Description]

Path 86:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
367. public static String join(List list, String
separator) //controlledvocabularysearchservlet.java
...
374. if (iterator.hasNext())
```

[Table of Contents] [Description]

Path 87:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
57. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularyservlet.java
...
66. if (request.getParameter("ID") != null)
```

[Table of Contents] [Description]

Path 88:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
57. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularyservlet.java
...
69. if (request.getParameter("filter") != null)
```

[Table of Contents] [Description]

Path 89:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
...
85. if (host.equals("localhost") || host.equals("127.0.0.1")
```

[Table of Contents] [Description]

Path 90:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
    ...
498. if(fieldLabel !=null && fieldLabel.length()>0)
```

[Table of Contents] [Description]

Path 91:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
    ...
498. if(fieldLabel !=null && fieldLabel.length()>0)
```

[Table of Contents] [Description]

Path 92:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
    ...
504. if(isDate)
```

[Table of Contents] [Description]

Path 93:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
426. private void collectionHome(Context context, HttpServletRequest
request, //handleservlet.java
    ...
442. if (context.getCurrentUser() == null &&
```

[Table of Contents] [Description]

Path 94:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. private static Bitstream getItemBitstreamByName(Item item, String
bsName) //htmlservlet.java
    ...
```

```
128. if (bsName.equals(bitstreams[k].getName()))
```

[Table of Contents] [Description]

Path 95:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. protected void doDSGet(Context context, //ldapservlet.java
    ...
94. if (ldap_enabled)
```

[Table of Contents] [Description]

Path 96:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. protected void doDSGet(Context context, //ldapservlet.java
    ...
94. if (ldap_enabled)
```

[Table of Contents] [Description]

Path 97:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. protected void doDSGet(Context context, //ldapservlet.java
    ...
94. if (ldap_enabled)
```

[Table of Contents] [Description]

Path 98:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
     ...
212. if ((ldapEmail!=null)&&(!ldapEmail.equals(""))) eperson.setEmail
(ldapEmail);
```

[Table of Contents] [Description]

Path 99:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
```

```
...
212. if ((ldapEmail!=null)&&(!ldapEmail.equals(""))) eperson.setEmail
(ldapEmail);
```

[Table of Contents] [Description]

Path 100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
...
212. if ((ldapEmail!=null)&&(!ldapEmail.equals(""))) eperson.setEmail
(ldapEmail);
```

[Table of Contents] [Description]

Path 101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
...
214. if ((ldapGivenName!=null)&&(!ldapGivenName.equals("")))
eperson.setFirstName(ldapGivenName);
```

[Table of Contents] [Description]

Path 102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
...
214. if ((ldapGivenName!=null)&&(!ldapGivenName.equals("")))
eperson.setFirstName(ldapGivenName);
```

[Table of Contents] [Description]

Path 103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
...
215. if ((ldapSurname!=null)&&(!ldapSurname.equals(""))) eperson.setLastName
(ldapSurname);
```

[Table of Contents] [Description]

Path 104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
       215. if ((ldapSurname!=null)&&(!ldapSurname.equals(""))) eperson.setLastName
       (ldapSurname);
```

[Table of Contents] [Description]

Path 105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
       216. if ((ldapPhone!=null)&&(!ldapPhone.equals(""))) eperson.setMetadata
       ("phone", ldapPhone);
```

[Table of Contents] [Description]

Path 106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. protected void doDSPost(Context context, //ldapservlet.java
       ...
       216. if ((ldapPhone!=null)&&(!ldapPhone.equals(""))) eperson.setMetadata
       ("phone", ldapPhone);
```

[Table of Contents] [Description]

Path 107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
       ...
       300. if (att != null) ldapEmail = (String)att.get();
```

[Table of Contents] [Description]

Path 108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
       ...
       306. if (att != null) ldapGivenName = (String)att.get();
```

[Table of Contents] [Description]

Path 109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
     ...
312. if (att != null) ldapSurname = (String)att.get();
```

[Table of Contents] [Description]

Path 110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
     ...
318. if (att != null) ldapPhone = (String)att.get();
```

[Table of Contents] [Description]

Path 111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
     ...
108. else if (status == AuthenticationMethod.CERT_REQUIRED)
```

[Table of Contents] [Description]

Path 112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. protected void doDSGet(Context context, HttpServletRequest
request, //registerservlet.java
     ...
137. if (ldap_enabled) JSPManager.showJSP(request, response, "/register/new-
ldap-user.jsp");
```

[Table of Contents] [Description]

Path 113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. protected void doDSGet(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    137. if (ldap_enabled) JSPManager.showJSP(request, response, "/register/new-
    ldap-user.jsp");
```

[Table of Contents] [Description]

Path 114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    261. if (netid!=null) eperson2 = EPerson.findByNetid(context, netid);
```

[Table of Contents] [Description]

Path 115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    425. if (ldap_enabled) JSPManager.showJSP(request, response, "/register/new-
    ldap-user.jsp");
```

[Table of Contents] [Description]

Path 116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    425. if (ldap_enabled) JSPManager.showJSP(request, response, "/register/new-
    ldap-user.jsp");
```

[Table of Contents] [Description]

Path 117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    425. if (ldap_enabled) JSPManager.showJSP(request, response, "/register/new-
```

```
      ldap-user.jsp");
```

[Table of Contents] [Description]

Path 118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
      ...
465. if ((netid!=null)&&(email==null)) email = request.getParameter("email");
```

[Table of Contents] [Description]

Path 119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
      ...
482. if (email!=null) eperson = EPerson.findByEmail(context, email);
```

[Table of Contents] [Description]

Path 120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
      ...
485. if (eperson2 !=null) eperson = eperson2;
```

[Table of Contents] [Description]

Path 121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
      ...
531. if (token!=null) AccountManager.deleteToken(context, token);
```

[Table of Contents] [Description]

Path 122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
```

```
...
531. if (token!=null) AccountManager.deleteToken(context, token);
```

[Table of Contents] [Description]

Path 123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. protected void doDSGet(Context context, //grouplistservlet.java
    ...
83. if (first == -1) first = 0;
```

[Table of Contents] [Description]

Path 124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
177. public static boolean startAuthentication(Context context, //authenticate.java
     ...
232. if (count == 1)
```

[Table of Contents] [Description]

Path 125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
177. public static boolean startAuthentication(Context context, //authenticate.java
     ...
232. if (count == 1)
```

[Table of Contents] [Description]

Path 126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
177. public static boolean startAuthentication(Context context, //authenticate.java
     ...
232. if (count == 1)
```

[Table of Contents] [Description]

Path 127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. public static Context obtainContext(HttpServletRequest request) //uiutil.java
```

```
...
99. if(request.getCharacterEncoding()==null)
```

[Table of Contents] [Description]

Path 128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. public static Context obtainContext(HttpServletRequest request) //uiutil.java
...
99. if(request.getCharacterEncoding()==null)
```

[Table of Contents] [Description]

Path 129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. private static List getResults(BrowseScope scope, List results, int
max) //browse.java
...
1084. if (isAuthorsBrowse)
```

[Table of Contents] [Description]

Path 130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. private static List getResults(BrowseScope scope, List results, int
max) //browse.java
...
1086. else if (isSubjectsBrowse)
```

[Table of Contents] [Description]

Path 131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1417. private static String getTargetColumns(BrowseScope scope) //browse.java
...
1421. if (browseType == AUTHORS_BROWSE)
```

[Table of Contents] [Description]

Path 132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1417. private static String getTargetColumns(BrowseScope scope) //browse.java
      |...
      |1423. else if (browseType == SUBJECTS_BROWSE)
```

[Table of Contents] [Description]

Path 133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public static ResultsPruner getPruner(String propsFile) //resultspruner.java
      |...
      |119. if (!matcher.matches())
```

[Table of Contents] [Description]

Path 134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
190. public static BitstreamFormat findByMIMEType(Context
context, //bitstreamformat.java
      |...
      |200. if (formatRow == null)
```

[Table of Contents] [Description]

Path 135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
598. public void delete() throws SQLException,
AuthorizeException //bitstreamformat.java
      |...
      |610. if (unknown.getID() == getID())
```

[Table of Contents] [Description]

Path 136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
568. public static MetadataField find(Context context, int id) //metadatafield.java
      |...
      |575. if (!id2field.containsKey(iid))
```

[Table of Contents] [Description]

Path 137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
588. private static void initCache(Context context) throws
SQLException //metadatafield.java
     |...
     |590. if (id2field != null)
```

[Table of Contents] [Description]

Path 138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public static MetadataSchema find(Context context, int
id) //metadataschema.java
     |...
     |460. if (!id2schema.containsKey(iid))
```

[Table of Contents] [Description]

Path 139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public static MetadataSchema find(Context context, String
shortName) //metadataschema.java
     |...
     |480. if (shortName == null)
```

[Table of Contents] [Description]

Path 140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public static MetadataSchema find(Context context, String
shortName) //metadataschema.java
     |...
     |485. if (!name2schema.containsKey(shortName))
```

[Table of Contents] [Description]

Path 141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
499. private static void initCache(Context context) throws
SQLException //metadataschema.java
     |...
     |501. if (id2schema != null && name2schema != null)
```

[Table of Contents] [Description]

Path 142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public Element disseminateElement(DSpaceObject
dso) //metsdisseminationcrosswalk.java
      ...
147. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public Element disseminateElement(DSpaceObject
dso) //metsdisseminationcrosswalk.java
      ...
153. if (dip == null)
```

[Table of Contents] [Description]

Path 144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public class MODSDisseminationCrosswalk extends
SelfNamedPlugin //modsdisseminationcrosswalk.java
      ...
131. if (key.startsWith(CONFIG_PREFIX))
```

[Table of Contents] [Description]

Path 145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public class MODSDisseminationCrosswalk extends
SelfNamedPlugin //modsdisseminationcrosswalk.java
      ...
131. if (key.startsWith(CONFIG_PREFIX))
```

[Table of Contents] [Description]

Path 146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |243. if (modsMap != null)
```

[Table of Contents] [Description]

Path 147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |283. if (pair.length < 2)
```

[Table of Contents] [Description]

Path 148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |283. if (pair.length < 2)
```

[Table of Contents] [Description]

Path 149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |289. if (trip != null)
```

[Table of Contents] [Description]

Path 150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. private void initMap() //modsdisseminationcrosswalk.java
      |...
      |289. if (trip != null)
```

[Table of Contents] [Description]

Path 151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
330. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
346. if (trip == null)
```

[Table of Contents] [Description]

Path 153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
346. if (trip == null)
```

[Table of Contents] [Description]

Path 154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
353. if (addSchema)
```

[Table of Contents] [Description]

Path 155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
353. if (addSchema)
```

[Table of Contents] [Description]

Path 156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
356. if (!ni.hasNext())
```

[Table of Contents] [Description]

Path 157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
356. if (!ni.hasNext())
```

[Table of Contents] [Description]

Path 158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
364. if (what instanceof Element)
```

[Table of Contents] [Description]

Path 159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
     ...
364. if (what instanceof Element)
```

[Table of Contents] [Description]

Path 160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
```

```
addSchema) //modsdisseminationcrosswalk.java
    ...
366. else if (what instanceof Attribute)
```

[Table of Contents] [Description]

Path 161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
    ...
366. else if (what instanceof Attribute)
```

[Table of Contents] [Description]

Path 162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
    ...
368. else if (what instanceof Text)
```

[Table of Contents] [Description]

Path 163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
    ...
368. else if (what instanceof Text)
```

[Table of Contents] [Description]

Path 164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //modsdisseminationcrosswalk.java
    ...
368. else if (what instanceof Text)
```

[Table of Contents] [Description]

Path 165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. public void ingest(Context context, DSpaceObject dso, Element
root) //premiscrosswalk.java
      |...
      |104. if (!(root.getName().equals("premis")))
```

[Table of Contents] [Description]

Path 166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |113. if (dso.getType() != Constants.BITSTREAM)
```

[Table of Contents] [Description]

Path 167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |125. if (me.getName().equals("premis"))
```

[Table of Contents] [Description]

Path 168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |125. if (me.getName().equals("premis"))
```

[Table of Contents] [Description]

Path 169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |
```

```
...
133. if (on != null)
```

[Table of Contents] [Description]

Path 170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
...
148. if (bitstream.getSize() != size)
```

[Table of Contents] [Description]

Path 171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
...
169. if (md.equals(b_md))
```

[Table of Contents] [Description]

Path 172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
...
169. if (md.equals(b_md))
```

[Table of Contents] [Description]

Path 173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
...
185. if (fd != null)
```

[Table of Contents] [Description]

Path 174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |202. if (bf == null)
```

[Table of Contents] [Description]

Path 175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |204. if (bf != null)
```

[Table of Contents] [Description]

Path 176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void ingest(Context context, DSpaceObject dso, List
ml) //premiscrosswalk.java
      |...
      |204. if (bf != null)
```

[Table of Contents] [Description]

Path 177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
      |...
      |234. if (dso.getType() != Constants.BITSTREAM)
```

[Table of Contents] [Description]

Path 178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
      |...
      |261. if (bi.length > 0)
```

[Table of Contents] [Description]

Path 179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
       ...
271. if (handle != null && baseUrl != null)
```

[Table of Contents] [Description]

Path 180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
       ...
271. if (handle != null && baseUrl != null)
```

[Table of Contents] [Description]

Path 181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
       ...
271. if (handle != null && baseUrl != null)
```

[Table of Contents] [Description]

Path 182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public Element disseminateElement(DSpaceObject dso) //premiscrosswalk.java
       ...
328. if (oname == null)
```

[Table of Contents] [Description]

Path 183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. public class QDCCrosswalk extends SelfNamedPlugin //qdccrosswalk.java
       ...
179. if (key.startsWith(propname))
```

[Table of Contents] [Description]

Path 184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. public class QDCCrosswalk extends SelfNamedPlugin //qdccrosswalk.java
       ...
179. if (key.startsWith(propname))
```

[Table of Contents] [Description]

Path 185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. private String makeQualifiedTagName(Element element) //qdccrosswalk.java
       ...
197. if (ns != null)
```

[Table of Contents] [Description]

Path 186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
       ...
249. if (inited)
```

[Table of Contents] [Description]

Path 187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
       ...
254. if (myName == null)
```

[Table of Contents] [Description]

Path 188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
       ...
265. if (key.startsWith(propname))
```

[Table of Contents] [Description]

Path 189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
       |
       ...
       |
265. if (key.startsWith(propname))
```

[Table of Contents] [Description]

Path 190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
246. private void init() //qdccrosswalk.java
       |
       ...
       |
278. if (propsFilename == null)
```

[Table of Contents] [Description]

Path 191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
       |
       ...
       |
360. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
       |
       ...
       |
380. if (dc[i].schema.equals(MetadataSchema.DC_SCHEMA))
```

[Table of Contents] [Description]

Path 193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
       |
       ...
```

```
380. if (dc[i].schema.equals(MetadataSchema.DC_SCHEMA))
```

[Table of Contents] [Description]

Path 194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
...
387. if (addSchema && schemaLocation != null)
```

[Table of Contents] [Description]

Path 195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
...
387. if (addSchema && schemaLocation != null)
```

[Table of Contents] [Description]

Path 196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
...
389. if (dc[i].language != null)
```

[Table of Contents] [Description]

Path 197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. private List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //qdccrosswalk.java
...
389. if (dc[i].language != null)
```

[Table of Contents] [Description]

Path 198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public Element disseminateElement(DSpaceObject dso) //qdccrosswalk.java
      ...
403. if (schemaLocation != null)
```

[Table of Contents] [Description]

Path 199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public Element disseminateElement(DSpaceObject dso) //qdccrosswalk.java
      ...
403. if (schemaLocation != null)
```

[Table of Contents] [Description]

Path 200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
414. public void ingest(Context context, DSpaceObject dso, Element
root) //qdccrosswalk.java
      ...
423. if (!(root.getName().equals("qualifieddc")))
```

[Table of Contents] [Description]

Path 201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      ...
434. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      ...
446. if (me.getName().equals("qualifieddc"))
```

[Table of Contents] [Description]

Path 203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      |...
      |446. if (me.getName().equals("qualifieddc"))
```

[Table of Contents] [Description]

Path 204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      |...
      |455. if (lang == null)
```

[Table of Contents] [Description]

Path 205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      |...
      |458. if (qdc.length == 3)
```

[Table of Contents] [Description]

Path 206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      |...
      |458. if (qdc.length == 3)
```

[Table of Contents] [Description]

Path 207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
      |
```

```
...
460. else if (qdc.length == 2)
```

[Table of Contents] [Description]

Path 208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public void ingest(Context context, DSpaceObject dso, List
ml) //qdccrosswalk.java
...
460. else if (qdc.length == 2)
```

[Table of Contents] [Description]

Path 209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //simpledcdisseminationcrosswalk.java
...
125. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //simpledcdisseminationcrosswalk.java
...
142. if (allDC[i].element.equals("contributor")
```

[Table of Contents] [Description]

Path 211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //simpledcdisseminationcrosswalk.java
...
150. if (addSchema)
```

[Table of Contents] [Description]

Path 212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public List disseminateListInternal(DSpaceObject dso, boolean
addSchema) //simpledcdisseminationcrosswalk.java
      ...
150. if (addSchema)
```

[Table of Contents] [Description]

Path 213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
      ...
138. if (namespaces != null || schemaLocation != null)
```

[Table of Contents] [Description]

Path 214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
      ...
153. if (schemaLocation == null)
```

[Table of Contents] [Description]

Path 215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
      ...
153. if (schemaLocation == null)
```

[Table of Contents] [Description]

Path 216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
      ...
170. if (key.startsWith(nsPrefix))
```

[Table of Contents] [Description]

Path 217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. private void init() //xsltdisseminationcrosswalk.java
       ...
170. if (key.startsWith(nsPrefix))
```

[Table of Contents] [Description]

Path 218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public Element disseminateElement(DSpaceObject
dso) //xsltdisseminationcrosswalk.java
       ...
211. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public Element disseminateElement(DSpaceObject
dso) //xsltdisseminationcrosswalk.java
       ...
215. if (xform == null)
```

[Table of Contents] [Description]

Path 220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public List disseminateList(DSpaceObject dso) //xsltdisseminationcrosswalk.java
       ...
237. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public List disseminateList(DSpaceObject dso) //xsltdisseminationcrosswalk.java
       ...
241. if (xform == null)
```

[Table of Contents] [Description]

Path 222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
266. if (dc[i].qualifier != null)
```

[Table of Contents] [Description]

Path 223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
266. if (dc[i].qualifier != null)
```

[Table of Contents] [Description]

Path 224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
268. if (dc[i].language != null)
```

[Table of Contents] [Description]

Path 225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
268. if (dc[i].language != null)
```

[Table of Contents] [Description]

Path 226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
270. if (dc[i].value != null)
```

[Table of Contents] [Description]

Path 227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private Element getDim(Item item) //xsltdisseminationcrosswalk.java
       ...
270. if (dc[i].value != null)
```

[Table of Contents] [Description]

Path 228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. private void applyDim(List dimList, Item item) //xsltingestioncrosswalk.java
       ...
115. if (elt.getName().equals("field") && elt.getNamespace().equals(DIM_NS))
```

[Table of Contents] [Description]

Path 229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. private void applyDim(List dimList, Item item) //xsltingestioncrosswalk.java
       ...
115. if (elt.getName().equals("field") && elt.getNamespace().equals(DIM_NS))
```

[Table of Contents] [Description]

Path 230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. private void applyDim(List dimList, Item item) //xsltingestioncrosswalk.java
       ...
119. else if (elt.getName().equals("dim") && elt.getNamespace().equals
(DIM_NS))
```

[Table of Contents] [Description]

Path 231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. private void applyDim(List dimList, Item item) //xsltingestioncrosswalk.java
       ...
119. else if (elt.getName().equals("dim") && elt.getNamespace().equals
(DIM_NS))
```

[Table of Contents] [Description]

Path 232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public void ingest(Context context, DSpaceObject dso, List
metadata) //xsltingestioncrosswalk.java
      |...
      |151. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public void ingest(Context context, DSpaceObject dso, List
metadata) //xsltingestioncrosswalk.java
      |...
      |156. if (xform == null)
```

[Table of Contents] [Description]

Path 234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
175. public void ingest(Context context, DSpaceObject dso, Element
root) //xsltingestioncrosswalk.java
      |...
      |178. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
175. public void ingest(Context context, DSpaceObject dso, Element
root) //xsltingestioncrosswalk.java
      |...
      |183. if (xform == null)
```

[Table of Contents] [Description]

Path 236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
      ...
227. if (xform == null)
```

[Table of Contents] [Description]

Path 237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
      ...
254. if (!(o instanceof Element))
```

[Table of Contents] [Description]

Path 238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
      ...
279. if (mf == null)
```

[Table of Contents] [Description]

Path 239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
      ...
279. if (mf == null)
```

[Table of Contents] [Description]

Path 240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
      ...
254. if (auth)
```

[Table of Contents] [Description]

Path 241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
      |...
      |254. if (auth)
```

[Table of Contents] [Description]

Path 242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public void disseminate(Context context, DSpaceObject
dso, //abstractmetsdisseminator.java
      |...
      |254. if (auth)
```

[Table of Contents] [Description]

Path 243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      |370. if (xwalk == null)
```

[Table of Contents] [Description]

Path 244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      |370. if (xwalk == null)
```

[Table of Contents] [Description]

Path 245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
```

```
item, //abstractmetsdisseminator.java
      ...
      403. if (!(unauth != null && unauth.equalsIgnoreCase("skip")))
```

[Table of Contents] [Description]

Path 246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
      403. if (!(unauth != null && unauth.equalsIgnoreCase("skip")))
```

[Table of Contents] [Description]

Path 247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
      430. if (xwalk == null)
```

[Table of Contents] [Description]

Path 248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
      445. if (PackageUtils.isMetaInfoBundle(bundles[i]))
```

[Table of Contents] [Description]

Path 249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      ...
      453. if (unauth != null &&
```

[Table of Contents] [Description]

Path 250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      467. if ((bName != null) && !bName.equals(""))
```

[Table of Contents] [Description]

Path 251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      467. if ((bName != null) && !bName.equals(""))
```

[Table of Contents] [Description]

Path 252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      490. if (unauth != null && unauth.equalsIgnoreCase("skip"))
```

[Table of Contents] [Description]

Path 253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |...
      492. else if (!(unauth != null && unauth.equalsIgnoreCase("zero")))
```

[Table of Contents] [Description]

Path 254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
      |
```

```
...
506. if (bitstreams[bits].getID() == primaryBitstreamID)
```

[Table of Contents] [Description]

Path 255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
...
615. if (licenseID != null)
```

[Table of Contents] [Description]

Path 256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. private void writeManifest(Context context, Item
item, //abstractmetsdisseminator.java
...
615. if (licenseID != null)
```

[Table of Contents] [Description]

Path 257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
696. private void crosswalkToMets(DisseminationCrosswalk
xwalk, //abstractmetsdisseminator.java
...
708. if (sloc != null && sloc.length > 1 && uri.equals(sloc[0]))
```

[Table of Contents] [Description]

Path 258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
696. private void crosswalkToMets(DisseminationCrosswalk
xwalk, //abstractmetsdisseminator.java
...
708. if (sloc != null && sloc.length > 1 && uri.equals(sloc[0]))
```

[Table of Contents] [Description]

Path 259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
696. private void crosswalkToMets(DisseminationCrosswalk
xwalk, //abstractmetsdisseminator.java
      ...
      708. if (sloc != null && sloc.length > 1 && uri.equals(sloc[0]))
```

[Table of Contents] [Description]

Path 260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. public Bitstream getBitstreamForMdRef(Element
mdref) //abstractmetsingester.java
      ...
      143. if (mdBundle == null)
```

[Table of Contents] [Description]

Path 261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public InputStream getInputStream(Element mdref) //abstractmetsingester.java
      ...
      159. if (mdbs == null)
```

[Table of Contents] [Description]

Path 262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      249. if (manifest == null)
```

[Table of Contents] [Description]

Path 263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      249. if (manifest == null)
```

[Table of Contents] [Description]

Path 264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      269. if (mfileId == null)
```

[Table of Contents] [Description]

Path 265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      290. if (lastSlash >= 0 && lastSlash+1 < fname.length())
```

[Table of Contents] [Description]

Path 266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      290. if (lastSlash >= 0 && lastSlash+1 < fname.length())
```

[Table of Contents] [Description]

Path 267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      ...
      299. if (bf == null)
```

[Table of Contents] [Description]

Path 268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |309. if (bns != null && bns.length > 0)
```

[Table of Contents] [Description]

Path 269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |318. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |318. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |318. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |333. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |333. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |333. if (packageFiles.contains(path))
```

[Table of Contents] [Description]

Path 275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |343. if (mdBundle == null)
```

[Table of Contents] [Description]

Path 276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
     |...
     |352. if (packageFiles.contains(MANIFEST_FILE))
```

[Table of Contents] [Description]

Path 277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
```

```
collection, //abstractmetsingester.java
    ...
    352. if (packageFiles.contains(MANIFEST_FILE))
```

[Table of Contents] [Description]

Path 278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
    ...
    428. if (pbs == null)
```

[Table of Contents] [Description]

Path 279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
    ...
    428. if (pbs == null)
```

[Table of Contents] [Description]

Path 280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
    ...
    434. if (bn.length > 0)
```

[Table of Contents] [Description]

Path 281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
    ...
    434. if (bn.length > 0)
```

[Table of Contents] [Description]

Path 282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      |...
      |434. if (bn.length > 0)
```

[Table of Contents] [Description]

Path 283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      |...
      |474. if (!success && wi != null)
```

[Table of Contents] [Description]

Path 284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public WorkspaceItem ingest(Context context, Collection
collection, //abstractmetsingester.java
      |...
      |474. if (!success && wi != null)
```

[Table of Contents] [Description]

Path 285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. private String makeDerivedFilename(String bundleName, String
origName) //abstractmetsingester.java
      |...
      |499. if (bundleName.equals(mf.getBundleName()))
```

[Table of Contents] [Description]

Path 286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public String bundleToFileGrp(String bname) //dspacemetsdisseminator.java
      |...
```

```
127. if (bname.equals("ORIGINAL"))
```

[Table of Contents] [Description]

Path 287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. public String [] getDmdTypes(PackageParameters
params) //dspacemetsdisseminator.java
     ...
147. if (params != null)
```

[Table of Contents] [Description]

Path 288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. private boolean addDepositLicense(Context context, Item item, AmdSec
amdSec) //dspacemetsdisseminator.java
     ...
187. if (licenseBs == null)
```

[Table of Contents] [Description]

Path 289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private static Bitstream findDepositLicense(Context context, Item
item) //dspacemetsdisseminator.java
     ...
277. if (bf != null)
```

[Table of Contents] [Description]

Path 290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. void checkManifest(METSManifest manifest) //dspacemetsingester.java
     ...
88. if (profile == null)
```

[Table of Contents] [Description]

Path 291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. void checkManifest(METSManifest manifest) //dspacemetsingester.java
      ...
90. else if (!profile.startsWith(PROFILE_START))
```

[Table of Contents] [Description]

Path 292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public void chooseItemDmd(Context context, Item item, //dspacemetsingester.java
      ...
146. if (g != null && !g.equals(groupID))
```

[Table of Contents] [Description]

Path 293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public void chooseItemDmd(Context context, Item item, //dspacemetsingester.java
      ...
146. if (g != null && !g.equals(groupID))
```

[Table of Contents] [Description]

Path 294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public void chooseItemDmd(Context context, Item item, //dspacemetsingester.java
      ...
156. if (dmds.length > 0)
```

[Table of Contents] [Description]

Path 295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public void chooseItemDmd(Context context, Item item, //dspacemetsingester.java
      ...
156. if (dmds.length > 0)
```

[Table of Contents] [Description]

Path 296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
218. if (!xsd.exists())
```

[Table of Contents] [Description]

Path 297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
220. if (!xsd.exists())
```

[Table of Contents] [Description]

Path 298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
220. if (!xsd.exists())
```

[Table of Contents] [Description]

Path 299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
227. if (result.length() > 0)
```

[Table of Contents] [Description]

Path 300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public class METSManifest //metsmanifest.java
      ...
227. if (result.length() > 0)
```

[Table of Contents] [Description]

Path 301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public static METSManifest create(InputStream is, boolean
validate) //metsmanifest.java
     |...
     |274. if (validate)
```

[Table of Contents] [Description]

Path 302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public static METSManifest create(InputStream is, boolean
validate) //metsmanifest.java
     |...
     |274. if (validate)
```

[Table of Contents] [Description]

Path 303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public static METSManifest create(InputStream is, boolean
validate) //metsmanifest.java
     |...
     |280. if (localSchemas.length() > 0)
```

[Table of Contents] [Description]

Path 304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public static METSManifest create(InputStream is, boolean
validate) //metsmanifest.java
     |...
     |280. if (localSchemas.length() > 0)
```

[Table of Contents] [Description]

Path 305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
322. public List getContentFiles() //metsmanifest.java
     |...
     |325. if (contentFiles != null)
```

[Table of Contents] [Description]

Path 306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
322. public List getContentFiles() //metsmanifest.java
       ...
329. if (fileSec == null)
```

[Table of Contents] [Description]

Path 307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
386. public Element getOriginalFile(Element file) //metsmanifest.java
       ...
389. if (groupID == null || groupID.equals(""))
```

[Table of Contents] [Description]

Path 308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
415. private static String normalizeBundleName(String in) //metsmanifest.java
       ...
417. if (in.equals("CONTENT"))
```

[Table of Contents] [Description]

Path 309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
415. private static String normalizeBundleName(String in) //metsmanifest.java
       ...
419. else if (in.equals("MANIFESTMD"))
```

[Table of Contents] [Description]

Path 310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. public static String getBundleName(Element file) //metsmanifest.java
       ...
434. if (fgUse == null)
```

[Table of Contents] [Description]

Path 311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
448. public static String getFileName(Element file) //metsmanifest.java
       ...
458. if (file.getChild("FContent", metsNS) == null)
```

[Table of Contents] [Description]

Path 312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
448. public static String getFileName(Element file) //metsmanifest.java
       ...
464. else if (file.getName().equals("mdRef"))
```

[Table of Contents] [Description]

Path 313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
448. public static String getFileName(Element file) //metsmanifest.java
       ...
472. if (result == null)
```

[Table of Contents] [Description]

Path 314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. public Element getPrimaryBitstream() //metsmanifest.java
       ...
491. if (fptr == null)
```

[Table of Contents] [Description]

Path 315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. public Element getPrimaryBitstream() //metsmanifest.java
       ...
494. if (id == null)
```

[Table of Contents] [Description]

Path 316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
486. public Element getPrimaryBitstream() //metsmanifest.java
       ...
497. if (result == null)
```

[Table of Contents] [Description]

Path 317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
505. public String getMdType(Element mdSec) //metsmanifest.java
       ...
509. if (md == null)
```

[Table of Contents] [Description]

Path 318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
505. public String getMdType(Element mdSec) //metsmanifest.java
       ...
511. if (md == null)
```

[Table of Contents] [Description]

Path 319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
505. public String getMdType(Element mdSec) //metsmanifest.java
       ...
514. if (result != null && result.equals("OTHER"))
```

[Table of Contents] [Description]

Path 320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
505. public String getMdType(Element mdSec) //metsmanifest.java
       ...
516. if (result == null)
```

[Table of Contents] [Description]

Path 321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
525. public String getMdContentMimeType(Element mdSec) //metsmanifest.java
      ...
532. if (mimeType == null && mdWrap.getChild("xmlData", metsNS) != null)
```

[Table of Contents] [Description]

Path 322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
525. public String getMdContentMimeType(Element mdSec) //metsmanifest.java
      ...
537. if (mdRef != null)
```

[Table of Contents] [Description]

Path 323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
549. public List getMdContentAsXml(Element mdSec, Mdref
callback) //metsmanifest.java
      ...
562. if (bin == null)
```

[Table of Contents] [Description]

Path 324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
622. public InputStream getMdContentAsStream(Element mdSec, Mdref
callback) //metsmanifest.java
      ...
633. if (bin == null)
```

[Table of Contents] [Description]

Path 325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
682. private Element getFirstDiv() //metsmanifest.java
      ...
686. if (sm == null)
```

[Table of Contents] [Description]

Path 326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
682. private Element getFirstDiv() //metsmanifest.java
       ...
690. if (result == null)
```

[Table of Contents] [Description]

Path 327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
699. private Element getElementByXPath(String path, boolean
nullOk) //metsmanifest.java
       ...
708. if (result == null && nullOk)
```

[Table of Contents] [Description]

Path 328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
699. private Element getElementByXPath(String path, boolean
nullOk) //metsmanifest.java
       ...
710. else if (result instanceof Element)
```

[Table of Contents] [Description]

Path 329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
727. private IngestionCrosswalk getCrosswalk(String type) //metsmanifest.java
       ...
730. if (xwalkName == null)
```

[Table of Contents] [Description]

Path 330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
743. public Element[] getItemDmds() //metsmanifest.java
       ...
```

```
749. if (dmds == null)
```

[Table of Contents] [Description]

Path 331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
764. public Element[] getItemRightsMD() //metsmanifest.java
        ...
781. if (rmds.size() > 0)
```

[Table of Contents] [Description]

Path 332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
764. public Element[] getItemRightsMD() //metsmanifest.java
        ...
781. if (rmds.size() > 0)
```

[Table of Contents] [Description]

Path 333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
790. public void crosswalkItem(Context context, Item item, Element dmd, Mdref
callback) //metsmanifest.java
        ...
797. if (xwalk == null)
```

[Table of Contents] [Description]

Path 334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
811. public void crosswalkBitstream(Context context, Bitstream
bitstream, //metsmanifest.java
        ...
817. if (file == null)
```

[Table of Contents] [Description]

Path 335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
811. public void crosswalkBitstream(Context context, Bitstream
bitstream, //metsmanifest.java
      ...
843. if (xwalk == null)
```

[Table of Contents] [Description]

Path 336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void log(Logger log) //packageexception.java
      ...
98. if (cause.getCause() != null)
```

[Table of Contents] [Description]

Path 337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      ...
94. if (v.length == 0)
```

[Table of Contents] [Description]

Path 338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      ...
94. if (v.length == 0)
```

[Table of Contents] [Description]

Path 339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      ...
96. else if (v.length == 1)
```

[Table of Contents] [Description]

Path 340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      |...
      96. else if (v.length == 1)
```

[Table of Contents] [Description]

Path 341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      |...
      103. if (i > 0)
```

[Table of Contents] [Description]

Path 342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public static PackageParameters create(ServletRequest
request) //packageparameters.java
      |...
      103. if (i > 0)
```

[Table of Contents] [Description]

Path 343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. public Object addProperty(String key, String value) //packageparameters.java
      |...
      126. if (oldVal == null)
```

[Table of Contents] [Description]

Path 344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. public Object addProperty(String key, String value) //packageparameters.java
      |...
      126. if (oldVal == null)
```

[Table of Contents] [Description]

Path 345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. public Object addProperty(String key, String value) //packageparameters.java
      ...
126. if (oldVal == null)
```

[Table of Contents] [Description]

Path 346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. public String[] getProperties(String key) //packageparameters.java
      ...
142. if (val == null)
```

[Table of Contents] [Description]

Path 347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public boolean getBooleanProperty(String key, boolean
defaultAnswer) //packageparameters.java
      ...
159. if (stringValue == null)
```

[Table of Contents] [Description]

Path 348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
78. public static void checkMetadata(Item item) //packageutils.java
      ...
82. if (t == null || t.length == 0)
```

[Table of Contents] [Description]

Path 349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. public static void addDepositLicense(Context context, String
license, //packageutils.java
      |
```

```
...
101. if (license == null)
```

[Table of Contents] [Description]

Path 350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. public static void addDepositLicense(Context context, String
license, //packageutils.java
...
108. if (bf == null)
```

[Table of Contents] [Description]

Path 351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public static Bitstream getBitstreamByName(Item item, String bsName, String
bnName) //packageutils.java
...
141. if (bnName == null)
```

[Table of Contents] [Description]

Path 352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public static Bitstream getBitstreamByName(Item item, String bsName, String
bnName) //packageutils.java
...
151. if (bsName.equals(bitstreams[k].getName()))
```

[Table of Contents] [Description]

Path 353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
167. public static Bitstream getBitstreamByFormat(Item item, //packageutils.java
...
173. if (bnName == null)
```

[Table of Contents] [Description]

Path 354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
167. public static Bitstream getBitstreamByFormat(Item item, //packageutils.java
     ...
183. if (bitstreams[k].getFormat().getID() == fid)
```

[Table of Contents] [Description]

Path 355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     ...
200. if (bis != null)
```

[Table of Contents] [Description]

Path 356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     ...
200. if (bis != null)
```

[Table of Contents] [Description]

Path 357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     ...
202. if (cos != null)
```

[Table of Contents] [Description]

Path 358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     ...
202. if (cos != null)
```

[Table of Contents] [Description]

Path 359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     |...
     211. if (original != null && bs != null)
```

[Table of Contents] [Description]

Path 360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     |...
     211. if (original != null && bs != null)
```

[Table of Contents] [Description]

Path 361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     |...
     213. if (wi != null)
```

[Table of Contents] [Description]

Path 362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
     |...
     213. if (wi != null)
```

[Table of Contents] [Description]

Path 363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public void disseminate(Context context, DSpaceObject dso, //pdfpackager.java
     |
```

```
...
241. if (dso.getType() != Constants.ITEM)
```

[Table of Contents] [Description]

Path 364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public void disseminate(Context context, DSpaceObject dso, //pdfpackager.java
...
249. if (pdff == null)
```

[Table of Contents] [Description]

Path 365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public void disseminate(Context context, DSpaceObject dso, //pdfpackager.java
...
252. if (pkgBs == null)
```

[Table of Contents] [Description]

Path 366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
...
281. if(cos.getEncryptionDictionary() != null)
```

[Table of Contents] [Description]

Path 367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
...
306. if (title == null)
```

[Table of Contents] [Description]

Path 368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
     |...
     |317. if ((value = docinfo.getCreator()) != null)
```

[Table of Contents] [Description]

Path 369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
     |...
     |317. if ((value = docinfo.getCreator()) != null)
```

[Table of Contents] [Description]

Path 370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
     |...
     |320. if ((value = docinfo.getProducer()) != null)
```

[Table of Contents] [Description]

Path 371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
     |...
     |320. if ((value = docinfo.getProducer()) != null)
```

[Table of Contents] [Description]

Path 372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
     |...
     |323. if ((value = docinfo.getSubject()) != null)
```

[Table of Contents] [Description]

Path 373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
      |...
      323. if ((value = docinfo.getSubject()) != null)
```

[Table of Contents] [Description]

Path 374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
      |...
      325. if ((value = docinfo.getKeywords()) != null)
```

[Table of Contents] [Description]

Path 375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
      |...
      325. if ((value = docinfo.getKeywords()) != null)
```

[Table of Contents] [Description]

Path 376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
      |...
      331. if ((calValue = docinfo.getCreationDate()) == null)
```

[Table of Contents] [Description]

Path 377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
```

```
metadata) //pdfpackager.java
    ...
333. if (calValue != null)
```

[Table of Contents] [Description]

Path 378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
    ...
333. if (calValue != null)
```

[Table of Contents] [Description]

Path 379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
    ...
340. if (cos != null)
```

[Table of Contents] [Description]

Path 380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. private void crosswalkPDF(Context context, Item item, InputStream
metadata) //pdfpackager.java
    ...
340. if (cos != null)
```

[Table of Contents] [Description]

Path 381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
215. public static Enumeration propertyNames() //configurationmanager.java
    ...
217. if (properties == null)
```

[Table of Contents] [Description]

Path 382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
215. public static Enumeration propertyNames() //configurationmanager.java
      ...
      217. if (properties == null)
```

[Table of Contents] [Description]

Path 383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
      ...
      506. if (value != null)
```

[Table of Contents] [Description]

Path 384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
      ...
      506. if (value != null)
```

[Table of Contents] [Description]

Path 385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      ...
      583. if (level > RECURSION_LIMIT)
```

[Table of Contents] [Description]

Path 386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      ...
      594. if (end < 0)
```

[Table of Contents] [Description]

Path 387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      |...
      |597. if (result == null)
```

[Table of Contents] [Description]

Path 388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      |...
      |597. if (result == null)
```

[Table of Contents] [Description]

Path 389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      |...
      |622. if (result != null && from < value.length())
```

[Table of Contents] [Description]

Path 390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private static String interpolate(String key, int
level) //configurationmanager.java
      |...
      |622. if (result != null && from < value.length())
```

[Table of Contents] [Description]

Path 391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public static Object getSinglePlugin(Class interfaceClass) //pluginmanager.java
      |...
```

```
157. if (classname != null)
```

[Table of Contents] [Description]

Path 392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
287. if (prevClassName != null)
```

[Table of Contents] [Description]

Path 393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
293. if (prevClassName != null)
```

[Table of Contents] [Description]

Path 394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
311. if (names == null || names.length == 0)
```

[Table of Contents] [Description]

Path 395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
     ...
311. if (names == null || names.length == 0)
```

[Table of Contents] [Description]

Path 396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
```

```
...
327. if (found == 0)
```

[Table of Contents] [Description]

Path 397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. private static void configureNamedPlugin(String iname) //pluginmanager.java
...
327. if (found == 0)
```

[Table of Contents] [Description]

Path 398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
333. private static int installNamedConfigs(String iname, String classname, String
names[]) //pluginmanager.java
...
340. if (namedPluginClasses.containsKey(key))
```

[Table of Contents] [Description]

Path 399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
333. private static int installNamedConfigs(String iname, String classname, String
names[]) //pluginmanager.java
...
340. if (namedPluginClasses.containsKey(key))
```

[Table of Contents] [Description]

Path 400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
333. private static int installNamedConfigs(String iname, String classname, String
names[]) //pluginmanager.java
...
340. if (namedPluginClasses.containsKey(key))
```

[Table of Contents] [Description]

Path 401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      ...
      370. if (cname == null)
```

[Table of Contents] [Description]

Path 402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      ...
      370. if (cname == null)
```

[Table of Contents] [Description]

Path 403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      ...
      385. if (cached instanceof SelfNamedPlugin)
```

[Table of Contents] [Description]

Path 404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      ...
      385. if (cached instanceof SelfNamedPlugin)
```

[Table of Contents] [Description]

Path 405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      ...
      397. if (result instanceof SelfNamedPlugin)
```

[Table of Contents] [Description]

Path 406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
361. public static Object getNamedPlugin(Class intfc, String
name) //pluginmanager.java
      |...
      |397. if (result instanceof SelfNamedPlugin)
```

[Table of Contents] [Description]

Path 407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. public static String[] getAllPluginNames(Class intfc) //pluginmanager.java
      |...
      |447. if (key.startsWith(prefix))
```

[Table of Contents] [Description]

Path 408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. public static String[] getAllPluginNames(Class intfc) //pluginmanager.java
      |...
      |447. if (key.startsWith(prefix))
```

[Table of Contents] [Description]

Path 409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. public static String[] getAllPluginNames(Class intfc) //pluginmanager.java
      |...
      |450. if (result.size() == 0)
```

[Table of Contents] [Description]

Path 410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. public static String[] getAllPluginNames(Class intfc) //pluginmanager.java
      |...
```

```
450. if (result.size() == 0)
```

[Table of Contents] [Description]

Path 411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private static void forgetInstance(Object plugin, Map
cacheMap) //pluginmanager.java
      ...
483. if (val == plugin)
```

[Table of Contents] [Description]

Path 412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private static void forgetInstance(Object plugin, Map
cacheMap) //pluginmanager.java
      ...
483. if (val == plugin)
```

[Table of Contents] [Description]

Path 413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
509. private static boolean checkSelfNamed(String iname) //pluginmanager.java
      ...
513. if (!checkSelfNamed(Class.forName(iname)))
```

[Table of Contents] [Description]

Path 414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
509. private static boolean checkSelfNamed(String iname) //pluginmanager.java
      ...
513. if (!checkSelfNamed(Class.forName(iname)))
```

[Table of Contents] [Description]

Path 415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private static boolean checkSelfNamed(Class cls) //pluginmanager.java
       ...
527. if (sup == null)
```

[Table of Contents] [Description]

Path 416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private static boolean checkSelfNamed(Class cls) //pluginmanager.java
       ...
529. else if (sup.equals(SelfNamedPlugin.class))
```

[Table of Contents] [Description]

Path 417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
590. if (line.startsWith("!") || line.startsWith("#"))
```

[Table of Contents] [Description]

Path 418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
600. if (keyMap.containsKey(key))
```

[Table of Contents] [Description]

Path 419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
       ...
600. if (keyMap.containsKey(key))
```

[Table of Contents] [Description]

Path 420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      600. if (keyMap.containsKey(key))
```

[Table of Contents] [Description]

Path 421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      605. if (key.startsWith(SINGLE_PREFIX))
```

[Table of Contents] [Description]

Path 422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      605. if (key.startsWith(SINGLE_PREFIX))
```

[Table of Contents] [Description]

Path 423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      607. else if (key.startsWith(SEQUENCE_PREFIX))
```

[Table of Contents] [Description]

Path 424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      607. else if (key.startsWith(SEQUENCE_PREFIX))
```

[Table of Contents] [Description]

Path 425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      ...
609. else if (key.startsWith(NAMED_PREFIX))
```

[Table of Contents] [Description]

Path 426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      ...
609. else if (key.startsWith(NAMED_PREFIX))
```

[Table of Contents] [Description]

Path 427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      ...
611. else if (key.startsWith(SELFNAMED_PREFIX))
```

[Table of Contents] [Description]

Path 428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      ...
611. else if (key.startsWith(SELFNAMED_PREFIX))
```

[Table of Contents] [Description]

Path 429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      ...
613. else if (key.startsWith(REUSABLE_PREFIX))
```

[Table of Contents] [Description]

Path 430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
613. else if (key.startsWith(REUSABLE_PREFIX))
```

[Table of Contents] [Description]

Path 431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
613. else if (key.startsWith(REUSABLE_PREFIX))
```

[Table of Contents] [Description]

Path 432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
632. if (!keyMap.containsKey(nk))
```

[Table of Contents] [Description]

Path 433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
632. if (!keyMap.containsKey(nk))
```

[Table of Contents] [Description]

Path 434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
640. if (!pn.contains(key))
```

[Table of Contents] [Description]

Path 435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      640. if (!pn.contains(key))
```

[Table of Contents] [Description]

Path 436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      669. if (val == null)
```

[Table of Contents] [Description]

Path 437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      669. if (val == null)
```

[Table of Contents] [Description]

Path 438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      674. if (checkClassname(val, "implementation class"))
```

[Table of Contents] [Description]

Path 439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
      |...
      674. if (checkClassname(val, "implementation class"))
```

[Table of Contents] [Description]

Path 440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
685. if (val == null)
```

[Table of Contents] [Description]

Path 441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
685. if (val == null)
```

[Table of Contents] [Description]

Path 442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
704. if (val == null)
```

[Table of Contents] [Description]

Path 443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
704. if (val == null)
```

[Table of Contents] [Description]

Path 444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
728. if (val == null)
```

[Table of Contents] [Description]

Path 445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
728. if (val == null)
```

[Table of Contents] [Description]

Path 446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
739. if (checkClassname(classname, "implementation class"))
```

[Table of Contents] [Description]

Path 447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
739. if (checkClassname(classname, "implementation class"))
```

[Table of Contents] [Description]

Path 448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
750. if (!(allImpls.containsKey(rk)))
```

[Table of Contents] [Description]

Path 449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public static void checkConfiguration() //pluginmanager.java
        ...
750. if (!(allImpls.containsKey(rk)))
```

[Table of Contents] [Description]

Path 450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
277. public static String addEntities(String value) //utils.java
       |
       ...
       |
279. if (value==null || value.length() == 0)
```

[Table of Contents] [Description]

Path 451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
       |
       ...
       |
206. if (ret == AuthenticationMethod.SUCCESS)
```

[Table of Contents] [Description]

Path 452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
       |
       ...
       |
208. if (ret < bestRet)
```

[Table of Contents] [Description]

Path 453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public static int[] getSpecialGroups(Context
context, //authenticationmanager.java
       |
       ...
       |
319. if (totalLen == 0)
```

[Table of Contents] [Description]

Path 454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public static int[] getSpecialGroups(Context
context, //authenticationmanager.java
       |
       ...
       |
321. else if (gll.size() == 1)
```

[Table of Contents] [Description]

Path 455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public static EPerson findByNetid(Context context, String netid) //eperson.java
       ...
183. if (netid == null)
```

[Table of Contents] [Description]

Path 456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public static Set allMemberGroupIDs(Context c, EPerson e) //group.java
       ...
469. if (i.hasNext())
```

[Table of Contents] [Description]

Path 457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
541. public static Set allMemberIDs(Context c, Group g) //group.java
       ...
577. if (i.hasNext())
```

[Table of Contents] [Description]

Path 458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
541. public static Set allMemberIDs(Context c, Group g) //group.java
       ...
586. if (i.hasNext())
```

[Table of Contents] [Description]

Path 459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1161. private Set getChildren(Map parents, Integer parent) //group.java
       ...
1166. if (!parents.containsKey(parent))
```

[Table of Contents] [Description]

Path 460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
167. if (eperson.getRequireCertificate())
```

[Table of Contents] [Description]

Path 461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
169. else if (!eperson.canLogIn())
```

[Table of Contents] [Description]

Path 462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
219. if ((ldap.ldapEmail!=null)&&(!ldap.ldapEmail.equals("")))
     eperson.setEmail(ldap.ldapEmail);
```

[Table of Contents] [Description]

Path 463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
219. if ((ldap.ldapEmail!=null)&&(!ldap.ldapEmail.equals("")))
     eperson.setEmail(ldap.ldapEmail);
```

[Table of Contents] [Description]

Path 464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
219. if ((ldap.ldapEmail!=null)&&(!ldap.ldapEmail.equals("")))
     eperson.setEmail(ldap.ldapEmail);
```

[Table of Contents] [Description]

Path 465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      221. if ((ldap.ldapGivenName!=null)&&(!ldap.ldapGivenName.equals("")))
      eperson.setFirstName(ldap.ldapGivenName);
```

[Table of Contents] [Description]

Path 466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      221. if ((ldap.ldapGivenName!=null)&&(!ldap.ldapGivenName.equals("")))
      eperson.setFirstName(ldap.ldapGivenName);
```

[Table of Contents] [Description]

Path 467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      222. if ((ldap.ldapSurname!=null)&&(!ldap.ldapSurname.equals("")))
      eperson.setLastName(ldap.ldapSurname);
```

[Table of Contents] [Description]

Path 468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      222. if ((ldap.ldapSurname!=null)&&(!ldap.ldapSurname.equals("")))
      eperson.setLastName(ldap.ldapSurname);
```

[Table of Contents] [Description]

Path 469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
```

```
...
223. if ((ldap.ldapPhone!=null)&&(!ldap.ldapPhone.equals("")))
eperson.setMetadata("phone", ldap.ldapPhone);
```

[Table of Contents] [Description]

Path 470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
...
223. if ((ldap.ldapPhone!=null)&&(!ldap.ldapPhone.equals("")))
eperson.setMetadata("phone", ldap.ldapPhone);
```

[Table of Contents] [Description]

Path 471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
...
337. if (att != null) ldapEmail = (String)att.get();
```

[Table of Contents] [Description]

Path 472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
...
343. if (att != null) ldapGivenName = (String)att.get();
```

[Table of Contents] [Description]

Path 473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
...
349. if (att != null) ldapSurname = (String)att.get();
```

[Table of Contents] [Description]

Path 474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
     ...
355. if (att != null) ldapPhone = (String)att.get();
```

[Table of Contents] [Description]

Path 475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
     ...
379. if (ctx != null)
```

[Table of Contents] [Description]

Path 476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
     ...
379. if (ctx != null)
```

[Table of Contents] [Description]

Path 477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
     ...
189. if (eperson == null)
```

[Table of Contents] [Description]

Path 478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
     ...
193. else if (!eperson.canLogIn())
```

[Table of Contents] [Description]

Path 479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public class X509Authentication //x509authentication.java
      ...
135. if (caCertPath == null)
```

[Table of Contents] [Description]

Path 480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public class X509Authentication //x509authentication.java
      ...
141. if (keystorePassword == null)
```

[Table of Contents] [Description]

Path 481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public class X509Authentication //x509authentication.java
      ...
169. if (cert != null)
```

[Table of Contents] [Description]

Path 482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. private static String getEmail(X509Certificate
certificate) //x509authentication.java
      ...
202. if (principal == null)
```

[Table of Contents] [Description]

Path 483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. private static String getEmail(X509Certificate
certificate) //x509authentication.java
      ...
206. if (dn == null)
```

[Table of Contents] [Description]

Path 484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. private static String getEmail(X509Certificate
certificate) //x509authentication.java
      |...
      220. if (token.length() <= len)
```

[Table of Contents] [Description]

Path 485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
      |...
      244. if (certificate == null)
```

[Table of Contents] [Description]

Path 486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
380. public int authenticate(Context context, //x509authentication.java
      |...
      389. if (request != null)
```

[Table of Contents] [Description]

Path 487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
380. public int authenticate(Context context, //x509authentication.java
      |...
      393. if ((certs == null) || (certs.length == 0))
```

[Table of Contents] [Description]

Path 488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
380. public int authenticate(Context context, //x509authentication.java
      |...
```

```
410. if (email != null)
```

[Table of Contents] [Description]

Path 489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
     ...
282. if(writer != null)
```

[Table of Contents] [Description]

Path 490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
     ...
282. if(writer != null)
```

[Table of Contents] [Description]

Path 491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
     ...
147. if (numFieldStr == null) numFieldStr ="3";
```

[Table of Contents] [Description]

Path 492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
     ...
170. if (tmp_field == null)
```

[Table of Contents] [Description]

Path 493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
     ...
170. if (tmp_field == null)
```

[Table of Contents] [Description]

Path 494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
     ...
170. if (tmp_field == null)
```

[Table of Contents] [Description]

Path 495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public HashMap buildQueryHash(HttpServletRequest request) //queryargs.java
     ...
245. if (numFieldStr == null) numFieldStr = "3";
```

[Table of Contents] [Description]

Path 496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
767. private static GeneralFile getFile(TableRow bitstream) throws
IOException //bitstreamstoragemanager.java
     ...
808. if (sInternalId.indexOf(File.separator) != -1)
```

[Table of Contents] [Description]

Path 497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
634. public static TableRow findByUnique(Context context, String
table, //databasemanager.java
     ...
639. if ( ! DB_SAFE_NAME.matcher(ctable).matches())
```

[Table of Contents] [Description]

Path 498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
634. public static TableRow findByUnique(Context context, String
table, //databasemanager.java
      ...
      642. if ( ! DB_SAFE_NAME.matcher(column).matches())
```

[Table of Contents] [Description]

Path 499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
689. public static int deleteByValue(Context context, String
table, //databasemanager.java
      ...
      694. if ( ! DB_SAFE_NAME.matcher(ctable).matches())
```

[Table of Contents] [Description]

Path 500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
689. public static int deleteByValue(Context context, String
table, //databasemanager.java
      ...
      697. if ( ! DB_SAFE_NAME.matcher(column).matches())
```

[Table of Contents] [Description]

Path 501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
      1250. if (longValue <= (long)Integer.MAX_VALUE)
```

[Table of Contents] [Description]

Path 502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      ...
      1250. if (longValue <= (long)Integer.MAX_VALUE)
```

[Table of Contents] [Description]

Path 503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
      |...
      1250. if (longValue <= (long)Integer.MAX_VALUE)
```

[Table of Contents] [Description]

Path 504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
      |...
      1438. if ("oracle".equals(dbName))
```

[Table of Contents] [Description]

Path 505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
      |...
      1438. if ("oracle".equals(dbName))
```

[Table of Contents] [Description]

Path 506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
      |...
      1438. if ("oracle".equals(dbName))
```

[Table of Contents] [Description]

Path 507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
       ...
218. if (results != null)
```

[Table of Contents] [Description]

Path 508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
       ...
218. if (results != null)
```

[Table of Contents] [Description]

Path 509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
       ...
228. if (statemt != null)
```

[Table of Contents] [Description]

Path 510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
       ...
228. if (statemt != null)
```

[Table of Contents] [Description]

Improper Error Handling

CWE ID	390
Description	Sometimes an error is detected, and bad or no action is taken.
Likelihood of Exploit	Medium
Potential Mitigations	Implementation: Properly handle each exception. This is the recommended solution. Ensure that all exceptions are handled in such a way that you can be sure of the state of your system at any given moment. Subject the software to extensive testing to discover some of the possible instances of where/how errors or return values are not handled. Consider testing techniques such as ad hoc,

equivalence partitioning, robustness and fault tolerance, mutation, and fuzzing.

Demonstrative Examples

C Example:

```
foo=malloc(sizeof(char); //the next line checks to see if malloc failed
if (foo==0) {
  //We do nothing so we just ignore the error.
}
```

C++ Example:

```
while (DoSomething()) {
  try {
    /* perform main loop here */
  }
  catch (Exception e) {
    /* do nothing, but catch so it'll compile... */
  }
}
```

Java Example:

```
while (DoSomething()) {
  try {
    /* perform main loop here */
  }
  catch (Exception e) {
    /* do nothing, but catch so it'll compile... */
  }
}
```

Context Notes

If a function returns an error, it is important to either fix the problem and try again, alert the user that an error has happened and let the program continue, or alert the user and close and cleanup the program.

Node Relationships ChildOf - Error Conditions, Return Values, Status Codes (389)

Source Taxonomies CLASP - Improper error handling

Applicable Platforms All

Path 1:

Query Name - **Empty_Catch**

Severity - **Information**

```
143. public int[] getSpecialGroups(Context context, HttpServletRequest
request) //mitspecialgroup.java
     |
     |...
     |172. catch (java.sql.SQLException e)
```

[Table of Contents] [Description]

Path 2:

Query Name - **Empty_Catch**

Severity - **Information**

```
945. private void processContentsFile(Context c, Item i, String
path, //itemimport.java
      |...
      |997. catch (NumberFormatException e)
```

[Table of Contents] [Description]

Path 3:

Query Name - **Empty_Catch**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
      |...
      |496. catch(java.util.MissingResourceException e) {}
```

[Table of Contents] [Description]

Path 4:

Query Name - **Empty_Catch**

Severity - **Information**

```
411. private com.sun.syndication.feed.rss.Item itemFromDSpaceItem(Context
context, //feedservlet.java
      |...
      |527. catch (ArrayIndexOutOfBoundsException e)
```

[Table of Contents] [Description]

Path 5:

Query Name - **Empty_Catch**

Severity - **Information**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
      |...
      |127. catch (NumberFormatException nfe)
```

[Table of Contents] [Description]

Path 6:

Query Name - **Empty_Catch**

Severity - **Information**

```
145. protected void doDSGet(Context context, HttpServletRequest
```

```
request, //htmlservlet.java
    ...
216. catch (NumberFormatException nfe)
```

[Table of Contents] [Description]

Path 7:

Query Name - **Empty_Catch**

Severity - **Information**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
    ...
104. catch (NumberFormatException nfe)
```

[Table of Contents] [Description]

Path 8:

Query Name - **Empty_Catch**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
    ...
167. catch (ParseException e)
```

[Table of Contents] [Description]

Path 9:

Query Name - **Empty_Catch**

Severity - **Information**

```
121. private void showStatistics(Context context, //statisticsservlet.java
    ...
219. catch (ParseException e)
```

[Table of Contents] [Description]

Path 10:

Query Name - **Empty_Catch**

Severity - **Information**

```
436. public void setShortDescription(String s) //bitstreamformat.java
    ...
443. } catch (IllegalStateException e) {
```

[Table of Contents] [Description]

Path 11:

Query Name - **Empty_Catch**

Severity

- **Information**

```
322. public Namespace[] getNamespaces() //qdccrosswalk.java
       ...
328. catch (Exception e)
```

[Table of Contents] [Description]

Path 12:

Query Name - **Empty_Catch**

Severity - **Information**

```
334. public String getSchemaLocation() //qdccrosswalk.java
       ...
340. catch (Exception e)
```

[Table of Contents] [Description]

Path 13:

Query Name - **Empty_Catch**

Severity - **Information**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
       ...
205. catch (IOException ie)
```

[Table of Contents] [Description]

Path 14:

Query Name - **Empty_Catch**

Severity - **Information**

```
537. private static void checkNames(String iname) //pluginmanager.java
       ...
543. catch (ClassNotFoundException ce)
```

[Table of Contents] [Description]

Path 15:

Query Name - **Empty_Catch**

Severity - **Information**

```
121. public static byte[] getMD5Bytes(byte[] data) //utils.java
       ...
129. catch (NoSuchAlgorithmException nsae)
```

[Table of Contents] [Description]

Path 16:

Query Name - **Empty_Catch**

Severity - **Information**

```
142. public int authenticate(Context context, //ldapauthentication.java
     ...
157. catch (SQLException e)
```

[Table of Contents] [Description]

Path 17:

Query Name - **Empty_Catch**

Severity - **Information**

```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
     ...
382. catch (NamingException e)
```

[Table of Contents] [Description]

Path 18:

Query Name - **Empty_Catch**

Severity - **Information**

```
168. public int authenticate(Context context, //passwordauthentication.java
     ...
183. catch (AuthorizeException e)
```

[Table of Contents] [Description]

Path 19:

Query Name - **Empty_Catch**

Severity - **Information**

```
322. public static void sendEmail(Context context, EPerson eperson, //subscribe.java
     ...
413. catch (ParseException pe)
```

[Table of Contents] [Description]

Path 20:

Query Name - **Empty_Catch**

Severity - **Information**

```
241. private static boolean isValid(Context context, //x509authentication.java
     ...
291. catch (CertificateException ce)
```

[Table of Contents] [Description]

Path 21:

Query Name - **Empty_Catch**

Severity - **Information**

```
238. public byte[][] getRawHandleValues(byte[] theHandle, int[]
indexList, //handleplugin.java
      |...
      |317. catch (SQLException sqle)
```

[Table of Contents] [Description]

Path 22:

Query Name - **Empty_Catch**

Severity - **Information**

```
374. public Enumeration getHandlesForNA(byte[] theNAHandle) //handleplugin.java
      |...
      |420. catch (SQLException sqle)
```

[Table of Contents] [Description]

Path 23:

Query Name - **Empty_Catch**

Severity - **Information**

```
469. private static String serialize(Context context, Object
obj) //historymanager.java
      |...
      |491. catch (IOException ioe)
```

[Table of Contents] [Description]

Path 24:

Query Name - **Empty_Catch**

Severity - **Information**

```
504. public static int updateQuery(Context context, String query, Object[]
parameters) //databasemanager.java
      |...
      |529. catch (SQLException sqle)
```

[Table of Contents] [Description]

Path 25:

Query Name - **Empty_Catch**

Severity - **Information**

```
1028. public static void loadSql(String sql) throws
SQLException //databasemanager.java
      |...
```

```
1034. catch (IOException ioe)
```

[Table of Contents] [Description]

Path 26:

Query Name - **Empty_Catch**

Severity - **Information**

```
1217. static TableRow process(ResultSet results, String
table) //databasemanager.java
...
1284. catch (UnsupportedEncodingException e)
```

[Table of Contents] [Description]

Path 27:

Query Name - **Empty_Catch**

Severity - **Information**

```
1397. private static int execute(Connection connection, String sql, List
columns, //databasemanager.java
...
1502. catch (SQLException sqle)
```

[Table of Contents] [Description]

Path 28:

Query Name - **Empty_Catch**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
...
221. catch (SQLException sqle)
```

[Table of Contents] [Description]

Path 29:

Query Name - **Empty_Catch**

Severity - **Information**

```
213. public void close() //tablerowiterator.java
...
232. catch (SQLException sqle)
```

[Table of Contents] [Description]

Failure to Account for Default Case in Switch

CWE ID	478
Description	The failure to account for the default case in switch

	statements may lead to complex logical errors and may aid in other, unexpected security-related conditions.
Common Consequences	Undefined: Depending on the logical circumstances involved, any consequences may result: e.g., issues of confidentiality, authentication, authorization, availability, integrity, accountability, or non-repudiation.
Potential Mitigations	Implementation: Ensure that there are no unaccounted for cases, when adjusting flow or values based on the value of a given variable. In switch statements, this can be accomplished through the use of the default label.

Demonstrative Examples	In general, a safe switch statement has this form:
	
	switch (value) {
	  case 'A':
	  printf("A!\n");
	  break;
	  case 'B':
	  printf("B!\n");
	  break;
	  default:
	  printf("Neither A nor B\n");
	}

	This is because the assumption cannot be made that all possible cases are accounted for. A good practice is to reserve the default case for error handling.
Context Notes	This flaw represents a common problem in software development, in which not all possible values for a variable are considered or handled by a given process. Because of this, further decisions are made based on poor information, and cascading failure results. This cascading failure may result in any number of security issues, and constitutes a significant failure in the system. In the case of switch style statements, the very simple act of creating a default case can mitigate this situation, if done correctly. Often however, the default cause is used simply to represent an assumed option, as opposed to working as a sanity check. This is poor practice and in some cases is as bad as omitting a default case entirely.
Node Relationships	ChildOf - Code Quality (398)
Source Taxonomies	CLASP - Failure to account for default case in switch
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **No_Default_Case**

Severity - **Information**

```
1. <%-- //wizard-permissions.jsp
     |
     ...
     73. switch (perm)
```

[Table of Contents] [Description]

Path 2:

Query Name - **No_Default_Case**

Severity - **Information**

```
1. <%-- //main.jsp
     |
     ...
     155. switch (owned[i].getState())
```

[Table of Contents] [Description]

Path 3:

Query Name - **No_Default_Case**

Severity - **Information**

```
1. <%-- //main.jsp
     |
     ...
     221. switch (pooled[i].getState())
```

[Table of Contents] [Description]

Path 4:

Query Name - **No_Default_Case**

Severity - **Information**

```
1. <%-- //review.jsp
     |
     ...
     295. switch (format.getSupportLevel())
```

[Table of Contents] [Description]

Path 5:

Query Name - **No_Default_Case**

Severity - **Information**

```
96. public static void main(String[] argv) throws
Exception //mediafiltermanager.java
     |
     ...
     196. switch (dso.getType())
```

[Table of Contents] [Description]

Path 6:

Query Name - **No_Default_Case**

Severity - **Information**

```
149. private void doSearch(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
     ...
247. switch (myType.intValue())
```

[Table of Contents] [Description]

Path 7:

Query Name - **No_Default_Case**

Severity - **Information**

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
     ...
218. switch (myType.intValue())
```

[Table of Contents] [Description]

Path 8:

Query Name - **No_Default_Case**

Severity - **Information**

```
318. private void processPermissions(Context context, //collectionwizardservlet.java
     ...
352. switch (permission)
```

[Table of Contents] [Description]

Path 9:

Query Name - **No_Default_Case**

Severity - **Information**

```
588. private void showNextPage(Context context, HttpServletRequest
request, //collectionwizardservlet.java
     ...
604. switch (stage)
```

[Table of Contents] [Description]

Path 10:

Query Name - **No_Default_Case**

Severity - **Information**

```
101. private void init() //handledispatcher.java
     ...
133. switch (dsoType)
```

[Table of Contents] [Description]

Path 11:

Query Name - **No_Default_Case**

Severity - **Information**

```
121. public DCDate(String fromDC) //dcdate.java
      ...
135. switch (fromDC.length())
```

[Table of Contents] [Description]

Path 12:

Query Name - **No_Default_Case**

Severity - **Information**

```
288. public static void claim(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      ...
293. switch (taskstate)
```

[Table of Contents] [Description]

Path 13:

Query Name - **No_Default_Case**

Severity - **Information**

```
341. public static void advance(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      ...
346. switch (taskstate)
```

[Table of Contents] [Description]

Path 14:

Query Name - **No_Default_Case**

Severity - **Information**

```
395. public static void unclaim(Context c, WorkflowItem wi, EPerson
e) //workflowmanager.java
      ...
400. switch (taskstate)
```

[Table of Contents] [Description]

Path 15:

Query Name - **No_Default_Case**

Severity - **Information**

```
469. private static boolean doState(Context c, WorkflowItem wi, int
newstate, //workflowmanager.java
```

```
...
479. switch (newstate)
```

[Table of Contents] [Description]

Path 16:

Query Name - **No_Default_Case**

Severity - **Information**

```
826. private static void notifyGroupOfTask(Context c, WorkflowItem
wi, //workflowmanager.java
...
854. switch (wi.getState())
```

[Table of Contents] [Description]

Erroneous String Compare

CWE ID	597
Description	Strings should be compared with the equals() method, not == or !=

Demonstrative Examples	The following branch will never be taken.
	
	if (args[0] == STRING_CONSTANT) {
	  logger.info("miracle");
	}

Context Notes	Using == or != to compare two strings for equality actually compares two objects for equality, not their values. Chances are good that the two references will never be equal.
Node Relationships	ChildOf - Incorrect Object Comparison: Syntactic (595) ChildOf - Incorrect Object Comparison: Syntactic (595) ChildOf - String Errors (133)

Path 1:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
1. <%-- //progressbar.jsp
...
130. if (step != SubmitServlet.SUBMISSION_COMPLETE)
```

[Table of Contents] [Description]

Path 2:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
1. <%-- //progressbar.jsp
...
```

```
169. if ( step == lastMDStep )
```

[Table of Contents] [Description]

Path 3:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
320. private boolean sanityCheck(HttpServletRequest
request, //metadatafieldregistryservlet.java
...
342. if (qualifier == "")
```

[Table of Contents] [Description]

Path 4:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
125. public void process() //checkercommand.java
...
151. || (info.getChecksumCheckResult() !=
ChecksumCheckResults.CHECKSUM_MATCH))
```

[Table of Contents] [Description]

Magic Numbers

CWE ID	10017
Description	Using literals in the code makes the code less readable and maintainable.
Likelihood of Exploit	Low
Common Consequences	Code become less readable and maintainable.
Potential Mitigations	Define constants with meaningful names and use them instead.
Applicable Platforms	All

Path 1:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1. <%-- //footer-default.jsp
...
54. int overallColSpan = 3;
```

[Table of Contents] [Description]

Path 2:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1. <%-- //eperson-list.jsp
      ...
63. int PAGESIZE = 50;
```

[Table of Contents] [Description]

Path 3:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1. <%-- //group-select-list.jsp
      ...
63. int PAGESIZE = 50;
```

[Table of Contents] [Description]

Path 4:

Query Name - **Magic_Numbers**

Severity - **Information**

```
71. public class Supervisor //supervisor.java
      ...
83. public static int POLICY_OBSERVER = 2;
```

[Table of Contents] [Description]

Path 5:

Query Name - **Magic_Numbers**

Severity - **Information**

```
88. public class HistoryManager //historymanager.java
      ...
121. private static int digitsPerLevel = 2;
```

[Table of Contents] [Description]

Path 6:

Query Name - **Magic_Numbers**

Severity - **Information**

```
88. public class HistoryManager //historymanager.java
      ...
123. private static int directoryLevels = 3;
```

[Table of Contents] [Description]

Path 7:

Query Name - **Magic_Numbers**

Severity - **Information**

```
56. public class QueryArgs //queryargs.java
      |
      ...
      65. private int pageSize = 10;
```

[Table of Contents] [Description]

Confusing naming

CWE ID	10032
Description	Naming a method and field with the same name, may be confusing to developers
Likelihood of Exploit	Low
Common Consequences	Difficulty to maintain the code
Potential Mitigations	Renaming methods and fields
Applicable Platforms	All

Path 1:

Query Name - **Confusing_Naming**

Severity - **Information**

```
63. public class HTMLReport extends Report //htmlreport.java
      |
      ...
      75. private String mainTitle = null;
```

[Table of Contents] [Description]

Path 2:

Query Name - **Confusing_Naming**

Severity - **Information**

```
57. abstract class Report //report.java
      |
      ...
      67. private String mainTitle = null;
```

[Table of Contents] [Description]

Path 3:

Query Name - **Confusing_Naming**

Severity - **Information**

```
55. public class HandleDispatcher implements
BitstreamDispatcher //handledispatcher.java
      |
      ...
      65. Boolean init = Boolean.FALSE;
```

[Table of Contents] [Description]

Path 4:

Query Name - **Confusing_Naming**

Severity - **Information**

```
106. public class XSLTDisseminationCrosswalk //xsltdisseminationcrosswalk.java
       ...
132. private boolean preferList = false;
```

[Table of Contents] [Description]

Path 5:

Query Name - **Confusing_Naming**

Severity - **Information**

```
1809. class ColumnInfo //databasemanager.java
       ...
1818. private boolean isPrimaryKey;
```

[Table of Contents] [Description]

Path 6:

Query Name - **Confusing_Naming**

Severity - **Information**

```
54. public class TableRowIterator //tablerowiterator.java
       ...
74. private boolean hasNext = true;
```

[Table of Contents] [Description]

Leftover Debug Code

CWE ID	489
Description	Debug code can create unintended entry points in an application.
Common Consequences	The severity of the exposed debug application will depend on the particular instance. At the least, it will give an attacker sensitive information about the settings and mechanics of web applications on the server. At worst, as is often the case, the debug application will allow an attacker complete control over the web application and server, as well as confidential information that either of these access.
Demonstrative Examples	Debug code can be used to bypass authentication. For example, suppose an application has a login script that receives a username and a password. Assume also that a third, optional, parameter, called "debug", is interpreted by the script as requesting a switch to debug mode, and that when this parameter is given the username and password are not checked. In such a case, it is very simple to bypass the authentication process if the special behavior of the application regarding the debug parameter

is known. In a case where the form is:

```
<FORM ACTION="/authenticate_login.cgi">
                    <INPUT TYPE=TEXT name=username>
                    <INPUT TYPE=PASSWORD
name=password>
                    <INPUT TYPE=SUBMIT>
                    </FORM>
```

Then a conforming link will look like:

```
http://TARGET/authenticate_login.cgi?
username=...&password=...
```

An attacker can change this to:

```
http://TARGET/authenticate_login.cgi?
username=&password=&debug=1
```

Which will grant the attacker access to the site, bypassing the authentication process.

Context Notes

A common development practice is to add "back door" code specifically designed for debugging or testing purposes that is not intended to be shipped or deployed with the application. In web-based applications, debug code is used to test and modify web application properties, configuration information, and functions. If a debug application is left on a production server, an attacker may be able to use it to perform these tasks. When this sort of debug code is left in the application, the application is open to unintended modes of interaction. These back door entry points create security risks because they are not considered during design or testing and fall outside of the expected operating conditions of the application.

While it is possible to leave debug code in an application in any language, in J2EE a main method may be a good indicator that debug code has been left in the application, although there may not be any direct security impact.

Node Relationships

ChildOf - Encapsulation (485)
ChildOf - Weaknesses Examined by SAMATE (630)

Source Taxonomies

7 Pernicious Kingdoms - Leftover Debug Code

Applicable Platforms

All

Path 1:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //communityfiliator.java
      '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
68. public static void main(String[] argv) //createadministrator.java
      '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
89. public static void main(String[] argv) throws Exception //registryloader.java
      '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      '
```

[Table of Contents] [Description]

Path 5:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //upgrade11to12.java
      '
```

[Table of Contents] [Description]

Path 6:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
      '
```

[Table of Contents] [Description]

Path 7:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
94. public static void main(String[] argv) throws Exception //itemexport.java
      '
```

[Table of Contents] [Description]

Path 8:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
139. public static void main(String[] argv) throws Exception //itemimport.java
      '
```

[Table of Contents] [Description]

Path 9:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
96. public static void main(String[] argv) throws
Exception //mediafiltermanager.java
      '
```

[Table of Contents] [Description]

Path 10:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
114. public static void main(String[] args) throws Exception //metsexport.java
      '
```

[Table of Contents] [Description]

Path 11:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      '
```

[Table of Contents] [Description]

Path 12:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
262. public static void main(String [] argv) //loganalyser.java
     '
```

[Table of Contents] [Description]

Path 13:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
192. public static void main(String [] argv) //reportgenerator.java
     '
```

[Table of Contents] [Description]

Path 14:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
67. public static void main(String[] argv) throws
Exception //fixdefaultpolicies.java
     '
```

[Table of Contents] [Description]

Path 15:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
65. public static void main(String[] argv) throws Exception //policyset.java
     '
```

[Table of Contents] [Description]

Path 16:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
65. public static void main(String[] argv) //initializebrowse.java
     '
```

[Table of Contents] [Description]

Path 17:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
167. public static void main(String[] args) //dailyreportemailer.java
     '
```

[Table of Contents] [Description]

Path 18:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
     '
```

[Table of Contents] [Description]

Path 19:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
781. public static void main(String[] argv) //configurationmanager.java
     '
```

[Table of Contents] [Description]

Path 20:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
760. public static void main(String[] argv) throws Exception //pluginmanager.java
     '
```

[Table of Contents] [Description]

Path 21:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
439. public static void main(String[] argv) //subscribe.java
     '
```

[Table of Contents] [Description]

Path 22:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
1115. public static void main(String[] argv) //historymanager.java
      '
```

[Table of Contents] [Description]

Path 23:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
     '
```

[Table of Contents] [Description]

Path 24:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
390. public static void main(String[] args) //dsquery.java
     '
```

[Table of Contents] [Description]

Path 25:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
68. public static void main(String[] argv) //cleanup.java
     '
```

[Table of Contents] [Description]

Path 26:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
58. public static void main(String[] argv) //initializedatabase.java
     '
```

[Table of Contents] [Description]

Missing Catch Block

CWE ID 600

Description If a Servlet fails to catch all exceptions, it may reveal debugging information that will help an adversary form a plan of attack.

Demonstrative Examples In the following method a DNS lookup failure will cause the Servlet to throw an exception.

```
protected void doPost (HttpServletRequest req,
HttpServletResponse res)
throws IOException {
  String ip = req.getRemoteAddr();
  InetAddress addr = InetAddress.getByName(ip);
  ...
  out.println("hello " + addr.getHostName());
}
```

Context Notes When a Servlet throws an exception, the default error response the Servlet container sends back to the user typically includes debugging information. This information is of great value to an attacker. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

Node Relationships ChildOf - Error Handling (388)
ResultsIn - Error Message Information Leaks (209)
Peer - Improper Error Handling (390)

Path 1:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
96. protected void doGet(HttpServletRequest request, //dspaceservlet.java
      |
```

[Table of Contents] [Description]

Path 2:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
102. protected void doPost(HttpServletRequest request, //dspaceservlet.java
      |
```

[Table of Contents] [Description]

Path 3:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
69. protected void doGet(HttpServletRequest request, //internalerrorservlet.java
      |
```

[Table of Contents] [Description]

Path 4:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
91. protected void doPost(HttpServletRequest request, //internalerrorservlet.java
      |81. + ":internal_error:" + logInfo;
```

[Table of Contents] [Description]

Missing XML Validation

CWE ID	112
Description	Failure to enable validation when parsing XML gives an attacker the opportunity to supply malicious input.
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Always validate XML input against a known XML Schema or DTD. Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Context Notes	Most successful attacks begin with a violation of the programmer's assumptions. By accepting an XML document without validating it against a DTD or XML schema, the programmer leaves a door open for attackers to provide unexpected, unreasonable, or malicious input. It is not possible for an XML parser to validate all aspects of a document's content; a parser cannot understand the complete semantics of the data. However, a parser can do a complete and thorough job of checking the document's structure and therefore guarantee to the code that processes the document that the content is well-formed.
Node Relationships	ChildOf - Insufficient Input Validation (20)
Source Taxonomies	7 Pernicious Kingdoms - Missing XML Validation
Applicable Platforms	All

Path 1:

Query Name - **Missing_XML_Validation**

Severity - **Information**

```
142. private void buildInputs(String fileName) //dcinputsreader.java
       ...
153. DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Missing_XML_Validation**

Severity - **Information**

```
112. public static Document transformDocument(Document xmlDocument, //xmlutil.java
       ...
134. DocumentBuilderFactory dFactory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

No static final logger

CWE ID	10034
Description	A dedicated static logging class should be used, rather than using a non static one, to ease the monitoring

	process
Likelihood of Exploit	Low
Common Consequences	Difficulty to maintain the application and to follow log files
Potential Mitigations	Using dedicated static logging class
Applicable Platforms	All

Path 1:

Query Name - **Not_Static_Final_Logger**

Severity - **Information**

```
270. public class SpeakerToLDAP //ldapauthentication.java
       ...
272. private Logger log = null;
```

[Table of Contents] [Description]

Overly-Broad Catch Block

CWE ID	396
Description	Catching overly broad exceptions promotes complex error handling code that is more likely to contain security vulnerabilities.
Demonstrative Examples	The following code excerpt handles three types of exceptions in an identical fashion.

```
try { doExchange(); }
  catch (IOException e) {
    logger.error("doExchange failed", e);
  }
  catch (InvocationTargetException e) {
    logger.error("doExchange failed", e);
  }
  catch (SQLException e) {
    logger.error("doExchange failed", e);
}
```

At first blush, it may seem preferable to deal with these exceptions in a single catch block, as follows: try { doExchange(); } catch (Exception e) { logger.error ("doExchange failed", e); } However, if doExchange() is modified to throw a new type of exception that should be handled in some different kind of way, the broad catch block will prevent the compiler from pointing out the situation. Further, the new catch block will now also handle exceptions derived from RuntimeException such as ClassCastException, and NullPointerException, which is not the programmer's intent.

Context Notes	Multiple catch blocks can get ugly and repetitive, but "condensing" catch blocks by catching a high-level class like Exception can obscure exceptions that deserve special

	treatment or that should not be caught at this point in the program. Catching an overly broad exception essentially defeats the purpose of Java's typed exceptions, and can become particularly dangerous if the program grows and begins to throw new types of exceptions. The new exception types will not receive any attention.
Node Relationships	ChildOf - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	7 Pernicious Kingdoms - Overly-Broad Catch Block
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
68. public static void main(String[] argv) //createadministrator.java
      ...
177. catch (Exception e)
```

[Table of Contents] [Description]

Path 2:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
116. catch (Exception e)
```

[Table of Contents] [Description]

Path 3:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
       ...
228. catch (Exception e)
```

[Table of Contents] [Description]

Path 4:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
139. public static void main(String[] argv) throws Exception //itemimport.java
```

```
...
429. catch (Exception e)
```

[Table of Contents] [Description]

Path 5:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
897. private String processHandleFile(Context c, Item i, String
path, //itemimport.java
...
916. catch (Exception e)
```

[Table of Contents] [Description]

Path 6:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
312. public static boolean filterBitstream(Context c, Item
myItem, //mediafiltermanager.java
...
334. catch (Exception e)
```

[Table of Contents] [Description]

Path 7:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
97. public InputStream getDestinationStream(InputStream source) //pdffilter.java
...
118. catch(Exception e)
```

[Table of Contents] [Description]

Path 8:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
...
242. catch (Exception ex)
```

[Table of Contents] [Description]

Path 9:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public String createMetadata(Object nativeItem) //metscrosswalk.java
      ...
97. catch (Exception e)
```

[Table of Contents] [Description]

Path 10:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
174. public String createMetadata(Object nativeItem) //plugincrosswalk.java
      ...
186. catch (Exception e)
```

[Table of Contents] [Description]

Path 11:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
      ...
304. catch (Exception e)
```

[Table of Contents] [Description]

Path 12:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
815. public static String getItemInfo(Context context, String
handle) //reportgenerator.java
      ...
825. catch (Exception e)
```

[Table of Contents] [Description]

Path 13:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
81. public int doStartTag() throws JspException //controlledvocabularytag.java
      ...
134. catch (Exception e)
```

[Table of Contents] [Description]

Path 14:

Query Name - **Overly_Broad_Catch**

Severity

- **Information**

```
229. public String renderVocabularyAsHTML(Document
vocabulary, //controlledvocabularytag.java
      ...
      247. catch (Exception e)
```

[Table of Contents] [Description]

Path 15:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
266. public Document filterVocabulary(Document
vocabulary, //controlledvocabularytag.java
      ...
      280. catch (Exception e)
```

[Table of Contents] [Description]

Path 16:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
297. private static Hashtable loadControlledVocabularies(String
directory) //controlledvocabularytag.java
      ...
      324. catch (Exception e)
```

[Table of Contents] [Description]

Path 17:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
92. public FileUploadRequest(HttpServletRequest req) throws
IOException //fileuploadrequest.java
      ...
      131. catch (Exception e)
```

[Table of Contents] [Description]

Path 18:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
88. public static Context obtainContext(HttpServletRequest request) //uiutil.java
      ...
      102. catch(Exception e)
```

[Table of Contents] [Description]

Path 19:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
      ...
409. catch (Exception e)
```

[Table of Contents] [Description]

Path 20:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
428. public static int[] getIntParameters(HttpServletRequest request, //uiutil.java
      ...
446. catch (Exception e)
```

[Table of Contents] [Description]

Path 21:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
562. catch (Exception e)
```

[Table of Contents] [Description]

Path 22:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1887. public static void updateIndexData(BrowseScope key) //browse.java
      ...
1918. catch (Exception e)
```

[Table of Contents] [Description]

Path 23:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
167. public static void main(String[] args) //dailyreportemailer.java
      ...
355. catch (Exception e)
```

[Table of Contents] [Description]

Path 24:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
167. public static void main(String[] args) //dailyreportemailer.java
       ...
367. catch (Exception e)
```

[Table of Contents] [Description]

Path 25:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
322. public Namespace[] getNamespaces() //qdccrosswalk.java
       ...
328. catch (Exception e)
```

[Table of Contents] [Description]

Path 26:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
334. public String getSchemaLocation() //qdccrosswalk.java
       ...
340. catch (Exception e)
```

[Table of Contents] [Description]

Path 27:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
119. public void loadData() //group.java
       ...
184. catch (Exception e)
```

[Table of Contents] [Description]

Path 28:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
439. public static void main(String[] argv) //subscribe.java
       ...
449. catch( Exception e )
```

[Table of Contents] [Description]

Path 29:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
151. public static void saveHistory(Context context, Object obj, int
flag, //historymanager.java
      |...
      |158. catch (Exception e)
```

[Table of Contents] [Description]

Path 30:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1031. private static String getTool() //historymanager.java
      |...
      |1055. catch (Exception e)
```

[Table of Contents] [Description]

Path 31:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1115. public static void main(String[] argv) //historymanager.java
      |...
      |1172. catch (Exception e)
```

[Table of Contents] [Description]

Path 32:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
336. private static Bitstream getBitstream(Item item, String
bitstream_name) //creativecommons.java
      |...
      |355. catch (Exception exc)
```

[Table of Contents] [Description]

Path 33:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
386. private static byte[] fetchURL(String url_string) //creativecommons.java
      |
```

```
...
412. catch (Exception exc)
```

[Table of Contents] [Description]

Path 34:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
...
276. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 35:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
296. public static void unIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
303. catch(Exception exception)
```

[Table of Contents] [Description]

Path 36:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
362. catch(Exception exception)
```

[Table of Contents] [Description]

Path 37:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
416. public static void main(String[] args) throws SQLException,
IOException //dsindexer.java
...
476. catch( Exception e)
```

[Table of Contents] [Description]

Path 38:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
547. public static void updateIndex(Context context, boolean force) //dsindexer.java
      |...
      577. catch(Exception e)
```

[Table of Contents] [Description]

Path 39:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
633. static Analyzer getAnalyzer() throws IllegalStateException //dsindexer.java
      |...
      651. catch (Exception e)
```

[Table of Contents] [Description]

Path 40:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
668. private static void emailException(Exception exception) //dsindexer.java
      |...
      696. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 41:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      |...
      1059. catch (Exception e)
```

[Table of Contents] [Description]

Path 42:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      |...
      1068. catch(Exception e)
```

[Table of Contents] [Description]

Path 43:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
357. public static void doCMDLineQuery(String query) //dsquery.java
       ...
384. catch (Exception e)
```

[Table of Contents] [Description]

Path 44:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
68. public static void main(String[] argv) //cleanup.java
       ...
113. catch (Exception e)
```

[Table of Contents] [Description]

Path 45:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
58. public static void main(String[] argv) //initializedatabase.java
       ...
74. catch (Exception e)
```

[Table of Contents] [Description]

Path 46:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
915. private static void notifyOfReject(Context c, WorkflowItem wi, EPerson
e, //workflowmanager.java
       ...
940. catch (Exception ex)
```

[Table of Contents] [Description]

Overly-Broad Throws Declaration

CWE ID	397
Description	Throwing overly broad exceptions promotes complex error handling code that is more likely to contain security vulnerabilities.
Demonstrative Examples	The following method throws three types of exceptions.

```
public void doExchange() throws IOException,
InvocationTargetException, SQLException {
  ...
}
```

Context Notes

While it might seem tidier to write public void doExchange() throws Exception { ... } doing so hampers the caller's ability to understand and handle the exceptions that occur. Further, if a later revision of doExchange() introduces a new type of exception that should be treated differently than previous exceptions, there is no easy way to enforce this requirement. Declaring a method to throw Exception or Throwable makes it difficult for callers to do good error handling and error recovery. Java's exception mechanism is set up to make it easy for callers to anticipate what can go wrong and write code to handle each specific exceptional circumstance. Declaring that a method throws a generic form of exception defeats this system.

Node Relationships ChildOf - Error Conditions, Return Values, Status Codes (389)

Source Taxonomies 7 Pernicious Kingdoms - Overly-Broad Throws Declaration

Applicable Platforms C

C++

Java

.NET

Path 1:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //communityfiliator.java
      '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
89. public static void main(String[] argv) throws Exception //registryloader.java
      '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //upgrade11to12.java
```

```
    '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
94. public static void main(String[] argv) throws Exception //itemexport.java
    '
```

[Table of Contents] [Description]

Path 5:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
262. private static void exportItem(Context c, ItemIterator i, //itemexport.java
    '
```

[Table of Contents] [Description]

Path 6:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
303. private static void exportItem(Context c, Item myItem, String
destDirName, //itemexport.java
    '
```

[Table of Contents] [Description]

Path 7:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
350. private static void writeMetadata(Context c, Item i, File
destDir) //itemexport.java
    '
```

[Table of Contents] [Description]

Path 8:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
371. private static void writeMetadata(Context c, String schema, Item i, File
destDir) //itemexport.java
    '
```

[Table of Contents] [Description]

Path 9:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
430. private static void writeHandle(Context c, Item i, File
destDir) //itemexport.java
     |
```

[Table of Contents] [Description]

Path 10:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
465. private static void writeBitstreams(Context c, Item i, File
destDir) //itemexport.java
     |
```

[Table of Contents] [Description]

Path 11:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
139. public static void main(String[] argv) throws Exception //itemimport.java
     |
```

[Table of Contents] [Description]

Path 12:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
455. private void addItems(Context c, Collection[] mycollections, //itemimport.java
     |
```

[Table of Contents] [Description]

Path 13:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
512. private void replaceItems(Context c, Collection[]
mycollections, //itemimport.java
     |
```

[Table of Contents] [Description]

Path 14:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
581. private void deleteItems(Context c, String mapFile) throws
Exception //itemimport.java
     |
```

[Table of Contents] [Description]

Path 15:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
616. private Item addItem(Context c, Collection[] mycollections, String
path, //itemimport.java
     |
```

[Table of Contents] [Description]

Path 16:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
696. private void deleteItem(Context c, Item myitem) throws
Exception //itemimport.java
     |
```

[Table of Contents] [Description]

Path 17:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
711. private void deleteItem(Context c, String myhandle) throws
Exception //itemimport.java
     |
```

[Table of Contents] [Description]

Path 18:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
731. private Map readMapFile(String filename) throws Exception //itemimport.java
     |
```

[Table of Contents] [Description]

Path 19:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
103. public InputStream getDestinationStream(InputStream
source) //brandedpreviewjpegfilter.java
     '
```

[Table of Contents] [Description]

Path 20:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
93. public InputStream getDestinationStream(InputStream source) //htmlfilter.java
     '
```

[Table of Contents] [Description]

Path 21:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
95. public InputStream getDestinationStream(InputStream source) //jpegfilter.java
     '
```

[Table of Contents] [Description]

Path 22:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
91. public abstract InputStream getDestinationStream(InputStream
source) //mediafilter.java
     '
```

[Table of Contents] [Description]

Path 23:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
112. public boolean processBitstream(Context c, Item item, Bitstream
source) //mediafilter.java
     '
```

[Table of Contents] [Description]

Path 24:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
96. public static void main(String[] argv) throws
Exception //mediafiltermanager.java
```

[Table of Contents] [Description]

Path 25:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
229. public static void applyFiltersAllItems(Context c) throws
Exception //mediafiltermanager.java
```

[Table of Contents] [Description]

Path 26:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
238. public static void applyFiltersCommunity(Context c, Community
community) //mediafiltermanager.java
```

[Table of Contents] [Description]

Path 27:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
254. public static void applyFiltersCollection(Context c, Collection
collection) //mediafiltermanager.java
```

[Table of Contents] [Description]

Path 28:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
264. public static void applyFiltersItem(Context c, Item item) throws
Exception //mediafiltermanager.java
```

[Table of Contents] [Description]

Path 29:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
284. public static boolean filterItem(Context c, Item myItem) throws
Exception //mediafiltermanager.java
     '
```

[Table of Contents] [Description]

Path 30:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
312. public static boolean filterBitstream(Context c, Item
myItem, //mediafiltermanager.java
     '
```

[Table of Contents] [Description]

Path 31:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
97. public InputStream getDestinationStream(InputStream source) //pdffilter.java
     '
```

[Table of Contents] [Description]

Path 32:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
91. public InputStream getDestinationStream(InputStream source) //wordfilter.java
     '
```

[Table of Contents] [Description]

Path 33:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
114. public static void main(String[] args) throws Exception //metsexport.java
     '
```

[Table of Contents] [Description]

Path 34:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
     '
```

[Table of Contents] [Description]

Path 35:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
262. public static void main(String [] argv) //loganalyser.java
     '
```

[Table of Contents] [Description]

Path 36:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
192. public static void main(String [] argv) //reportgenerator.java
     '
```

[Table of Contents] [Description]

Path 37:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
     '
```

[Table of Contents] [Description]

Path 38:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
93. public static Document transformDocument(Document xmlDocument, //xmlutil.java
     '
```

[Table of Contents] [Description]

Path 39:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
112. public static Document transformDocument(Document xmlDocument, //xmlutil.java
     '
```

[Table of Contents] [Description]

Path 40:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
161. public static String transformDocumentAsString(Document
xmlDocument, //xmlutil.java
     '
```

[Table of Contents] [Description]

Path 41:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
201. public static String transformDocumentAsString(Document
xmlDocument, //xmlutil.java
     '
```

[Table of Contents] [Description]

Path 42:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
67. public static void main(String[] argv) throws
Exception //fixdefaultpolicies.java
     '
```

[Table of Contents] [Description]

Path 43:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
65. public static void main(String[] argv) throws Exception //policyset.java
     '
```

[Table of Contents] [Description]

Path 44:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
     '
```

[Table of Contents] [Description]

Path 45:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
760. public static void main(String[] argv) throws Exception //pluginmanager.java
     '
```

[Table of Contents] [Description]

Path 46:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
97. public void init(StreamTable st) throws Exception //handleplugin.java
     '
```

[Table of Contents] [Description]

	Use of system output stream
CWE ID	10033
Description	A dedicated logging class should be used, rather than using default output stream, to ease the monitoring process
Likelihood of Exploit	Low
Common Consequences	Difficulty to maintain the application and to follow log files
Potential Mitigations	Using dedicated logging class
Applicable Platforms	All

Path 1:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    |...
    |89. out.println( "<li class=\"communityLink\">" );
```

[Table of Contents] [Description]

Path 2:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    |...
    |90. out.println( "<strong><a href=\"" + request.getContextPath() + "/handle/"
    + c.getHandle() + "\">" + c.getMetadata("name") + "</a></strong>");
```

[Table of Contents] [Description]

Path 3:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
96. out.println("<ul>");
```

[Table of Contents] [Description]

Path 4:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
99. out.println("<li class=\"collectionListItem\">");
```

[Table of Contents] [Description]

Path 5:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
100. out.println("<a href=\"" + request.getContextPath() + "/handle/" + cols
[j].getHandle() + "\">" + cols[j].getMetadata("name") +"</a>");
```

[Table of Contents] [Description]

Path 6:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
103. out.println(" [" + cols[j].countItems() + "]");
```

[Table of Contents] [Description]

Path 7:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
    ...
106. out.println("</li>");
```

[Table of Contents] [Description]

Path 8:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      ...
108. out.println("</ul>");
```

[Table of Contents] [Description]

Path 9:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      ...
115. out.println("<ul>");
```

[Table of Contents] [Description]

Path 10:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      ...
120. out.println("</ul>");
```

[Table of Contents] [Description]

Path 11:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
      ...
122. out.println("<br />");
```

[Table of Contents] [Description]

Path 12:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
87. void showCommunity(Community c) throws IOException, SQLException //community-
list.jsp
```

```
...
123. out.println("</li>");
```

[Table of Contents] [Description]

Path 13:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //communityfiliator.java
...
94. .println("\nestablish a relationship: CommunityFiliator -s -p parentID -c
childID");
```

[Table of Contents] [Description]

Path 14:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public static void main(String[] argv) //createadministrator.java
...
84. System.out.println("Creating an initial administrator account");
```

[Table of Contents] [Description]

Path 15:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public static void main(String[] argv) //createadministrator.java
...
96. System.out.print("E-mail address: ");
```

[Table of Contents] [Description]

Path 16:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public static void main(String[] argv) //createadministrator.java
...
179. System.err.println("Exception occurred:" + e);
```

[Table of Contents] [Description]

Path 17:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
89. public static void main(String[] argv) throws Exception //registryloader.java
      ...
115. System.err.println(usage);
```

[Table of Contents] [Description]

Path 18:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
112. System.out.println("Last modified dates set");
```

[Table of Contents] [Description]

Path 19:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
118. System.err.println("Exception occurred:" + e);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
67. public static void main(String[] argv) throws Exception //upgrade11to12.java
      ...
79. System.out.println("Setting item owningCollection fields in database");
```

[Table of Contents] [Description]

Path 21:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
      ...
173. System.out.println("Pruned " + count
```

[Table of Contents] [Description]

Path 22:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
108. public static void main(String[] args) //checksumchecker.java
       ...
205. System.err.println("The following argument: " + ids[i]
```

[Table of Contents] [Description]

Path 23:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
94. public static void main(String[] argv) throws Exception //itemexport.java
       ...
125. .println("\nfull collection: ItemExport -t COLLECTION -i ID -d dest -n
number");
```

[Table of Contents] [Description]

Path 24:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
430. private static void writeHandle(Context c, Item i, File
destDir) //itemexport.java
       ...
441. out.println(i.getHandle());
```

[Table of Contents] [Description]

Path 25:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
465. private static void writeBitstreams(Context c, Item i, File
destDir) //itemexport.java
       ...
510. out.println("-r -s " + b.getStoreNumber()
```

[Table of Contents] [Description]

Path 26:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
465. private static void writeBitstreams(Context c, Item i, File
destDir) //itemexport.java
       ...
514. out.println(myName + "\tbundle:" + bundleName);
```

[Table of Contents] [Description]

Path 27:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
139. public static void main(String[] argv) throws Exception //itemimport.java
     ...
179. .println("\nadding items: ItemImport -a -e eperson -c collection -s
sourcedir -m mapfile");
```

[Table of Contents] [Description]

Path 28:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
897. private String processHandleFile(Context c, Item i, String
path, //itemimport.java
     ...
933. .println("Non-critical problem releasing resources.");
```

[Table of Contents] [Description]

Path 29:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
103. public InputStream getDestinationStream(InputStream
source) //brandedpreviewjpegfilter.java
     ...
126. System.out.println("original size: " + xsize + "," + ysize);
```

[Table of Contents] [Description]

Path 30:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
95. public InputStream getDestinationStream(InputStream source) //jpegfilter.java
     ...
115. System.out.println("original size: " + xsize + "," + ysize);
```

[Table of Contents] [Description]

Path 31:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
112. public boolean processBitstream(Context c, Item item, Bitstream
source) //mediafilter.java
     ...
```

```
149. System.out.println("SKIPPED: bitstream " + source.getID()
```

[Table of Contents] [Description]

Path 32:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
96. public static void main(String[] argv) throws
Exception //mediafiltermanager.java
...
153. System.out.println("Invalid maximum value '" +
```

[Table of Contents] [Description]

Path 33:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
97. public InputStream getDestinationStream(InputStream source) //pdffilter.java
...
128. System.out.println(extractedText);
```

[Table of Contents] [Description]

Path 34:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
91. public InputStream getDestinationStream(InputStream source) //wordfilter.java
...
103. System.out.println(extractedText);
```

[Table of Contents] [Description]

Path 35:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
114. public static void main(String[] args) throws Exception //metsexport.java
...
139. .println("\nExport a collection: metsexport -c hdl:123.456/789");
```

[Table of Contents] [Description]

Path 36:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
114. public static void main(String[] args) throws Exception //metsexport.java
```

```
...
174. System.err.println(line.getOptionValue('i')
```

[Table of Contents] [Description]

Path 37:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
...
264. System.err.println("Caught exception:"+sqle.getCause());
```

[Table of Contents] [Description]

Path 38:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
123. private static void usageError(String msg) //packager.java
...
125. System.out.println(msg);
```

[Table of Contents] [Description]

Path 39:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
130. public static void main(String[] argv) throws Exception //packager.java
...
208. .println("Warning: Illegal package option format: \""
```

[Table of Contents] [Description]

Path 40:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
343. public static void processLogs(Context context, String
myLogDir, //loganalyser.java
...
400. System.out.println("Failed to read log file " + logFiles[i].toString());
```

[Table of Contents] [Description]

Path 41:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
        ...
462. System.out.println("Unable to write to output file " + output);
```

[Table of Contents] [Description]

Path 42:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
547. public static void readMap(String map) //reportgenerator.java
        ...
562. System.err.println("Failed to read map file: log file actions will be
displayed without translation");
```

[Table of Contents] [Description]

Path 43:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
        ...
74. out.println("<table align=\"center\" class=\"miscTable\"
title=\"Collection List\">");
```

[Table of Contents] [Description]

Path 44:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
        ...
77. out.print("<tr><th id=\"t4\" class=\"oddRowOddCol\">"
```

[Table of Contents] [Description]

Path 45:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
        ...
91. out.print("<tr><td headers=\"t4\" class=\"" + row + "RowEvenCol\">");
```

[Table of Contents] [Description]

Path 46:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
92. out.print("<a href=\"");
```

[Table of Contents] [Description]

Path 47:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
96. out.print(hrq.getContextPath() + "/handle/");
```

[Table of Contents] [Description]

Path 48:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
97. out.print(collections[i].getHandle());
```

[Table of Contents] [Description]

Path 49:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
98. out.print("\">");
```

[Table of Contents] [Description]

Path 50:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
      ...
99. out.print(name);
```

[Table of Contents] [Description]

Path 51:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
    ...
100. out.print("</a>");
```

[Table of Contents] [Description]

Path 52:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
    ...
102. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 53:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //collectionlisttag.java
    ...
107. out.println("</table>");
```

[Table of Contents] [Description]

Path 54:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
    ...
74. out.println("<table align=\"center\" class=\"miscTable\" title=\"Community
List\">");
```

[Table of Contents] [Description]

Path 55:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
    ...
77. out.print("<tr><th id=\"t5\" class=\"oddRowOddCol\">"
```

[Table of Contents] [Description]

Path 56:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
91. out.print("<tr><td headers=\"t5\" class=\"" + row + "RowEvenCol\">");
```

[Table of Contents] [Description]

Path 57:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
92. out.print("<a href=\"");
```

[Table of Contents] [Description]

Path 58:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
96. out.print(hrq.getContextPath() + "/handle/");
```

[Table of Contents] [Description]

Path 59:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
97. out.print(communities[i].getHandle());
```

[Table of Contents] [Description]

Path 60:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
98. out.print("\">");
```

[Table of Contents] [Description]

Path 61:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
      99. out.print(name);
```

[Table of Contents] [Description]

Path 62:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
      100. out.print("</a>");
```

[Table of Contents] [Description]

Path 63:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
      102. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 64:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public int doStartTag() throws JspException //communitylisttag.java
      ...
      107. out.println("</table>");
```

[Table of Contents] [Description]

Path 65:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
      161. out.println("<table align=\"center\" class=\"miscTable\" summary=\"This
      table browse all dspace content\">");
```

[Table of Contents] [Description]

Path 66:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
162. out.println("<tr>");
```

[Table of Contents] [Description]

Path 67:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
213. out.print("<th id=\"" + id + "\" class=\"" + css + "\">"
```

[Table of Contents] [Description]

Path 68:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
308. out.println("</tr>");
```

[Table of Contents] [Description]

Path 69:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
315. out.println("<tr>");
```

[Table of Contents] [Description]

Path 70:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
317. out.println(frags[i]);
```

[Table of Contents] [Description]

Path 71:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
320. out.println("</tr>");
```

[Table of Contents] [Description]

Path 72:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
119. public int doStartTag() throws JspException //itemlisttag.java
      ...
325. out.println("</table>");
```

[Table of Contents] [Description]

Path 73:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
      ...
120. out.println("<br/><p align=\"center\">");
```

[Table of Contents] [Description]

Path 74:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
      ...
121. out.println("<img src=\""
```

[Table of Contents] [Description]

Path 75:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
      ...
147. out.println("<br/>" + dcValue[0].value);
```

[Table of Contents] [Description]

Path 76:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
107. private void showPreview() throws SQLException,
IOException //itempreviewtag.java
      |...
      |151. out.println("</p>");
```

[Table of Contents] [Description]

Path 77:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
      |...
      |290. out.println("<center><table class=\"itemDisplayTable\">");
```

[Table of Contents] [Description]

Path 78:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
      |...
      |339. out.print("<tr><td class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 79:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
      |...
      |341. out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 80:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
      |...
```

```
344. out.print(": </td><td class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 81:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
350. out.print("<br />");
```

[Table of Contents] [Description]

Path 82:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
355. out.print("<a href=\"" + values[j].value + "\">"
```

[Table of Contents] [Description]

Path 83:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
443. out.print(Utils.addEntities(values[i].value));
```

[Table of Contents] [Description]

Path 84:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
     ...
367. out.print(Utils.addEntities(values[j].value));
```

[Table of Contents] [Description]

Path 85:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
```

```
...
371. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 86:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
...
377. out.println("</table></center><br/>");
```

[Table of Contents] [Description]

Path 87:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. private void render() throws IOException //itemtag.java
...
385. out.println("<br/><br/>");
```

[Table of Contents] [Description]

Path 88:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
400. out.println("<p align=\"center\">"
```

[Table of Contents] [Description]

Path 89:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
405. out.println("<center><table class=\"itemDisplayTable\">");
```

[Table of Contents] [Description]

Path 90:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
```

```
...
406. out.println("<tr><th id=\"s1\" class=\"standard\">"
```

[Table of Contents] [Description]

Path 91:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
432. .print("<tr><td headers=\"s1\" class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 92:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
433. out.print(values[i].schema);
```

[Table of Contents] [Description]

Path 93:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
434. out.print("." + values[i].element);
```

[Table of Contents] [Description]

Path 94:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
438. out.print("." + values[i].qualifier);
```

[Table of Contents] [Description]

Path 95:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
```

```
...
442. .print("</td><td headers=\"s2\" class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 96:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
443. out.print(Utils.addEntities(values[i].value));
```

[Table of Contents] [Description]

Path 97:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
445. .print("</td><td headers=\"s3\" class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 98:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
509. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 99:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
...
453. out.print(values[i].language);
```

[Table of Contents] [Description]

Path 100:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
```

```
      ...
456.  out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 101:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
      ...
462.  out.println("</table></center><br/>");
```

[Table of Contents] [Description]

Path 102:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
393. private void renderFull() throws IOException //itemtag.java
      ...
469.  out.println("<br/><br/>");
```

[Table of Contents] [Description]

Path 103:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
485.  out.print("<tr><td class=\"metadataFieldLabel\">");
```

[Table of Contents] [Description]

Path 104:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
      ...
488.  out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 105:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
```

```
...
493. out.print(LocaleSupport.getLocalizedMessage(pageContext,
```

[Table of Contents] [Description]

Path 106:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
496. out.print("</td><td class=\"metadataFieldValue\">");
```

[Table of Contents] [Description]

Path 107:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
500. out.print("<a href=\"");
```

[Table of Contents] [Description]

Path 108:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
501. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 109:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
502. out.print("/handle/");
```

[Table of Contents] [Description]

Path 110:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
```

```
...
503. out.print(collections[i].getHandle());
```

[Table of Contents] [Description]

Path 111:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
504. out.print("\">");
```

[Table of Contents] [Description]

Path 112:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
505. out.print(collections[i].getMetadata("name"));
```

[Table of Contents] [Description]

Path 113:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
506. out.print("</a><br/>");
```

[Table of Contents] [Description]

Path 114:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
477. private void listCollections() throws IOException //itemtag.java
...
509. out.println("</td></tr>");
```

[Table of Contents] [Description]

Path 115:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
522. out.print("<table align=\"center\" class=\"miscTable\"><tr>");
```

[Table of Contents] [Description]

Path 116:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
523. out.println("<td class=\"evenRowEvenCol\"><p><strong>"
```

[Table of Contents] [Description]

Path 117:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
534. out.println("<p>"
```

[Table of Contents] [Description]

Path 118:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
577. .println("<table cellpadding=\"6\"><tr><th id=\"t1\" class=\"standard\">"
```

[Table of Contents] [Description]

Path 119:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
586. .println("<th id=\"t2\" class=\"standard\">"
```

[Table of Contents] [Description]

Path 120:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
593. out.println("<th id=\"t3\" class=\"standard\">"
```

[Table of Contents] [Description]

Path 121:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
613. out.print("<tr><td headers=\"t1\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 122:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
614. out.print("<a target=\"_blank\" href=\"");
```

[Table of Contents] [Description]

Path 123:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
615. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 124:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
616. out.print("/html/");
```

[Table of Contents] [Description]

Path 125:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
617. out.print(handle + "/");
```

[Table of Contents] [Description]

Path 126:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
619. .print(UIUtil.encodeBitstreamName(primaryBitstream
```

[Table of Contents] [Description]

Path 127:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
621. out.print("\">");
```

[Table of Contents] [Description]

Path 128:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
622. out.print(primaryBitstream.getName());
```

[Table of Contents] [Description]

Path 129:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
623. out.print("</a>");
```

[Table of Contents] [Description]

Path 130:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
628. out.print("</td><td headers=\"t2\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 131:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
641. out.print("/html/");
```

[Table of Contents] [Description]

Path 132:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
634. out.print("</td><td headers=\"t3\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 133:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
635. out.print(UIUtil.formatFileSize(primaryBitstream.getSize()));
```

[Table of Contents] [Description]

Path 134:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
636. out.print("</td><td headers=\"t4\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 135:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
637. out.print(primaryBitstream.getFormatDescription());
```

[Table of Contents] [Description]

Path 136:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
639. .print("</td><td class=\"standard\"><a target=\"_blank\" href=\"");
```

[Table of Contents] [Description]

Path 137:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
640. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 138:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
641. out.print("/html/");
```

[Table of Contents] [Description]

Path 139:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
642. out.print(handle + "/");
```

[Table of Contents] [Description]

Path 140:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
644. .print(UIUtil.encodeBitstreamName(primaryBitstream
```

[Table of Contents] [Description]

Path 141:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
646. out.print("\">"
```

[Table of Contents] [Description]

Path 142:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
688. .print("<tr><td headers=\"t1\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 143:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
689. out.print(bsLink);
```

[Table of Contents] [Description]

Path 144:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
690. out.print(bitstreams[k].getName());
```

[Table of Contents] [Description]

Path 145:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
691. out.print("</a>");
```

[Table of Contents] [Description]

Path 146:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
697. .print("</td><td headers=\"t2\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 147:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
700. out.print((desc != null) ? desc : "");
```

[Table of Contents] [Description]

Path 148:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
704. .print("</td><td headers=\"t3\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 149:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
705. out.print(UIUtil.formatFileSize(bitstreams[k].getSize()));
```

[Table of Contents] [Description]

Path 150:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
707. .print("</td><td headers=\"t4\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 151:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
708. out.print(bitstreams[k].getFormatDescription());
```

[Table of Contents] [Description]

Path 152:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
710. .print("</td><td class=\"standard\" align=\"center\">");
```

[Table of Contents] [Description]

Path 153:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
730. out.print(bsLink);
```

[Table of Contents] [Description]

Path 154:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
731. out.print("<img src=\"" + myPath + "\" ");
```

[Table of Contents] [Description]

Path 155:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
```

```
...
732. out.print("alt=\"" + tAltText
```

[Table of Contents] [Description]

Path 156:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
738. .print(bsLink
```

[Table of Contents] [Description]

Path 157:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
749. out.println("</table>");
```

[Table of Contents] [Description]

Path 158:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
516. private void listBitstreams() throws IOException //itemtag.java
...
757. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 159:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
785. out.println("<table align=\"center\" class=\"attentionTable\"><tr>");
```

[Table of Contents] [Description]

Path 160:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
```

```
...
787. out.println("<td class=\"attentionCell\"><p><strong>"
```

[Table of Contents] [Description]

Path 161:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
798. out.print("<div align=\"center\" class=\"standard\">");
```

[Table of Contents] [Description]

Path 162:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
799. out.print("<strong><a target=\"_blank\" href=\"");
```

[Table of Contents] [Description]

Path 163:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
800. out.print(request.getContextPath());
```

[Table of Contents] [Description]

Path 164:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
801. out.print("/retrieve/");
```

[Table of Contents] [Description]

Path 165:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
```

```
...
802. out.print(bitstreams[k].getID() + "/");
```

[Table of Contents] [Description]

Path 166:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
803. out.print(UIUtil.encodeBitstreamName(bitstreams[k].getName(),
```

[Table of Contents] [Description]

Path 167:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
806. .print("\">"
```

[Table of Contents] [Description]

Path 168:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
769. private void showLicence() throws IOException //itemtag.java
...
814. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 169:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
...
140. out.print("<table><tr><td colspan=\"2\" align=\"center\"><select
multiple=\"multiple\" name=\"eperson_id\" size=\"");
```

[Table of Contents] [Description]

Path 170:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
141. out.print(multiple ? "10" : "1");
```

[Table of Contents] [Description]

Path 171:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
142. out.println("\">");
```

[Table of Contents] [Description]

Path 172:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
146. out.print("<option value=\"\"> </option>");
```

[Table of Contents] [Description]

Path 173:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
153. out.print("<option value=\"" + epeople[i].getID() + "\">");
```

[Table of Contents] [Description]

Path 174:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
154. out.print(epeople[i].getFullName() + " ("
```

[Table of Contents] [Description]

Path 175:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
156. out.println("</option>");
```

[Table of Contents] [Description]

Path 176:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
161. out.print("</select></td>");
```

[Table of Contents] [Description]

Path 177:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
165. out.print("</tr><tr><td width=\"50%\" align=\"center\">");
```

[Table of Contents] [Description]

Path 178:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
169. out.print("<td>");
```

[Table of Contents] [Description]

Path 179:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
177. out.print("<input type=\"button\" value=\"" + p
```

[Table of Contents] [Description]

Path 180:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
184. out.print("</td><td width=\"50%\" align=\"center\">");
```

[Table of Contents] [Description]

Path 181:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
185. out.print("<input type=\"button\" value=\""
```

[Table of Contents] [Description]

Path 182:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
132. public int doStartTag() throws JspException //selectepersontag.java
     ...
191. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 183:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
     ...
137. out.print("<table><tr><td colspan=\"2\" align=\"center\"><select
     multiple=\"multiple\" name=\"group_ids\" size=\"");
```

[Table of Contents] [Description]

Path 184:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
     ...
138. out.print(multiple ? "10" : "1");
```

[Table of Contents] [Description]

Path 185:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
139. out.println("\">");
```

[Table of Contents] [Description]

Path 186:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
144. out.print("<option value=\"\"> </option>");
```

[Table of Contents] [Description]

Path 187:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
161. out.print("</tr><tr><td width=\"50%\" align=\"center\">");
```

[Table of Contents] [Description]

Path 188:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
152. out.print(groups[i].getName() + " (" + groups[i].getID() + ")");
```

[Table of Contents] [Description]

Path 189:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
153. out.println("</option>");
```

[Table of Contents] [Description]

Path 190:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
157. out.print("</select></td>");
```

[Table of Contents] [Description]

Path 191:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
161. out.print("</tr><tr><td width=\"50%\" align=\"center\">");
```

[Table of Contents] [Description]

Path 192:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
165. out.print("<td>");
```

[Table of Contents] [Description]

Path 193:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
173. out.print("<input type=\"button\" value=\"" + p
```

[Table of Contents] [Description]

Path 194:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
       ...
180. out.print("</td><td width=\"50%\" align=\"center\">");
```

[Table of Contents] [Description]

Path 195:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
        ...
181. out.print("<input type=\"button\" value=\""
```

[Table of Contents] [Description]

Path 196:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
129. public int doStartTag() //selectgrouptag.java
        ...
187. out.println("</td></tr></table>");
```

[Table of Contents] [Description]

Path 197:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
67. public static void main(String[] argv) throws
Exception //fixdefaultpolicies.java
        ...
83. System.out.println("Collection " + t + " " + t.getMetadata("name"));
```

[Table of Contents] [Description]

Path 198:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
65. public static void main(String[] argv) throws Exception //policyset.java
        ...
70. .println("Args: containerType containerID contentType actionID groupID
command");
```

[Table of Contents] [Description]

Path 199:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
65. public static void main(String[] argv) //initializebrowse.java
        ...
71. System.out.print("Indexing all Items in DSpace....");
```

[Table of Contents] [Description]

Path 200:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
65. public static void main(String[] argv) //initializebrowse.java
      |...
      77. System.out.println(" ... Done");
```

[Table of Contents] [Description]

Path 201:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
65. public static void main(String[] argv) //initializebrowse.java
      |...
      86. System.err.println("Error: Browse index NOT created");
```

[Table of Contents] [Description]

Path 202:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
167. public static void main(String[] args) //dailyreportemailer.java
      |...
      215. .println("\nSend Deleted bitstream email report: DailyReportEmailer -d");
```

[Table of Contents] [Description]

Path 203:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
202. public static void main(String[] argv) throws
Exception //xsltingestioncrosswalk.java
      |...
      206. System.err.println("Usage: java XSLTIngestionCrosswalk [-l] <crosswalk-
      name> <input-file>");
```

[Table of Contents] [Description]

Path 204:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
374. public static String writeNewsFile(int position, String
news) //configurationmanager.java
      |...
      393. out.print(news);
```

[Table of Contents] [Description]

Path 205:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
410. public static void writeLicenseFile(String
newLicense) //configurationmanager.java
     ...
421. out.print(newLicense);
```

[Table of Contents] [Description]

Path 206:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
673. private static void installConfigurationFile(String
template) //configurationmanager.java
     ...
760. out.println(line);
```

[Table of Contents] [Description]

Path 207:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
781. public static void main(String[] argv) //configurationmanager.java
     ...
802. System.out.println(val);
```

[Table of Contents] [Description]

Path 208:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
781. public static void main(String[] argv) //configurationmanager.java
     ...
814. .println("Usage: ConfigurationManager OPTION\n -installTemplates install
config files for external tools\n -property prop.name get value of prop.name
from dspace.cfg");
```

[Table of Contents] [Description]

Path 209:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1115. public static void main(String[] argv) //historymanager.java
      ...
```

```
1127. System.out.println("Got unique id " + getUniqueId(c));
```

[Table of Contents] [Description]

Path 210:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
357. public static void doCMDLineQuery(String query) //dsquery.java
       ...
359. System.out.println("Command line query: " + query);
```

[Table of Contents] [Description]

SQL Injection

CWE ID	89
Description	SQL injection attacks are another instantiation of injection attack, in which SQL commands are injected into data-plane input in order to effect the execution of predefined SQL commands.
Likelihood of Exploit	Very High
Common Consequences	Confidentiality: Since SQL databases generally hold sensitive data, loss of confidentiality is a frequent problem with SQL injection vulnerabilities. Authentication: If poor SQL commands are used to check user names and passwords, it may be possible to connect to a system as another user with no previous knowledge of the password. Authorization: If authorization information is held in a SQL database, it may be possible to change this information through the successful exploitation of a SQL injection vulnerability. Integrity: Just as it may be possible to read sensitive information, it is also possible to make changes or even delete this information with a SQL injection attack.
Potential Mitigations	Requirements specification: A non-SQL style database which is not subject to this flaw may be chosen. Design: Follow the principle of least privilege when creating user accounts to a SQL database. Users should only have the minimum privileges necessary to use their account. If the requirements of the system indicate that a user can read and modify their own data, then limit their privileges so they cannot read/write others' data. Design: Duplicate any filtering done on the client-side on the server side.

Implementation: Implement SQL strings using prepared statements that bind variables. Prepared statements that do not bind variables can be vulnerable to attack.

Implementation: Use vigorous white-list style checking on any user input that may be used in a SQL command. Rather than escape meta-characters, it is safest to disallow them entirely. Reason: Later use of data that have been entered in the database may neglect to escape meta-characters before use. Narrowly define the set of safe characters based on the expected value of the parameter in the request.

Demonstrative Examples

The following code dynamically constructs and executes a SQL query that searches for items matching a specified name. The query restricts the items displayed to those where owner matches the user name of the currently-authenticated user.

C# Example:

```
...
string userName = ctx.getAuthenticatedUserName();
string query = "SELECT * FROM items WHERE owner = '"
+ userName + "' AND itemname = '" + ItemName.Text +
"'";
sda = new SqlDataAdapter(query, conn);
DataTable dt = new DataTable();
sda.Fill(dt);
...
```

The query that this code intends to execute follows: SELECT * FROM items WHERE owner = <userName> AND itemname = <itemName>; However, because the query is constructed dynamically by concatenating a constant base query string and a user input string, the query only behaves correctly if itemName does not contain a single-quote character. If an attacker with the user name wiley enters the string "name' OR 'a'='a" for itemName, then the query becomes the following: SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name' OR 'a'='a'; The addition of the OR 'a'='a' condition causes the where clause to always evaluate to true, so the query becomes logically equivalent to the much simpler query: SELECT * FROM items; This simplification of the query allows the attacker to bypass the requirement that the query only return items owned by the authenticated user; the query now returns all entries stored in the items table, regardless of their specified owner.

This example examines the effects of a different malicious

value passed to the query constructed and executed in the above example. If an attacker with the user name hacker enters the string "hacker'); DELETE FROM items; --" for itemName, then the query becomes the following two queries:

SQL Example:

```
SELECT * FROM items WHERE owner = 'wiley' AND
itemname = 'name';
DELETE FROM items;
--'
```

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database. Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [19]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in this way, the general attack could still be made effective using a trick similar to the one shown in Example 1. If an attacker enters the string "name'); DELETE FROM items; SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name'; DELETE FROM items; SELECT * FROM items WHERE 'a'='a'; One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, blacklisting is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers can: - Target fields that are not quoted - Find ways to bypass the need for certain escaped meta-characters - Use stored procedures to hide the injected meta-characters Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks. Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against

many others. For example, the following PL/SQL procedure is vulnerable to the same SQL injection attack shown in the first example. procedure get_item (itm_cv IN OUT ItmCurTyp, usr in varchar2, itm in varchar2) is open itm_cv for ' SELECT * FROM items WHERE ' || 'owner = '|| usr || ' AND itemname = ' || itm || '; end get_item; Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL injection attacks.

MS SQL has a built in function that enables shell command execution. An SQL injection in such a context could be disastrous. For example, a query of the form:

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='$user_input' ORDER BY PRICE

Where $user_input is taken from the user and unfiltered.

If the user provides the string:

' exec master..xp_cmdshell 'vol' --

The query will take the following form: "

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' exec master..xp_cmdshell 'vol' --' ORDER BY PRICE

Now, this query can be broken down into: [1] a first SQL query: SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' [2] a second SQL query, which executes a shell command: exec master..xp_cmdshell 'vol' [3] an MS SQL comment: --' ORDER BY PRICE As can be seen, the malicious input changes the semantics of the query into a query, a shell command execution and a comment.

Observed Examples

CVE-2004-0366 -

CVE-2004-0343 -

CVE-2003-0779 -

CVE-2003-0500 -

CVE-2003-0377 -

Context Notes

SQL injection has become a common issue with database-driven web sites. The flaw is easily detected, and easily exploited, and as such, any site or software package with even a minimal user base is likely to be subject to an attempted attack of this kind. Essentially, the attack is accomplished by placing a meta character into data input to then place SQL commands in the control plane, which did not exist there before. This flaw depends on the fact that SQL makes no real distinction between the control and data planes.

If successful, SQL Injection attacks can give an attacker access to backend database contents, the ability to remotely execute system commands, or in some circumstances the means to take control of the Windows server hosting the database.

Dynamically generating queries that include user input can lead to SQL injection attacks. An attacker can insert SQL commands or modifiers in the user input that can cause the query to behave in an unsafe manner.

Constructing a dynamic SQL statement with user input may allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.

Factors: resultant to special character mismanagement, MAID, or blacklist/whitelist problems. Can be primary to authentication errors.

References

M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.

Node Relationships

ChildOf - Injection (74)
ChildOf - Weaknesses in OWASP Top Ten (629)
ChildOf - Weaknesses Examined by SAMATE (630)
ChildOf - Weaknesses Used by NVD (635)
ParentOf - SQL Injection: Hibernate (564)

Source Taxonomies

PLOVER - SQL injection

7 Pernicious Kingdoms - SQL Injection

CLASP - SQL injection

Applicable Platforms

All

Path 1:

Query Name - **Second_Order_SQL_Injection**

Severity - **Critical**

```
1049. public static void loadSql(Reader r) throws SQLException,
IOException //databasemanager.java
```

```
...
1068. while ((line = reader.readLine()) != null)
...
1074. commentStart) : line;
...
1083. sql.append(input.replace(';', ' ')); // remove all semicolons
...
1130. SQL = sql.toString();
...
1136. boolean succeeded = statement.execute(SQL);
```

[Table of Contents] [Description]

Cross-site Scripting (XSS)

CWE ID	79
Description	A cross-site scripting weakness occurs when dynamically generated web pages display unvalidated, unfiltered, and unencoded user input allowing an attacker to embed malicious scripts into the generated page. This can be leveraged to execute scripting code as if it came from the site's server on the computer of anyone who uses the site.
Alternate Terms	"CSS" was once used as the acronym for this problem, but this can cause confusion with the "Cascading Style Sheets," so this acronym has declined significantly. Its use is discouraged by CWE.
Likelihood of Exploit	High to Very High
Weakness Ordinality	Resultant (Weakness is typically related to the presence of some other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Common Consequences	Confidentiality: The most common attack performed with cross-site scripting involves the disclosure of information stored in user cookies. Access control: In some circumstances it may be possible to run arbitrary code on a victim's computer when cross-site scripting is combined with other flaws. If successful, cross-site scripting vulnerabilities can be exploited to manipulate or steal cookies, create requests that can be mistaken for those of a valid user, compromise confidential information, or execute malicious code on the end user systems for a variety of nefarious purposes.
Potential Mitigations	Carefully check each input parameter against a rigorous positive specification (white list) defining the specific characters and format allowed. All input should be sanitized, not just parameters that the user is supposed to specify, but all data in the request, including hidden fields, cookies, headers, the URL itself, and so forth. A common mistake that leads to continuing XSS vulnerabilities is to validate only fields that are expected to be redisplayed by the site. We often encounter data from the request that is reflected by the application server or the application that the development team did not anticipate. Also, a field that

is not currently reflected may be used by a future developer. Therefore, validating ALL parts of the HTTP request is recommended.

This involves "HTML Entity Encoding" all non-alphanumeric characters from data that was received from the user and is now being written to the request.

With Struts, you should write all data from form beans with the bean's filter attribute set to true.

To help mitigate XSS attacks against the user's session cookie, set the session cookie to be HttpOnly. In browsers that support the HttpOnly feature (such as Internet Explorer), this attribute prevents the user's session cookie from being accessed by client-side scripts, including scripts inserted due to a XSS attack.

Demonstrative Examples

The following JSP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

JSP Example:

```
<% String eid = request.getParameter("eid"); %>
...
Employee ID: <%= eid %>
```

The following ASP.NET code segment reads an employee ID number from an HTTP request and displays it to the user.

ASP.NET Example:

```
protected System.Web.UI.WebControls.TextBox Login;
protected System.Web.UI.WebControls.Label EmployeeID;
  ...
EmployeeID.Text = Login.Text;
```

The code in this example operates correctly if eid contains only standard alphanumeric text. If eid has a value that includes meta-characters or source code, then the code will be executed by the web browser as it displays the HTTP response. Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use e-mail or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to

their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

The following JSP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

JSP Example:

```
<%...
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("select * from emp
where id="+eid);
if (rs != null) {
rs.next();
String name = rs.getString("name");
%>

Employee Name: <%= name %>
```

The following ASP.NET code segment queries a database for an employee with a given employee ID and prints the name corresponding with the ID.

ASP.NET Example:

```
protected System.Web.UI.WebControls.Label
EmployeeName;
  ...
string query = "select * from emp where id=" + eid;
sda = new SqlDataAdapter(query, conn);
sda.Fill(dt);
string name = dt.Rows[0]["Name"];
  ...
EmployeeName.Text = name;
```

This code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if they are not. Again, this code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker can execute malicious commands in the user's web browser. This type

	of exploit, known as Stored XSS, is particularly insidious because the indirection caused by the data store makes it more difficult to identify the threat and increases the possibility that the attack will affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code. As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim: * As in the previous example, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or e-mailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that may include session information, from the user's machine to the attacker or perform other nefarious activities. * As in this example, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Stored XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. * A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.
Context Notes	Cross-site scripting (XSS) vulnerabilities occur when an attacker uses a web application to send malicious code, generally JavaScript, to a different end user. When a web application uses input from a user in the output it generates without filtering it, an attacker can insert an attack in that input and the web application sends the attack to other users. The end user trusts the web application, and the attacks exploit that trust to do things that would not normally be allowed. Attackers frequently

use a variety of methods to encode the malicious portion of the tag, such as using Unicode, so the request looks less suspicious to the user. XSS attacks can generally be categorized into two categories: stored and reflected. Stored attacks are those where the injected code is permanently stored on the target servers in a database, message forum, visitor log, and so forth. Reflected attacks are those where the injected code takes another route to the victim, such as in an email message, or on some other server. When a user is tricked into clicking a link or submitting a form, the injected code travels to the vulnerable web server, which reflects the attack back to the user's browser. The browser then executes the code because it came from a 'trusted' server. For a reflected XSS attack to work, the victim must submit the attack to the server. This is still a very dangerous attack given the number of possible ways to trick a victim into submitting such a malicious request, including clicking a link on a malicious Web site, in an email, or in an inner-office posting. XSS flaws are very likely in web applications, as they require a great deal of developer discipline to avoid them in most applications. It is relatively easy for an attacker to find XSS vulnerabilities. Some of these vulnerabilities can be found using scanners, and some exist in older web application servers. The consequence of an XSS attack is the same regardless of whether it is stored or reflected. The difference is in how the payload arrives at the server. XSS can cause a variety of problems for the end user that range in severity from an annoyance to complete account compromise. The most severe XSS attacks involve disclosure of the user's session cookie, which allows an attacker to hijack the user's session and take over their account. Other damaging attacks include the disclosure of end user files, installation of Trojan horse programs, redirecting the user to some other page or site, and modifying presentation of content. Cross-site scripting (XSS) vulnerabilities occur when: 1. Data enters a Web application through an untrusted source, most frequently a web request. 2. The data is included in dynamic content that is sent to a web user without being validated for malicious code. The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Cross-site scripting attacks can occur wherever an untrusted user has the ability to publish content to a trusted web site. Typically, a malicious user will craft a client-side script, which -- when parsed by a web browser

	-- performs some activity (such as sending all site cookies to a given E-mail address). If the input is unchecked, this script will be loaded and run by each user visiting the web site. Since the site requesting to run the script has access to the cookies in question, the malicious script does also. There are several other possible attacks, such as running "Active X" controls (under Microsoft Internet Explorer) from sites that a user perceives as trustworthy; cookie theft is however by far the most common. All of these attacks are easily prevented by ensuring that no script tags -- or for good measure, HTML tags at all -- are allowed in data to be posted publicly. Cross-site scripting attacks may occur anywhere that possibly malicious users are allowed to post unregulated material to a trusted web site for the consumption of other valid users. The most common example can be found in bulletin-board web sites which provide web based mailing list-style functionality. When used in conjunction with Cross-Site Request Forgery (CSRF), XSS worms can cause significant damage in web sites with large user populations.
References	Jeremiah Grossman, Robert "RSnake" Hansen, Petko "pdp" D. Petkov, Anton Rager and Seth Fogie. "XSS Attacks". Syngress. 2007. M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.
Node Relationships	ChildOf - Injection (74) ResultsIn - Mobile Code: Invoking Untrusted Mobile Code (494) Peer - Cross-Site Request Forgery (CSRF) (352) ChildOf - Weaknesses in OWASP Top Ten (629) ChildOf - Weaknesses Used by NVD (635) ParentOf - Basic XSS (80) ParentOf - XSS in Error Pages (81) ParentOf - Script in IMG Tags (82) ParentOf - XSS using Script in Attributes (83) ParentOf - XSS using Script Via Encoded URI Schemes (84) ParentOf - Doubled Character XSS Manipulations (85) ParentOf - Invalid Characters in Identifiers (86) ParentOf - Alternate XSS Syntax (87)
Source Taxonomies	PLOVER - Cross-site scripting (XSS) 7 Pernicious Kingdoms - Cross-site Scripting CLASP - Cross-site scripting
Applicable Platforms	All

Path 1:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <%-- //eperson-browse.jsp
     ...
     169. <input type="hidden" name="sortby" value="<%= request.getParameter
     ("sortby") %>"/>
```

[Table of Contents] [Description]

Path 2:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
57. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularyservlet.java
     ...
     64. String callerUrl = request.getParameter("callerUrl");
     ...
     75. response.sendRedirect(callerUrl);
```

[Table of Contents] [Description]

Cross-site Scripting (XSS)

CWE ID	79
Description	A cross-site scripting weakness occurs when dynamically generated web pages display unvalidated, unfiltered, and unencoded user input allowing an attacker to embed malicious scripts into the generated page. This can be leveraged to execute scripting code as if it came from the site's server on the computer of anyone who uses the site.
Alternate Terms	"CSS" was once used as the acronym for this problem, but this can cause confusion with the "Cascading Style Sheets," so this acronym has declined significantly. Its use is discouraged by CWE.
Likelihood of Exploit	High to Very High
Weakness Ordinality	Resultant (Weakness is typically related to the presence of some other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Common Consequences	Confidentiality: The most common attack performed with cross-site scripting involves the disclosure of information stored in user cookies. Access control: In some circumstances it may be possible to run arbitrary code on a victim's computer when cross-site scripting is combined with other flaws. If successful, cross-site scripting vulnerabilities can be exploited to manipulate or steal cookies, create requests that can be mistaken for those of a valid user, compromise confidential information, or execute malicious code on the end user systems for a variety of nefarious purposes.

Potential Mitigations

Carefully check each input parameter against a rigorous positive specification (white list) defining the specific characters and format allowed. All input should be sanitized, not just parameters that the user is supposed to specify, but all data in the request, including hidden fields, cookies, headers, the URL itself, and so forth. A common mistake that leads to continuing XSS vulnerabilities is to validate only fields that are expected to be redisplayed by the site. We often encounter data from the request that is reflected by the application server or the application that the development team did not anticipate. Also, a field that is not currently reflected may be used by a future developer. Therefore, validating ALL parts of the HTTP request is recommended.

This involves "HTML Entity Encoding" all non-alphanumeric characters from data that was received from the user and is now being written to the request.

With Struts, you should write all data from form beans with the bean's filter attribute set to true.

To help mitigate XSS attacks against the user's session cookie, set the session cookie to be HttpOnly. In browsers that support the HttpOnly feature (such as Internet Explorer), this attribute prevents the user's session cookie from being accessed by client-side scripts, including scripts inserted due to a XSS attack.

Demonstrative Examples

The following JSP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

JSP Example:

```
<% String eid = request.getParameter("eid"); %>
...
Employee ID: <%= eid %>
```

The following ASP.NET code segment reads an employee ID number from an HTTP request and displays it to the user.

ASP.NET Example:

```
protected System.Web.UI.WebControls.TextBox Login;
protected System.Web.UI.WebControls.Label EmployeeID;
  ...
EmployeeID.Text = Login.Text;
```

The code in this example operates correctly if eid contains

only standard alphanumeric text. If eid has a value that includes meta-characters or source code, then the code will be executed by the web browser as it displays the HTTP response. Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use e-mail or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

The following JSP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

JSP Example:

```
<%...
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("select * from emp
where id="+eid);
if (rs != null) {
rs.next();
String name = rs.getString("name");
%>

Employee Name: <%= name %>
```

The following ASP.NET code segment queries a database for an employee with a given employee ID and prints the name corresponding with the ID.

ASP.NET Example:

```
protected System.Web.UI.WebControls.Label
EmployeeName;
  ...
string query = "select * from emp where id=" + eid;
sda = new SqlDataAdapter(query, conn);
sda.Fill(dt);
string name = dt.Rows[0]["Name"];
  ...
EmployeeName.Text = name;
```

This code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if they are not. Again, this code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker can execute malicious commands in the user's web browser. This type of exploit, known as Stored XSS, is particularly insidious because the indirection caused by the data store makes it more difficult to identify the threat and increases the possibility that the attack will affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code. As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim: * As in the previous example, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or e-mailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that may include session information, from the user's machine to the attacker or perform other nefarious activities. * As in this example, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Stored XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. * A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included

	in dynamic content.
Context Notes	Cross-site scripting (XSS) vulnerabilities occur when an attacker uses a web application to send malicious code, generally JavaScript, to a different end user. When a web application uses input from a user in the output it generates without filtering it, an attacker can insert an attack in that input and the web application sends the attack to other users. The end user trusts the web application, and the attacks exploit that trust to do things that would not normally be allowed. Attackers frequently use a variety of methods to encode the malicious portion of the tag, such as using Unicode, so the request looks less suspicious to the user. XSS attacks can generally be categorized into two categories: stored and reflected. Stored attacks are those where the injected code is permanently stored on the target servers in a database, message forum, visitor log, and so forth. Reflected attacks are those where the injected code takes another route to the victim, such as in an email message, or on some other server. When a user is tricked into clicking a link or submitting a form, the injected code travels to the vulnerable web server, which reflects the attack back to the user's browser. The browser then executes the code because it came from a 'trusted' server. For a reflected XSS attack to work, the victim must submit the attack to the server. This is still a very dangerous attack given the number of possible ways to trick a victim into submitting such a malicious request, including clicking a link on a malicious Web site, in an email, or in an inner-office posting. XSS flaws are very likely in web applications, as they require a great deal of developer discipline to avoid them in most applications. It is relatively easy for an attacker to find XSS vulnerabilities. Some of these vulnerabilities can be found using scanners, and some exist in older web application servers. The consequence of an XSS attack is the same regardless of whether it is stored or reflected. The difference is in how the payload arrives at the server. XSS can cause a variety of problems for the end user that range in severity from an annoyance to complete account compromise. The most severe XSS attacks involve disclosure of the user's session cookie, which allows an attacker to hijack the user's session and take over their account. Other damaging attacks include the disclosure of end user files, installation of Trojan horse programs, redirecting the user to some other page or site, and modifying presentation of content. Cross-site scripting (XSS) vulnerabilities occur when: 1. Data enters a Web application through an untrusted source, most frequently a web request. 2. The data is included in dynamic content that is sent to a web user without being validated for malicious code. The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks

based on XSS is almost limitless, but they commonly include transmitting private data like cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Cross-site scripting attacks can occur wherever an untrusted user has the ability to publish content to a trusted web site. Typically, a malicious user will craft a client-side script, which -- when parsed by a web browser -- performs some activity (such as sending all site cookies to a given E-mail address). If the input is unchecked, this script will be loaded and run by each user visiting the web site. Since the site requesting to run the script has access to the cookies in question, the malicious script does also. There are several other possible attacks, such as running "Active X" controls (under Microsoft Internet Explorer) from sites that a user perceives as trustworthy; cookie theft is however by far the most common. All of these attacks are easily prevented by ensuring that no script tags -- or for good measure, HTML tags at all -- are allowed in data to be posted publicly.

Cross-site scripting attacks may occur anywhere that possibly malicious users are allowed to post unregulated material to a trusted web site for the consumption of other valid users. The most common example can be found in bulletin-board web sites which provide web based mailing list-style functionality.

When used in conjunction with Cross-Site Request Forgery (CSRF), XSS worms can cause significant damage in web sites with large user populations.

References

Jeremiah Grossman, Robert "RSnake" Hansen, Petko "pdp" D. Petkov, Anton Rager and Seth Fogie. "XSS Attacks". Syngress. 2007.

M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.

Node Relationships

ChildOf - Injection (74)
ResultsIn - Mobile Code: Invoking Untrusted Mobile Code (494)
Peer - Cross-Site Request Forgery (CSRF) (352)
ChildOf - Weaknesses in OWASP Top Ten (629)
ChildOf - Weaknesses Used by NVD (635)
ParentOf - Basic XSS (80)
ParentOf - XSS in Error Pages (81)
ParentOf - Script in IMG Tags (82)
ParentOf - XSS using Script in Attributes (83)
ParentOf - XSS using Script Via Encoded URI Schemes (84)
ParentOf - Doubled Character XSS Manipulations (85)
ParentOf - Invalid Characters in Identifiers (86)
ParentOf - Alternate XSS Syntax (87)

Source Taxonomies PLOVER - Cross-site scripting (XSS)

7 Pernicious Kingdoms - Cross-site Scripting

CLASP - Cross-site scripting

Applicable Platforms All

Path 1:

Query Name - **Stored_XSS**

Severity - **Critical**

```
325. public static String readNewsFile(int position) //configurationmanager.java
      ...
      349. while ((lineIn = br.readLine()) != null)
      ...
      351. text += lineIn;
      ...
      361. return text;
```

```
1. <%-- //home.jsp
      ...
      63. String topNews = ConfigurationManager.readNewsFile(Constants.NEWS_TOP);
      ...
      79. <td class="oddRowEvenCol"><%= topNews %></td>
```

[Table of Contents] [Description]

Path 2:

Query Name - **Stored_XSS**

Severity - **Critical**

```
325. public static String readNewsFile(int position) //configurationmanager.java
      ...
      349. while ((lineIn = br.readLine()) != null)
      ...
      351. text += lineIn;
      ...
      361. return text;
```

```
1. <%-- //home.jsp
      ...
      64. String sideNews = ConfigurationManager.readNewsFile(Constants.NEWS_SIDE);
      ...
      137. <%= sideNews %>
```

[Table of Contents] [Description]

Path 3:

Query Name - **Stored_XSS**

Severity - **Critical**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
```

```
...
521. while ((lineIn = br.readLine()) != null)
...
523. license = license + lineIn + '\n';
```

```
296. public static String getDefaultSubmissionLicense() //configurationmanager.java
...
300. loadConfig(null);
...
303. return license;
```

```
653. public String getLicense() //collection.java
...
660. license = ConfigurationManager.getDefaultSubmissionLicense();
...
663. return license;
```

```
1. <%-- //edit-collection.jsp
...
100. license = collection.getLicense();
...
215. <textarea name="license" rows="6" cols="50"><%= license %></textarea>
```

[Table of Contents] [Description]

LDAP Injection

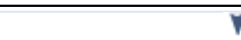
CWE ID	90
Description	The software does not properly filter or quote special characters or reserved words that are used in LDAP queries or responses, allowing attackers to modify the syntax, contents, or commands of the LDAP query before it is executed.
Potential Mitigations	Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Context Notes	Factors: resultant to special character mismanagement, MAID, or blacklist/whitelist problems. Can be primary to authentication and verification errors.
Research Gaps	Under-reported. This is likely found very frequently by third party code auditors, but there are very few publicly reported examples.
References	SPI Dynamics. "Web Applications and LDAP Injection".
Node Relationships	ChildOf - Injection (74) ChildOf - Weaknesses in OWASP Top Ten (629)
Source Taxonomies	PLOVER - LDAP injection
Applicable Platforms	All

Path 1:

Query Name - **LDAP_Injection**

Severity - **Critical**

```
101. protected void doDSPost(Context context, //ldapservlet.java
        ...
        107. String netid = request.getParameter("login_netid");
        ...
        189. if (ldapAuthenticate(netid, password, context))
```



```
253. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapservlet.java
        ...
        270. env.put(javax.naming.Context.SECURITY_PRINCIPAL,
        ldap_id_field+"="+netid+","+ldap_object_context);
        271. env.put(javax.naming.Context.SECURITY_CREDENTIALS, password);
        ...
        276. DirContext ctx = new InitialDirContext(env);
```

[Table of Contents] [Description]

Path 2:

Query Name - **LDAP_Injection**

Severity - **Critical**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
        ...
        257. String netid = request.getParameter("netid");
        ...
        336. env.put(javax.naming.Context.SECURITY_PRINCIPAL,
        ldap_id_field+"="+netid+","+ldap_search_context);
        ...
        341. DirContext ctx = new InitialDirContext(env);
```

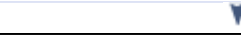
[Table of Contents] [Description]

Path 3:

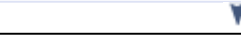
Query Name - **LDAP_Injection**

Severity - **Critical**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
        ...
        88. String email = request.getParameter("login_email");
        ...
        93. int status = AuthenticationManager.authenticate(context, email, password,
```



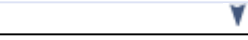
```
131. public static int authenticate(Context context, //authenticationmanager.java
        132. String username,
        ...
        137. return authenticateInternal(context, username, password, realm,
```



```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
        183. String username,
        ...
```

```
200. ret = methodStack[i].authenticate(context, username, password, realm,
request);
```

```
142. public int authenticate(Context context, //ldapauthentication.java
143. String netid,
...
172. if (ldap.ldapAuthenticate(netid, password, context))
```



```
290. protected boolean ldapAuthenticate(String netid, String password, Context
context) //ldapauthentication.java
...
306. env.put(javax.naming.Context.SECURITY_PRINCIPAL,
ldap_id_field+"="+netid+","+ldap_object_context);
...
313. ctx = new InitialDirContext(env);
```

[Table of Contents] [Description]

Log Forging

CWE ID	117
Description	Writing unvalidated user input into log files can allow an attacker to forge log entries or inject malicious content into logs.
Likelihood of Exploit	Medium
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid, expected and appropriate input is processed by the system.
Demonstrative Examples	The following web application code attempts to read an integer value from a request object. If the value fails to parse as an integer, then the input is logged with an error message indicating what happened.

```
...
String val = request.getParameter("val");
  try {
    int value = Integer.parseInt(val);
  }
  catch (NumberFormatException) {
    log.info("Failed to parse val = " + val);
  }
...
```

If a user submits the string "twenty-one" for val, the following entry is logged: INFO: Failed to parse val=twenty-one However, if an attacker submits the string "twenty-one%0a%0aINFO:+User+logged+out%3dbadguy", the following entry is logged: INFO: Failed to parse val=twenty-one INFO: User

	logged out=badguy Clearly, attackers can use this same mechanism to insert arbitrary log entries.
Context Notes	Log forging vulnerabilities occur when: 1. Data enters an application from an untrusted source. 2. The data is written to an application or system log file. Applications typically use log files to store a history of events or transactions for later review, statistics gathering, or debugging. Depending on the nature of the application, the task of reviewing log files may be performed manually on an as-needed basis or automated with a tool that automatically culls logs for important events or trending information. Interpretation of the log files may be hindered or misdirected if an attacker can supply data to the application that is subsequently logged verbatim. In the most benign case, an attacker may be able to insert false entries into the log file by providing the application with input that includes appropriate characters. If the log file is processed automatically, the attacker can render the file unusable by corrupting the format of the file or injecting unexpected characters. A more subtle attack might involve skewing the log file statistics. Forged or otherwise, corrupted log files can be used to cover an attacker's tracks or even to implicate another party in the commission of a malicious act [22]. In the worst case, an attacker may inject code or other commands into the log file and take advantage of a vulnerability in the log processing utility [17].
References	[17] G. Hoglund and G. McGraw. "Exploiting Software: How to Break Code". Addison-Wesley. February 2004. [22] A. Muffet. "The night the log was forged". <http://doc.novsu.ac.ru/oreilly/tcpip/puis/ch10_05.htm>.
Node Relationships	ChildOf - Output Validation (116)
Source Taxonomies	7 Pernicious Kingdoms - Log Forging
Applicable Platforms	All

Path 1:

Query Name - **Log_Forgery**

Severity - **Warning**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
      ...
      92. String idString = request.getPathInfo();
      ...
      108. idString = idString.substring(1);
      ...
      118. handle = idString.substring(0, slashIndex);
      ...
      148. "handle=" + handle + ",withdrawn=true"));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
```

```
90. return result;
```

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
...
147. log.info(LogManager.getHeader(context, "view_bitstream",
```

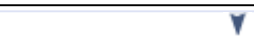
[Table of Contents] [Description]

Path 2:

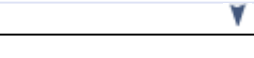
Query Name - **Log_Forgery**

Severity - **Warning**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
...
92. String idString = request.getPathInfo();
...
108. idString = idString.substring(1);
...
177. + idString));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
...
176. log.info(LogManager.getHeader(context, "invalid_id", "path="
```

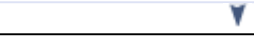
[Table of Contents] [Description]

Path 3:

Query Name - **Log_Forgery**

Severity - **Warning**

```
122. private List extractKeywords(HttpServletRequest
request) //controlledvocabularysearchservlet.java
...
131. keywords.add("\"" + request.getParameter(element) + "\"");
...
134. return keywords;
```



```
87. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
...
97. List keywords = extractKeywords(request);
```

```
98. String query = join(keywords, " or ");
```

```
367. public static String join(List list, String
separator) //controlledvocabularysearchservlet.java
     ...
     370. Iterator iterator = list.listIterator();
     ...
     373. result += iterator.next().toString();
     ...
     375. result += separator;
     ...
     377. return result;
```

```
87. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
     ...
     98. String query = join(keywords, " or ");
     99. doSearch(context, request, query);
```

```
149. private void doSearch(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
     150. String query) throws IOException, SQLException
     ...
     322. + query + "\",results=(" + resultsCommunities.length + ","
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

```
149. private void doSearch(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
     ...
     321. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Path 4:

Query Name - **Log_Forgery**

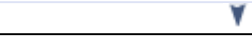
Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
     368. Enumeration e = request.getParameterNames();
     ...
     372. String name = (String) e.nextElement();
     ...
     378. report = report + "-- " + name + ": *not logged*\n";
     ...
     382. report = report + "-- " + name + ": \""
```

```
...
387. return report;
```

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
...
141. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
...
140. log.debug(LogManager.getHeader(context, "http_request", UIUtil
```

[Table of Contents] [Description]

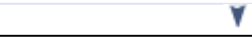
Path 5:

Query Name - **Log_Forgery**

Severity - **Warning**

```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
...
102. String formEmail = request.getParameter("email");
...
157. + formEmail));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
73. protected void doDSGet(Context context, HttpServletRequest
request, //feedbackservlet.java
...
156. log.info(LogManager.getHeader(context, "sent_feedback", "from="
```

[Table of Contents] [Description]

Path 6:

Query Name - **Log_Forgery**

Severity - **Warning**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
```

```
...
162. String path = request.getPathInfo();
...
176. path = path.substring(1);
...
198. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```



```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
    198. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```

[Table of Contents] [Description]

Path 7:

Query Name - **Log_Forgery**

Severity - **Warning**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
    162. String path = request.getPathInfo();
    ...
    176. path = path.substring(1);
    ...
    206. log.info(LogManager.getHeader(context, "invalid_syndformat", "path=" +
    path));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```



```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
    ...
    206. log.info(LogManager.getHeader(context, "invalid_syndformat", "path=" +
    path));
```

[Table of Contents] [Description]

Path 8:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    102. String path = request.getPathInfo();
    ...
    107. path = path.substring(1);
    ...
    142. .getHeader(context, "invalid_id", "path=" + path));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    142. .getHeader(context, "invalid_id", "path=" + path));
```

[Table of Contents] [Description]

Path 9:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    102. String path = request.getPathInfo();
    ...
    107. path = path.substring(1);
    ...
    118. handle = path.substring(0, secondSlash);
    ...
    235. + handle));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    233. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 10:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    102. String path = request.getPathInfo();
    ...
    107. path = path.substring(1);
    ...
    118. handle = path.substring(0, secondSlash);
    ...
    168. displayItem(context, request, response, item, handle);
```

```
256. private void displayItem(Context context, HttpServletRequest
request, //handleservlet.java
    257. HttpServletResponse response, Item item, String handle)
    ...
    274. + handle));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
256. private void displayItem(Context context, HttpServletRequest
request, //handleservlet.java
    ...
    273. .info(LogManager.getHeader(context, "view_item", "handle="
```

[Table of Contents] [Description]

Path 11:

Query Name - **Log_Forgery**

Severity - **Warning**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
    ...
    152. String idString = request.getPathInfo();
    ...
    163. idString = idString.substring(1);
    ...
    173. handle = idString.substring(0, slashIndex);
    ...
    239. + handle + ",bitstream_id=" + bitstream.getID()));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java

    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
```

```
90. return result;
```

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
...
238. log.info(LogManager.getHeader(context, "view_html", "handle="
```

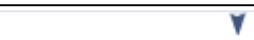
[Table of Contents] [Description]

Path 12:

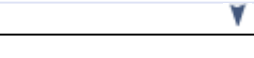
Query Name - **Log_Forgery**

Severity - **Warning**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
...
152. String idString = request.getPathInfo();
...
163. idString = idString.substring(1);
...
259. "invalid_bitstream_id=" + idString));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
...
258. log.info(LogManager.getHeader(context, "view_html",
```

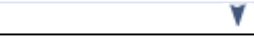
[Table of Contents] [Description]

Path 13:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```



```
69. protected void doGet(HttpServletRequest request, //internalerrorservlet.java
      ...
      76. String logInfo = UIUtil.getRequestLogInfo(request);
      ...
      81. + ":internal_error:" + logInfo;
      ...
      83. log.warn(logMessage, t);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Log_Forgery**

Severity - **Warning**

```
101. protected void doDSPost(Context context, //ldapservlet.java
      ...
      107. String netid = request.getParameter("login_netid");
      ...
      195. "autoregister", "netid=" + netid));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

▼

```
101. protected void doDSPost(Context context, //ldapservlet.java
      ...
      194. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 15:

Query Name - **Log_Forgery**

Severity - **Warning**

```
101. protected void doDSPost(Context context, //ldapservlet.java
      ...
      107. String netid = request.getParameter("login_netid");
      ...
      246. "netid=" + netid));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

▼

```
101. protected void doDSPost(Context context, //ldapservlet.java
      ...
```

```
244. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 16:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
368. Enumeration e = request.getParameterNames();
     ...
372. String name = (String) e.nextElement();
     ...
378. report = report + "-- " + name + ": *not logged*\n";
     ...
382. report = report + "-- " + name + ": \""
     ...
387. return report;
```

▼

```
102. protected void doDSPost(Context context, HttpServletRequest
request, //mydspaceservlet.java
     ...
138. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

▼

```
102. protected void doDSPost(Context context, HttpServletRequest
request, //mydspaceservlet.java
     ...
137. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 17:

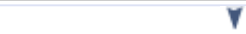
Query Name - **Log_Forgery**

Severity - **Warning**

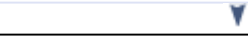
```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
368. Enumeration e = request.getParameterNames();
     ...
372. String name = (String) e.nextElement();
     ...
378. report = report + "-- " + name + ": *not logged*\n";
     ...
```

```
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
221. UIUtil.getRequestLogInfo(request)));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
220. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 18:

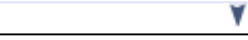
Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
...
275. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      274. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 19:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      368. Enumeration e = request.getParameterNames();
      ...
      372. String name = (String) e.nextElement();
      ...
      378. report = report + "-- " + name + ": *not logged*\n";
      ...
      382. report = report + "-- " + name + ": \""
      ...
      387. return report;
```

▼

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      312. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

▼

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      311. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 20:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      368. Enumeration e = request.getParameterNames();
      ...
```

```
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
    ...
369. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
    ...
368. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 21:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
    ...
426. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
      ...
      425. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 22:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      368. Enumeration e = request.getParameterNames();
      ...
      372. String name = (String) e.nextElement();
      ...
      378. report = report + "-- " + name + ": *not logged*\n";
      ...
      382. report = report + "-- " + name + ": \""
      ...
      387. return report;
```

▼

```
512. private void processRejectReason(Context context, //mydspaceservlet.java
      ...
      535. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

▼

```
512. private void processRejectReason(Context context, //mydspaceservlet.java
      ...
      534. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 23:

Query Name - **Log_Forgery**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java

      ...
      73. String title = request.getParameter("title");
      ...
      81. query = query + " " + title;
      ...
      87. query = query + " " + authorFirst;
      ...
```

```
93. query = query + " " + authorLast;
...
98. + "dspacequery=" + query));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    97. log.info(LogManager.getHeader(context, "openURL", logInfo
```

[Table of Contents] [Description]

Path 24:

Query Name - **Log_Forgery**

Severity - **Warning**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
    ...
    88. String email = request.getParameter("login_email");
    ...
    115. "email=" + email + ", result=" + String.valueOf(status)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
    ...
    114. log.info(LogManager.getHeader(context, "failed_login",
```

[Table of Contents] [Description]

Path 25:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
```

```
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```



```
193. protected void doDSPost(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    225. .getRequestLogInfo(request)));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```



```
193. protected void doDSPost(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    224. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 26:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    271. "already_registered", "email=" + email));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```



```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    270. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 27:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    290. "sendtoken_register", "email=" + email));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    289. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 28:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    303. "email=" + email));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    301. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 29:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    257. String netid = request.getParameter("netid");
    ...
    351. "netid=" + netid + e));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    349. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 30:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    374. "email=" + email));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    373. log.info(LogManager.getHeader(context, "unknown_email",
```

[Table of Contents] [Description]

Path 31:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    385. "unregistered_forgot_password", "email=" + email));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    384. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 32:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    246. String email = request.getParameter("email");
    ...
    254. email = email.toLowerCase().trim();
    ...
    395. + email));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    393. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 33:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
     246. String email = request.getParameter("email");
     ...
     254. email = email.toLowerCase().trim();
     ...
     404. "sendtoken_forgotpw", "email=" + email));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

▼

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
     403. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 34:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
     246. String email = request.getParameter("email");
     ...
     254. email = email.toLowerCase().trim();
     ...
     419. + email));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

▼

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
     418. log.info(LogManager.getHeader(context, "bad_email", "email="
```

[Table of Contents] [Description]

Path 35:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
246. String email = request.getParameter("email");
...
254. email = email.toLowerCase().trim();
...
438. + email), me);
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
     ...
437. log.info(LogManager.getHeader(context, "error_emailing", "email="
```

[Table of Contents] [Description]

Path 36:

Query Name - **Log_Forgery**

Severity - **Warning**

```
454. private void processPersonalInfo(Context context, //registerservlet.java
     ...
460. String token = request.getParameter("token");
...
471. + token));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
454. private void processPersonalInfo(Context context, //registerservlet.java
     ...
470. log.info(LogManager.getHeader(context, "invalid_token", "token="
```

[Table of Contents] [Description]

Path 37:

Query Name - **Log_Forgery**

Severity - **Warning**

```
571. private void processNewPassword(Context context, //registerservlet.java
      ...
      577. String token = request.getParameter("token");
      ...
      586. + token));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
571. private void processNewPassword(Context context, //registerservlet.java
      ...
      585. log.info(LogManager.getHeader(context, "invalid_token", "token="
```

[Table of Contents] [Description]

Path 38:

Query Name - **Log_Forgery**

Severity - **Warning**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
      ...
      79. String idString = request.getPathInfo();
      ...
      86. idString = idString.substring(1);
      ...
      95. idString = idString.substring(0, slashIndex);
      ...
      134. "invalid_bitstream_id=" + idString));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //retrieveservlet.java
      ...
      133. log.info(LogManager.getHeader(context, "view_bitstream",
```

[Table of Contents] [Description]

Path 39:

Query Name - **Log_Forgery**

Severity - **Warning**

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
    ...
    96. String query = request.getParameter("query");
    ...
    293. + query + "\",results=(" + resultsCommunities.length + ","
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
    ...
    292. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Path 40:

Query Name - **Log_Forgery**

Severity - **Warning**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    232. String workspaceID = request.getParameter("resume");
    ...
    248. "bad_id=" + workspaceID));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    247. log.warn(LogManager.getHeader(context, "bad_workspace_id",
```

[Table of Contents] [Description]

Path 41:

Query Name - **Log_Forgery**

Severity - **Warning**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    256. String workflowID = request.getParameter("workflow");
    ...
    272. "bad_id=" + workflowID));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    271. log.warn(LogManager.getHeader(context, "bad_workflow_id",
```

[Table of Contents] [Description]

Path 42:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
    372. String name = (String) e.nextElement();
    ...
    378. report = report + "-- " + name + ": *not logged*\n";
    ...
    382. report = report + "-- " + name + ": \""
    ...
    387. return report;
```

```
320. protected void doDSPost(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    440. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
320. protected void doDSPost(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
439. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 43:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

▼

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1099. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

▼

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1098. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 44:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
```

```
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1178. private void processGetFileFormat(Context context, //submitservlet.java
...
1218. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1178. private void processGetFileFormat(Context context, //submitservlet.java
...
1217. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

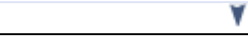
[Table of Contents] [Description]

Path 45:

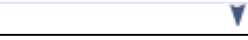
Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```



```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
...
1315. UIUtil.getRequestLogInfo(request)));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1314. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 46:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1346. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1345. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 47:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
```

```
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
...
1390. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
...
1389. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 48:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1448. private void processChangeFileDescription(Context
context, //submitservlet.java
...
1473. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1448. private void processChangeFileDescription(Context
context, //submitservlet.java
      ...
      1472. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 49:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1747. request = fur.getOriginalRequest();
      ...
      1784. .getRequestLogInfo(request)));
```

▼

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      362. report = "-- URL Was: " + getOriginalURL(request) + "\n";
```

▼

```
219. public static String getOriginalURL(HttpServletRequest request) //uiutil.java
      ...
      224. return ((String) request.getAttribute("dspace.original.url"));
```

▼

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      362. report = "-- URL Was: " + getOriginalURL(request) + "\n";
      363. report = report + "-- Method: " + request.getMethod() + "\n";
      ...
      366. report = report + "-- Parameters were:\n";
      ...
      378. report = report + "-- " + name + ": *not logged*\n";
      ...
      382. report = report + "-- " + name + ": \""
      ...
      387. return report;
```

▼

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1784. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1783. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 50:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1747. request = fur.getOriginalRequest();
      ...
      1790. doStep(context, request, response, subInfo, nextStep);
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1841. showFirstUploadPage(context, request, response, subInfo);
```

```
1936. private void showFirstUploadPage(Context context, //submitservlet.java
      1937. HttpServletRequest request, HttpServletResponse response,
      ...
      1951. doStep(context, request, response, subInfo, CHOOSE_FILE);
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1884. .getRequestLogInfo(request)));
```

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      362. report = "-- URL Was: " + getOriginalURL(request) + "\n";
```

```
219. public static String getOriginalURL(HttpServletRequest request) //uiutil.java
      ...
      224. return ((String) request.getAttribute("dspace.original.url"));
```

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      362. report = "-- URL Was: " + getOriginalURL(request) + "\n";
      363. report = report + "-- Method: " + request.getMethod() + "\n";
      ...
      366. report = report + "-- Parameters were:\n";
```

```
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
...
1884. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
...
1883. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 51:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
...
104. String path = request.getPathInfo();
105. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
...
105. log.info(LogManager.getHeader(context, "invalid_id", "path=" + path));
```

[Table of Contents] [Description]

Path 52:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
    ...
    161. String senderAddr = request.getParameter("sender_email");
    ...
    207. "from=" + senderAddr));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
    ...
    206. log.info(LogManager.getHeader(context, "sent_suggest",
```

[Table of Contents] [Description]

Path 53:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
    372. String name = (String) e.nextElement();
    ...
    378. report = report + "-- " + name + ": *not logged*\n";
    ...
    382. report = report + "-- " + name + ": \""
    ...
    387. return report;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    168. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
```

```
request, //collectionwizardservlet.java
    ...
    167. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 54:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
    372. String name = (String) e.nextElement();
    ...
    378. report = report + "-- " + name + ": *not logged*\n";
    ...
    382. report = report + "-- " + name + ": \""
    ...
    387. return report;
```

▼

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    201. UIUtil.getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

▼

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    200. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 55:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
    372. String name = (String) e.nextElement();
    ...
```

```
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    230. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    229. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

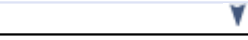
Path 56:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    368. Enumeration e = request.getParameterNames();
    ...
    372. String name = (String) e.nextElement();
    ...
    378. report = report + "-- " + name + ": *not logged*\n";
    ...
    382. report = report + "-- " + name + ": \""
    ...
    387. return report;
```

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
    462. .getRequestLogInfo(wrapper)));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
    ...
461. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 57:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

▼

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
    ...
289. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

▼

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
    ...
288. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 58:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
368. Enumeration e = request.getParameterNames();
...
```

```
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
171. protected void doDSPost(Context context, HttpServletRequest
request, //edititemservlet.java
...
276. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
171. protected void doDSPost(Context context, HttpServletRequest
request, //edititemservlet.java
...
275. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 59:

Query Name - **Log_Forgery**

Severity - **Warning**

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
...
119. dc.setElement(request.getParameter("element"));
```

```
173. public void setElement(String element) //metadatafield.java
...
175. this.element = element;
```

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
...
119. dc.setElement(request.getParameter("element"));
...
129. dc.update(context);
```

```
411. public void update(Context context) throws SQLException, //metadatafield.java
...
447. "metadata_field_id=" + getFieldID() + "element=" + getElement()
```

```
163. public String getElement() //metadatafield.java
      ...
      165. return element;
```

```
411. public void update(Context context) throws SQLException, //metadatafield.java
      ...
      447. "metadata_field_id=" + getFieldID() + "element=" + getElement()
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
411. public void update(Context context) throws SQLException, //metadatafield.java
      ...
      446. log.info(LogManager.getHeader(context, "update_metadatafieldregistry",
```

[Table of Contents] [Description]

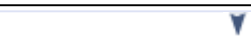
Path 60:

Query Name - **Log_Forgery**

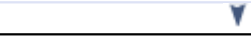
Severity - **Warning**

```
81. protected void doDSPost(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
      ...
      118. schema.setNamespace(request.getParameter("namespace"));
```

```
161. public void setNamespace(String namespace) //metadataschema.java
      ...
      163. this.namespace = namespace;
```



```
81. protected void doDSPost(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
      ...
      118. schema.setNamespace(request.getParameter("namespace"));
      ...
      120. schema.update(context);
```



```
291. public void update(Context context) throws SQLException, //metadataschema.java
      ...
      323. + getNamespace() + "name=" + getName()));
```

```
151. public String getNamespace() //metadataschema.java
      ...
      153. return namespace;
```

```
291. public void update(Context context) throws SQLException, //metadataschema.java
      ...
      323. + getNamespace() + "name=" + getName()));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
291. public void update(Context context) throws SQLException, //metadataschema.java
      ...
      321. log.info(LogManager.getHeader(context, "update_metadata_schema",
```

[Table of Contents] [Description]

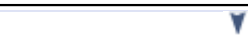
Path 61:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
      ...
      246. String email = request.getParameter("email");
      ...
      254. email = email.toLowerCase().trim();
      ...
      294. AccountManager.sendRegistrationInfo(context, email);
```

```
95. public static void sendRegistrationInfo(Context context, String
email) //accountmanager.java
      ...
      99. sendInfo(context, email, true, true);
```



```
221. protected static TableRow sendInfo(Context context, String
email, //accountmanager.java
      ...
      248. + " with email \"" + email + "\"");
```

[Table of Contents] [Description]

Path 62:

Query Name - **Log_Forgery**

Severity - **Warning**

```
242. private void processEnterEmail(Context context, HttpServletRequest
request, //registerservlet.java
      ...
      246. String email = request.getParameter("email");
```

```
...
254. email = email.toLowerCase().trim();
...
294. AccountManager.sendRegistrationInfo(context, email);
```



```
95. public static void sendRegistrationInfo(Context context, String
email) //accountmanager.java
...
99. sendInfo(context, email, true, true);
```



```
221. protected static TableRow sendInfo(Context context, String
email, //accountmanager.java
...
254. sendEmail(email, isRegister, rd);
```



```
278. private static void sendEmail(String email, boolean isRegister, TableRow
rd) //accountmanager.java
...
300. + " information to " + email);
```

[Table of Contents] [Description]

Path 63:

Query Name - **Log_Forgery**

Severity - **Warning**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
...
88. String email = request.getParameter("login_email");
...
93. int status = AuthenticationManager.authenticate(context, email, password,
```



```
131. public static int authenticate(Context context, //authenticationmanager.java
132. String username,
...
137. return authenticateInternal(context, username, password, realm,
```



```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
183. String username,
...
200. ret = methodStack[i].authenticate(context, username, password, realm,
request);
```



```
142. public int authenticate(Context context, //ldapauthentication.java
143. String netid,
...
149. log.info(LogManager.getHeader(context, "auth", "attempting trivial auth
of user="+netid));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      149. log.info(LogManager.getHeader(context, "auth", "attempting trivial auth
      of user="+netid));
```

[Table of Contents] [Description]

Path 64:

Query Name - **Log_Forgery**

Severity - **Warning**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
      ...
      88. String email = request.getParameter("login_email");
      ...
      93. int status = AuthenticationManager.authenticate(context, email, password,
```

```
131. public static int authenticate(Context context, //authenticationmanager.java
      132. String username,
      ...
      137. return authenticateInternal(context, username, password, realm,
```

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
      183. String username,
      ...
      200. ret = methodStack[i].authenticate(context, username, password, realm,
      request);
```

```
142. public int authenticate(Context context, //ldapauthentication.java
      143. String netid,
      ...
      191. "autoregister", "netid=" + netid));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
142. public int authenticate(Context context, //ldapauthentication.java
      ...
      190. log.info(LogManager.getHeader(context,
```

[Table of Contents] [Description]

Path 65:

Query Name - **Log_Forgery**

Severity - **Warning**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
    ...
    88. String email = request.getParameter("login_email");
    ...
    93. int status = AuthenticationManager.authenticate(context, email, password,
```

▼

```
131. public static int authenticate(Context context, //authenticationmanager.java
    132. String username,
    ...
    137. return authenticateInternal(context, username, password, realm,
```

▼

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
    183. String username,
    ...
    200. ret = methodStack[i].authenticate(context, username, password, realm,
    request);
```

▼

```
168. public int authenticate(Context context, //passwordauthentication.java
    169. String username,
    ...
    178. log.info(LogManager.getHeader(context, "authenticate", "attempting
    password auth of user="+username));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

▼

```
168. public int authenticate(Context context, //passwordauthentication.java
    ...
    178. log.info(LogManager.getHeader(context, "authenticate", "attempting
    password auth of user="+username));
```

[Table of Contents] [Description]

Path 66:

Query Name - **Log_Forgery**

Severity - **Warning**

```
83. protected void doDSPost(Context context, HttpServletRequest
request, //passwordservlet.java
```

```
...
88. String email = request.getParameter("login_email");
...
93. int status = AuthenticationManager.authenticate(context, email, password,
```

▼

```
131. public static int authenticate(Context context, //authenticationmanager.java
    132. String username,
    ...
    137. return authenticateInternal(context, username, password, realm,
```

▼

```
182. private static int authenticateInternal(Context
context, //authenticationmanager.java
    183. String username,
    ...
    200. ret = methodStack[i].authenticate(context, username, password, realm,
    request);
```

▼

```
168. public int authenticate(Context context, //passwordauthentication.java
    169. String username,
    ...
    199. log.warn(LogManager.getHeader(context, "authenticate", "rejecting
    PasswordAuthentication because "+username+" requires certificate."));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

▼

```
168. public int authenticate(Context context, //passwordauthentication.java
    ...
    199. log.warn(LogManager.getHeader(context, "authenticate", "rejecting
    PasswordAuthentication because "+username+" requires certificate."));
```

[Table of Contents] [Description]

Path 67:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
    ...
    83. String handle = request.getParameter("handle");
    ...
    171. String itemUrl = HandleManager.resolveToURL(context,handle);
```

▼

```
95. public static String resolveToURL(Context context, String
handle) //handlemanager.java
    ...
    106. + "/handle/" + handle;
```

```
...
110. log.debug("Resolved " + handle + " to " + url);
```

[Table of Contents] [Description]

Path 68:

Query Name - **Log_Forgery**

Severity - **Warning**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
    ...
92. String idString = request.getPathInfo();
    ...
108. idString = idString.substring(1);
    ...
118. handle = idString.substring(0, slashIndex);
    ...
138. DSpaceObject dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 69:

Query Name - **Log_Forgery**

Severity - **Warning**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
    ...
92. String idString = request.getPathInfo();
    ...
108. idString = idString.substring(1);
    ...
118. handle = idString.substring(0, slashIndex);
    ...
138. DSpaceObject dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 70:

Query Name - **Log_Forgery**

Severity - **Warning**

```
84. protected void doDSGet(Context context, HttpServletRequest
request, //bitstreamservlet.java
      ...
      92. String idString = request.getPathInfo();
      ...
      108. idString = idString.substring(1);
      ...
      118. handle = idString.substring(0, slashIndex);
      ...
      138. DSpaceObject dso = HandleManager.resolveToObject(context, handle);
```



```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 71:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      133. String focus = request.getParameter("focus");
      ...
      188. .resolveToObject(context, focus);
```



```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 72:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      133. String focus = request.getParameter("focus");
      ...
      188. .resolveToObject(context, focus);
```



```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 73:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      133. String focus = request.getParameter("focus");
      ...
      188. .resolveToObject(context, focus);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 74:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      135. String top = request.getParameter("top");
      ...
      311. .encode(top, Constants.DEFAULT_ENCODING)
      ...
      490. request.setAttribute("flip.ordering.query", flipOrderingQuery);
      491. JSPManager.showJSP(request, response,
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
```

```
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 75:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
    135. String top = request.getParameter("top");
    ...
    270. val = top;
    ...
    282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 76:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
    135. String top = request.getParameter("top");
    ...
    270. val = top;
    ...
    282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 77:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      135. String top = request.getParameter("top");
      ...
      270. val = top;
      ...
      282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 78:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      136. String bottom = request.getParameter("bottom");
      ...
      317. + URLEncoder.encode(bottom,
      ...
      490. request.setAttribute("flip.ordering.query", flipOrderingQuery);
      491. JSPManager.showJSP(request, response,
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 79:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      136. String bottom = request.getParameter("bottom");
      ...
      265. String val = bottom;
      ...
      282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 80:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      136. String bottom = request.getParameter("bottom");
      ...
      265. String val = bottom;
      ...
      282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 81:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
```

```
request, //browseservlet.java
    ...
136. String bottom = request.getParameter("bottom");
    ...
265. String val = bottom;
    ...
282. Item item = (Item) HandleManager.resolveToObject(context, val);
```

▼

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 82:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
139. String month = request.getParameter("month");
    ...
166. month = "0" + month;
    ...
169. startsWith = year + "-" + month;
    ...
224. + URLEncoder.encode(startsWith,
    ...
490. request.setAttribute("flip.ordering.query", flipOrderingQuery);
491. JSPManager.showJSP(request, response,
```

▼

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
80. contextExtraInfo = context.getExtraLogInfo();
```

▼

```
195. public String getExtraLogInfo() //context.java
    ...
197. return extraLogInfo;
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
80. contextExtraInfo = context.getExtraLogInfo();
    ...
87. String result = new String(email + ":" + contextExtraInfo + ":"
    ...
90. return result;
```

▼

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 83:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      140. String year = request.getParameter("year");
      ...
      158. startsWith = year;
      ...
      251. startsWith = startsWith + "-32";
      ...
      257. logInfo = "starts_with=" + startsWith + ",";
      ...
      333. logInfo = logInfo + ",collection_id=" + collection.getID() + ",";
      ...
      387. log.info(LogManager.getHeader(context, "browse_" + what, logInfo
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      387. log.info(LogManager.getHeader(context, "browse_" + what, logInfo
```

[Table of Contents] [Description]

Path 84:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      140. String year = request.getParameter("year");
      ...
      158. startsWith = year;
      ...
      224. + URLEncoder.encode(startsWith,
      ...
      490. request.setAttribute("flip.ordering.query", flipOrderingQuery);
      491. JSPManager.showJSP(request, response,
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```



```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```



```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 85:

Query Name - **Log_Forgery**

Severity - **Warning**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
      ...
      162. String path = request.getPathInfo();
      ...
      176. path = path.substring(1);
      ...
      181. handle = path.substring(split+1);
      ...
      192. dso = HandleManager.resolveToObject(context, handle);
```



```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 86:

Query Name - **Log_Forgery**

Severity - **Warning**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
      ...
      162. String path = request.getPathInfo();
      ...
      176. path = path.substring(1);
      ...
      181. handle = path.substring(split+1);
      ...
      192. dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 87:

Query Name - **Log_Forgery**

Severity - **Warning**

```
158. protected void doDSGet(Context context, HttpServletRequest
request, //feedservlet.java
      ...
      162. String path = request.getPathInfo();
      ...
      176. path = path.substring(1);
      ...
      181. handle = path.substring(split+1);
      ...
      192. dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 88:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
      ...
      102. String path = request.getPathInfo();
      ...
      107. path = path.substring(1);
      ...
      118. handle = path.substring(0, secondSlash);
```

```
...
136. dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
...
247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 89:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
...
102. String path = request.getPathInfo();
...
107. path = path.substring(1);
...
118. handle = path.substring(0, secondSlash);
...
136. dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
...
259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 90:

Query Name - **Log_Forgery**

Severity - **Warning**

```
92. protected void doDSGet(Context context, HttpServletRequest
request, //handleservlet.java
...
102. String path = request.getPathInfo();
...
107. path = path.substring(1);
...
118. handle = path.substring(0, secondSlash);
...
136. dso = HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
...
271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 91:

Query Name - **Log_Forgery**

Severity - **Warning**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
    ...
    152. String idString = request.getPathInfo();
    ...
    163. idString = idString.substring(1);
    ...
    173. handle = idString.substring(0, slashIndex);
    ...
    213. .resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 92:

Query Name - **Log_Forgery**

Severity - **Warning**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
    ...
    152. String idString = request.getPathInfo();
    ...
    163. idString = idString.substring(1);
    ...
    173. handle = idString.substring(0, slashIndex);
    ...
    213. .resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

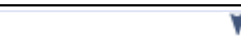
Path 93:

Query Name - **Log_Forgery**

Severity - **Warning**

```
145. protected void doDSGet(Context context, HttpServletRequest
request, //htmlservlet.java
    ...
    152. String idString = request.getPathInfo();
```

```
...
163. idString = idString.substring(1);
...
173. handle = idString.substring(0, slashIndex);
...
213. .resolveToObject(context, handle);
```



```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    271. log.debug("Resolved handle " + handle + " to community "
```

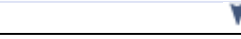
[Table of Contents] [Description]

Path 94:

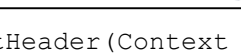
Query Name - **Log_Forgery**

Severity - **Warning**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbyauthorservlet.java
    ...
    83. String author = request.getParameter("author");
    ...
    132. request.setAttribute("author", author);
    133. request.setAttribute("order.by.title", new Boolean(orderByTitle));
    134. request.setAttribute("browse.info", browseInfo);
    ...
    136. JSPManager.showJSP(request, response, "/browse/items-by-author.jsp");
```

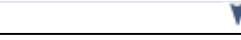


```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

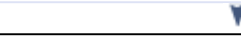


```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
    ...
    197. return extraLogInfo;
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
    ...
    87. String result = new String(email + ":" + contextExtraInfo + ":"
    ...
    90. return result;
```



```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
```

```
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 95:

Query Name - **Log_Forgery**

Severity - **Warning**

```
71. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbysubjectservlet.java
    ...
    82. String subject = request.getParameter("subject");
    ...
    131. request.setAttribute("subject", subject);
    132. request.setAttribute("order.by.title", new Boolean(orderByTitle));
    133. request.setAttribute("browse.info", browseInfo);
    ...
    135. JSPManager.showJSP(request, response, "/browse/items-by-subject.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
    ...
    197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
    ...
    87. String result = new String(email + ":" + contextExtraInfo + ":"
    ...
    90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 96:

Query Name - **Log_Forgery**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
```

```
request, //openurlservlet.java
    ...
74. String authorFirst = request.getParameter("aufirst");
    ...
87. query = query + " " + authorFirst;
    ...
93. query = query + " " + authorLast;
    ...
98. + "dspacequery=" + query));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    97. log.info(LogManager.getHeader(context, "openURL", logInfo
```

[Table of Contents] [Description]

Path 97:

Query Name - **Log_Forgery**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    75. String authorLast = request.getParameter("aulast");
    ...
    93. query = query + " " + authorLast;
    ...
    98. + "dspacequery=" + query));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    97. log.info(LogManager.getHeader(context, "openURL", logInfo
```

[Table of Contents] [Description]

Path 98:

Query Name - **Log_Forgery**

Severity - **Warning**

```
116. protected void doDSGet(Context context, HttpServletRequest
request, //registerservlet.java
      ...
      129. String token = request.getParameter("token");
      ...
      163. request.setAttribute("token", token);
      ...
      170. request.setAttribute("set.password", new Boolean(setPassword));
      ...
      173. JSPManager.showJSP(request, response,
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 99:

Query Name - **Log_Forgery**

Severity - **Warning**

```
571. private void processNewPassword(Context context, //registerservlet.java
      ...
      577. String token = request.getParameter("token");
      ...
      619. request.setAttribute("token", token);
      620. request.setAttribute("eperson", eperson);
      ...
      622. JSPManager.showJSP(request, response, "/register/new-password.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 100:

Query Name - **Log_Forgery**

Severity - **Warning**

```
121. private void showStatistics(Context context, //statisticsservlet.java
      ...
      125. String date = (String) request.getParameter("date");
      126. request.setAttribute("date", date);
      ...
      128. request.setAttribute("general", new Boolean(false));
      ...
      151. request.setAttribute("general", new Boolean(true));
      ...
      203. JSPManager.showJSP(request, response, "statistics/no-report.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
```

```
...
197. return extraLogInfo;
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
...
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```



```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
...
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 101:

Query Name - **Log_Forgery**

Severity - **Warning**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
...
232. String workspaceID = request.getParameter("resume");
...
249. JSPManager.showInvalidIDError(request, response, workspaceID,
```



```
162. public static void showInvalidIDError(HttpServletRequest
request, //jspmanager.java
163. HttpServletResponse response, String badID, int type)
...
166. request.setAttribute("bad.id", badID);
...
171. request.setAttribute("bad.type", new Integer(type));
...
174. showJSP(request, response, "/error/invalid-id.jsp");
```



```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
...
87. .getAttribute("dspace.context"), "view_jsp", jsp));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
```



```
195. public String getExtraLogInfo() //context.java
...
197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 102:

Query Name - **Log_Forgery**

Severity - **Warning**

```
211. protected void doDSGet(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      256. String workflowID = request.getParameter("workflow");
      ...
      274. .showInvalidIDError(request, response, workflowID, -1);
```

```
162. public static void showInvalidIDError(HttpServletRequest
request, //jspmanager.java
      163. HttpServletResponse response, String badID, int type)
      ...
      166. request.setAttribute("bad.id", badID);
      ...
      171. request.setAttribute("bad.type", new Integer(type));
      ...
      174. showJSP(request, response, "/error/invalid-id.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
```

```
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 103:

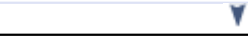
Query Name - **Log_Forgery**

Severity - **Warning**

```
462. private void processSelectCollection(Context context, //submitservlet.java
    ...
    488. .getParameter("collection"), Constants.COLLECTION);
```

```
162. public static void showInvalidIDError(HttpServletRequest
request, //jspmanager.java
    163. HttpServletResponse response, String badID, int type)
    ...
    166. request.setAttribute("bad.id", badID);
    ...
    171. request.setAttribute("bad.type", new Integer(type));
    ...
    174. showJSP(request, response, "/error/invalid-id.jsp");
```

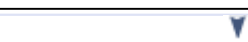
```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    87. .getAttribute("dspace.context"), "view_jsp", jsp));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
    ...
    197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
    ...
    87. String result = new String(email + ":" + contextExtraInfo + ":"
    ...
    90. return result;
```



```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
```

```
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 104:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1747. request = fur.getOriginalRequest();
      ...
      1785. JSPManager.showIntegrityError(request, response);
```

▼

```
118. public static void showIntegrityError(HttpServletRequest
request, //jspmanager.java
      ...
      122. showJSP(request, response, "/error/integrity.jsp");
```

▼

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

▼

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

▼

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 105:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
        ...
        83. String handle = request.getParameter("handle");
        ...
        90. Item item = (Item) HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
        ...
        247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 106:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
        ...
        83. String handle = request.getParameter("handle");
        ...
        90. Item item = (Item) HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
        ...
        259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 107:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
        ...
        83. String handle = request.getParameter("handle");
        ...
        90. Item item = (Item) HandleManager.resolveToObject(context, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
        ...
        271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 108:

Query Name - **Log_Forgery**

Severity - **Warning**

```
78. protected void doDSGet(Context context, HttpServletRequest
request, //suggestservlet.java
      ...
      104. String path = request.getPathInfo();
      ...
      106. JSPManager.showInvalidIDError(request, response, path, -1);
```

```
162. public static void showInvalidIDError(HttpServletRequest
request, //jspmanager.java
      163. HttpServletResponse response, String badID, int type)
      ...
      166. request.setAttribute("bad.id", badID);
      ...
      171. request.setAttribute("bad.type", new Integer(type));
      ...
      174. showJSP(request, response, "/error/invalid-id.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 109:

Query Name - **Log_Forgery**

Severity - **Warning**

```
90. protected void doDSPost(Context c, HttpServletRequest
request, //authorizeadminservlet.java
      ...
      138. String handle = request.getParameter("handle");
      ...
      148. DSpaceObject dso = HandleManager.resolveToObject(c, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 110:

Query Name - **Log_Forgery**

Severity - **Warning**

```
90. protected void doDSPost(Context c, HttpServletRequest
request, //authorizeadminservlet.java
      ...
      138. String handle = request.getParameter("handle");
      ...
      148. DSpaceObject dso = HandleManager.resolveToObject(c, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 111:

Query Name - **Log_Forgery**

Severity - **Warning**

```
90. protected void doDSPost(Context c, HttpServletRequest
request, //authorizeadminservlet.java
      ...
      138. String handle = request.getParameter("handle");
      ...
      148. DSpaceObject dso = HandleManager.resolveToObject(c, handle);
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
      ...
      271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 112:

Query Name - **Log_Forgery**

Severity - **Warning**

```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
      220. response.sendRedirect(response.encodeRedirectURL(request
      ...
      400. String side = request.getParameter("side_bar_text");
      ...
      409. community.setMetadata("side_bar_text", side);
```

```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
      ...
      355. communityRow.setColumn(field, value);
```

```
412. public void setColumn(String column, String s) //tablerow.java
      ...
      419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```

```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
      ...
      355. communityRow.setColumn(field, value);
```

```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
      ...
      409. community.setMetadata("side_bar_text", side);
      410. community.update();
```

```
427. public void update() throws SQLException, IOException,
AuthorizeException //community.java
      ...
      441. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
```

```
force) //dsindexer.java
     '
```



```
353. public String getHandle() //collection.java
      |
      |...
      |355. return handle;
```



```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      |
      |...
      |230. String handle = dso.getHandle();
      |...
      |256. log.info("Wrote Item: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 113:

Query Name - **Log_Forgery**

Severity - **Warning**

```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
      |220. response.sendRedirect(response.encodeRedirectURL(request
      |...
      |400. String side = request.getParameter("side_bar_text");
      |...
      |409. community.setMetadata("side_bar_text", side);
```



```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
      |...
      |355. communityRow.setColumn(field, value);
```



```
412. public void setColumn(String column, String s) //tablerow.java
      |...
      |419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```



```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
      |...
      |355. communityRow.setColumn(field, value);
```



```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
      |...
      |409. community.setMetadata("side_bar_text", side);
      |410. community.update();
```



```
427. public void update() throws SQLException, IOException,
```

```
AuthorizeException //community.java
      ...
      441. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      263. log.info("Wrote Collection: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 114:

Query Name - **Log_Forgery**

Severity - **Warning**

```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
      220. response.sendRedirect(response.encodeRedirectURL(request
      ...
      400. String side = request.getParameter("side_bar_text");
      ...
      409. community.setMetadata("side_bar_text", side);
```

```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
      ...
      355. communityRow.setColumn(field, value);
```



```
412. public void setColumn(String column, String s) //tablerow.java
```

```
...
419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```

```
342. public void setMetadata(String field, String value)throws
MissingResourceException //community.java
...
355. communityRow.setColumn(field, value);
```

```
352. private void processConfirmEditCommunity(Context
context, //editcommunitiesservlet.java
...
409. community.setMetadata("side_bar_text", side);
410. community.update();
```

```
427. public void update() throws SQLException, IOException,
AuthorizeException //community.java
...
441. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
...
355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
...
230. String handle = dso.getHandle();
...
269. log.info("Wrote Community: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 115:

Query Name - **Log_Forgery**

Severity - **Warning**

```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      ...
      506. String license = request.getParameter("license");
      ...
      523. collection.setMetadata("license", license);
```

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
      ...
      399. collectionRow.setColumn(field, value);
```

```
412. public void setColumn(String column, String s) //tablerow.java
      ...
      419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
      ...
      399. collectionRow.setColumn(field, value);
```

```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      ...
      523. collection.setMetadata("license", license);
      ...
      553. collection.update();
```

```
850. public void update() throws SQLException, IOException,
AuthorizeException //collection.java
      ...
      865. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      256. log.info("Wrote Item: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 116:

Query Name - **Log_Forgery**

Severity - **Warning**

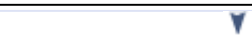
```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      ...
      506. String license = request.getParameter("license");
      ...
      523. collection.setMetadata("license", license);
```

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
      ...
      399. collectionRow.setColumn(field, value);
```

```
412. public void setColumn(String column, String s) //tablerow.java
      ...
      419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
      ...
      399. collectionRow.setColumn(field, value);
```

```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      ...
      523. collection.setMetadata("license", license);
      ...
      553. collection.update();
```



```
850. public void update() throws SQLException, IOException,
AuthorizeException //collection.java
      ...
      865. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      263. log.info("Wrote Collection: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 117:

Query Name - **Log_Forgery**

Severity - **Warning**

```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      ...
      506. String license = request.getParameter("license");
      ...
      523. collection.setMetadata("license", license);
```

```
386. public void setMetadata(String field, String value) throws
MissingResourceException //collection.java
      ...
      399. collectionRow.setColumn(field, value);
```

```
412. public void setColumn(String column, String s) //tablerow.java
      ...
      419. data.put(canonicalize(column), (s == null) ? NULL_OBJECT : s);
```

```
386. public void setMetadata(String field, String value) throws
```

```
MissingResourceException //collection.java
      |...
      |399. collectionRow.setColumn(field, value);
```

```
467. private void processConfirmEditCollection(Context
context, //editcommunitiesservlet.java
      |...
      |523. collection.setMetadata("license", license);
      |...
      |553. collection.update();
```

```
850. public void update() throws SQLException, IOException,
AuthorizeException //collection.java
      |...
      |865. DSIndexer.reIndexContent(ourContext, this);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      |...
      |360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      |...
      |212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      '
```

```
353. public String getHandle() //collection.java
      |...
      |355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      |...
      |230. String handle = dso.getHandle();
      |...
      |269. log.info("Wrote Community: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 118:

Query Name - **Log_Forgery**

Severity - **Warning**

```
112. protected void doDSGet(Context context, HttpServletRequest
```

```
request, //edititemservlet.java
    ...
    124. String handle = request.getParameter("handle");
    ...
    139. DSpaceObject dso = HandleManager.resolveToObject(context, handle.trim());
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    247. log.debug("Resolved handle " + handle + " to item "
```

[Table of Contents] [Description]

Path 119:

Query Name - **Log_Forgery**

Severity - **Warning**

```
112. protected void doDSGet(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    124. String handle = request.getParameter("handle");
    ...
    139. DSpaceObject dso = HandleManager.resolveToObject(context, handle.trim());
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    259. log.debug("Resolved handle " + handle + " to collection "
```

[Table of Contents] [Description]

Path 120:

Query Name - **Log_Forgery**

Severity - **Warning**

```
112. protected void doDSGet(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    124. String handle = request.getParameter("handle");
    ...
    139. DSpaceObject dso = HandleManager.resolveToObject(context, handle.trim());
```

```
221. public static DSpaceObject resolveToObject(Context context, String
handle) //handlemanager.java
    ...
    271. log.debug("Resolved handle " + handle + " to community "
```

[Table of Contents] [Description]

Path 121:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      442. String language = request.getParameter("language_" + key + "_"
      ...
      460. item.addMetadata(schema, element, qualifier, language, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      ...
      638. dcv.language = lang;
      ...
      660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      460. item.addMetadata(schema, element, qualifier, language, value);
      ...
      602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      256. log.info("Wrote Item: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 122:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      442. String language = request.getParameter("language_" + key + "_"
      ...
      460. item.addMetadata(schema, element, qualifier, language, value);
```

▼

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      ...
      638. dcv.language = lang;
      ...
      660. dublinCore.add(dcv);
```

▼

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      460. item.addMetadata(schema, element, qualifier, language, value);
      ...
      602. DSIndexer.reIndexContent(context, item);
```

▼

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

▼

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

▼

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

▼

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

▼

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      263. log.info("Wrote Collection: " + handle + " to Index");
```

Path 123:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
442. String language = request.getParameter("language_" + key + "_"
...
460. item.addMetadata(schema, element, qualifier, language, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
    ...
638. dcv.language = lang;
...
660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
460. item.addMetadata(schema, element, qualifier, language, value);
...
602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
    ...
355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
    ...
```

```
230. String handle = dso.getHandle();
...
269. log.info("Wrote Community: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 124:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    442. String language = request.getParameter("language_" + key + "_"
    ...
    460. item.addMetadata(schema, element, qualifier, language, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
    ...
    638. dcv.language = lang;
    ...
    660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    460. item.addMetadata(schema, element, qualifier, language, value);
    ...
    602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
    360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
    212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
    ...
    248. Document doc = buildDocument(context, (Item) dso);
```

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
    ...
    871. String handle = item.getHandle();
```

```
313. public String getHandle() //item.java
        ...
        315. return handle;
```

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
        ...
        871. String handle = item.getHandle();
        ...
        883. log.debug("Building Item: " + handle);
```

[Table of Contents] [Description]

Path 125:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
        ...
        563. String value = request.getParameter("addfield_value").trim();
        ...
        576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
        628. String[] values)
        ...
        642. String temp = values[i].trim();
        643. char[] dcvalue = temp.toCharArray();
        ...
        654. dcv.value = String.valueOf(dcvalue);
        ...
        660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
        ...
        575. item.addMetadata(schema.getName(), field.getElement(), field
        ...
        602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
        ...
        360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
        ...
        212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
     ...
355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
     ...
230. String handle = dso.getHandle();
     ...
256. log.info("Wrote Item: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 126:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
     ...
563. String value = request.getParameter("addfield_value").trim();
     ...
576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
     628. String[] values)
     ...
     642. String temp = values[i].trim();
     643. char[] dcvalue = temp.toCharArray();
     ...
     654. dcv.value = String.valueOf(dcvalue);
     ...
     660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
     ...
575. item.addMetadata(schema.getName(), field.getElement(), field
     ...
602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
     ...
360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```



```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```



```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```



```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      263. log.info("Wrote Collection: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 127:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      563. String value = request.getParameter("addfield_value").trim();
      ...
      576. .getQualifier(), lang, value);
```



```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      628. String[] values)
      ...
      642. String temp = values[i].trim();
      643. char[] dcvalue = temp.toCharArray();
      ...
      654. dcv.value = String.valueOf(dcvalue);
      ...
      660. dublinCore.add(dcv);
```



```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      575. item.addMetadata(schema.getName(), field.getElement(), field
      ...
      602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      269. log.info("Wrote Community: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 128:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      564. String lang = request.getParameter("addfield_language");
      ...
      576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      ...
      638. dcv.language = lang;
      ...
      660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      575. item.addMetadata(schema.getName(), field.getElement(), field
```

```
...
583. request.setAttribute("item", item);
...
591. request.setAttribute("item", item);
...
593. .showJSP(request, response, "/tools/upload-bitstream.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
...
87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
...
197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
...
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
...
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 129:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
...
564. String lang = request.getParameter("addfield_language");
...
576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
...
638. dcv.language = lang;
...
660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      575. item.addMetadata(schema.getName(), field.getElement(), field
      ...
      602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
      ...
      355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      230. String handle = dso.getHandle();
      ...
      256. log.info("Wrote Item: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 130:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      564. String lang = request.getParameter("addfield_language");
      ...
      576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      ...
      638. dcv.language = lang;
```

```
...
660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
...
575. item.addMetadata(schema.getName(), field.getElement(), field
...
602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
...
212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
...
355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
...
230. String handle = dso.getHandle();
...
263. log.info("Wrote Collection: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 131:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
...
564. String lang = request.getParameter("addfield_language");
...
576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
```

```
lang, //item.java
    ...
    638. dcv.language = lang;
    ...
    660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    575. item.addMetadata(schema.getName(), field.getElement(), field
    ...
    602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
    360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
    ...
    212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
```

```
353. public String getHandle() //collection.java
    ...
    355. return handle;
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
    ...
    230. String handle = dso.getHandle();
    ...
    269. log.info("Wrote Community: " + handle + " to Index");
```

[Table of Contents] [Description]

Path 132:

Query Name - **Log_Forgery**

Severity - **Warning**

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
    ...
    564. String lang = request.getParameter("addfield_language");
    ...
    576. .getQualifier(), lang, value);
```

```
627. public void addMetadata(String schema, String element, String qualifier, String
lang, //item.java
      ...
      638. dcv.language = lang;
      ...
      660. dublinCore.add(dcv);
```

```
379. private void processUpdateItem(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      575. item.addMetadata(schema.getName(), field.getElement(), field
      ...
      602. DSIndexer.reIndexContent(context, item);
```

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      360. indexContent(context, dso);
```

```
209. public static void indexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      212. indexContent(context, dso, false);
```

```
226. public static void indexContent(Context context, DSpaceObject dso, boolean
force) //dsindexer.java
      ...
      248. Document doc = buildDocument(context, (Item) dso);
```

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      ...
      871. String handle = item.getHandle();
```

```
313. public String getHandle() //item.java
      ...
      315. return handle;
```

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
      ...
      871. String handle = item.getHandle();
      ...
      883. log.debug("Building Item: " + handle);
```

[Table of Contents] [Description]

Path 133:

Query Name - **Log_Forgery**

Severity - **Warning**

```
79. protected void doDSPost(Context context, HttpServletRequest
request, //itemmapservlet.java
    ...
    337. String myQuery = (String) request.getParameter("namepart");
    ...
    366. request.setAttribute("browsetext", myQuery);
    367. request.setAttribute("items", items);
    368. request.setAttribute("browsetype", new String("Add"));
    ...
    371. JSPManager.showJSP(request, response, jspPage);
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
    ...
    197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    ...
    80. contextExtraInfo = context.getExtraLogInfo();
    ...
    87. String result = new String(email + ":" + contextExtraInfo + ":"
    ...
    90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
    ...
    86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 134:

Query Name - **Log_Forgery**

Severity - **Warning**

```
82. protected void doDSPost(Context c, HttpServletRequest
request, //licenseeditservlet.java
    ...
    108. String license = (String)request.getParameter("license");
    ...
    131. request.setAttribute("license", license);
    ...
    134. request.setAttribute("edited", "true");
    ...
    137. JSPManager.showJSP(request, response, "/dspace-admin/license-edit.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
      ...
      197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      ...
      80. contextExtraInfo = context.getExtraLogInfo();
      ...
      87. String result = new String(email + ":" + contextExtraInfo + ":"
      ...
      90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
      ...
      86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 135:

Query Name - **Log_Forgery**

Severity - **Warning**

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
      ...
      121. String qual = request.getParameter("qualifier");
      ...
      127. dc.setQualifier(qual);
```

```
203. public void setQualifier(String qualifier) //metadatafield.java
      ...
      205. this.qualifier = qualifier;
```

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
      ...
      127. dc.setQualifier(qual);
      ...
      129. dc.update(context);
```

```
411. public void update(Context context) throws SQLException, //metadatafield.java
```

```
...
448. + "qualifier=" + getQualifier()));
```



```
193. public String getQualifier() //metadatafield.java
...
195. return qualifier;
```



```
411. public void update(Context context) throws SQLException, //metadatafield.java
...
448. + "qualifier=" + getQualifier()));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
411. public void update(Context context) throws SQLException, //metadatafield.java
...
446. log.info(LogManager.getHeader(context, "update_metadatafieldregistry",
```

[Table of Contents] [Description]

Path 136:

Query Name - **Log_Forgery**

Severity - **Warning**

```
81. protected void doDSPost(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
...
119. schema.setName(request.getParameter("short_name"));
```



```
181. public void setName(String name) //metadataschema.java
...
183. this.name = name;
```



```
81. protected void doDSPost(Context context, HttpServletRequest
request, //metadataschemaregistryservlet.java
...
119. schema.setName(request.getParameter("short_name"));
120. schema.update(context);
```



```
291. public void update(Context context) throws SQLException, //metadataschema.java
...
323. + getNamespace() + "name=" + getName()));
```



```
171. public String getName() //metadataschema.java
```

```
...
173. return name;
```

```
291. public void update(Context context) throws SQLException, //metadataschema.java
...
323. + getNamespace() + "name=" + getName()));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
291. public void update(Context context) throws SQLException, //metadataschema.java
...
321. log.info(LogManager.getHeader(context, "update_metadata_schema",
```

[Table of Contents] [Description]

Path 137:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
...
1784. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1783. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 138:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
368. Enumeration e = request.getParameterNames();
...
372. String name = (String) e.nextElement();
...
378. report = report + "-- " + name + ": *not logged*\n";
...
382. report = report + "-- " + name + ": \""
...
387. return report;
```

▼

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1884. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

▼

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1883. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 139:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

▼

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
      ...
      141. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
      ...
      140. log.debug(LogManager.getHeader(context, "http_request", UIUtil
```

[Table of Contents] [Description]

Path 140:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```

```
69. protected void doGet(HttpServletRequest request, //internalerrorservlet.java
      ...
      76. String logInfo = UIUtil.getRequestLogInfo(request);
      ...
      81. + ":internal_error:" + logInfo;
      ...
      83. log.warn(logMessage, t);
```

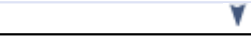
[Table of Contents] [Description]

Path 141:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```



```
102. protected void doDSPost(Context context, HttpServletRequest
request, //mydspaceservlet.java
      |
```

```
...
138. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
102. protected void doDSPost(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
    137. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 142:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    383. + request.getParameter(name) + "\"\n";
    ...
    387. return report;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
    221. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
    ...
    220. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 143:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      275. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      274. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 144:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
      ...
      312. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
290. private void processRemoveItem(Context context, HttpServletRequest
request, //mydspaceservlet.java
```

```
...
311. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 145:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

▼

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
...
369. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

▼

```
346. private void processPreviewTask(Context context, //mydspaceservlet.java
...
368. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 146:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

▼

```
403. private void processPerformTask(Context context, //mydspaceservlet.java
...
426. .getRequestLogInfo(request)));
```

▼

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
```

```
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
403. private void processPerformTask(Context context, //mydspaceservlet.java
...
425. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 147:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```



```
512. private void processRejectReason(Context context, //mydspaceservlet.java
...
535. .getRequestLogInfo(request)));
```



```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```



```
512. private void processRejectReason(Context context, //mydspaceservlet.java
...
534. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 148:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```



```
193. protected void doDSPost(Context context, HttpServletRequest
```

```
request, //registerservlet.java
    ...
225. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
193. protected void doDSPost(Context context, HttpServletRequest
request, //registerservlet.java
    ...
    224. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 149:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
    ...
    383. + request.getParameter(name) + "\"\n";
    ...
    387. return report;
```

```
320. protected void doDSPost(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    440. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
    66. String extrainfo)
    ...
    88. + action + ":" + extrainfo);
    ...
    90. return result;
```

```
320. protected void doDSPost(Context context, HttpServletRequest
request, //submitservlet.java
    ...
    439. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 150:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
     383. + request.getParameter(name) + "\"\n";
     ...
     387. return report;
```

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
     ...
     1099. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
     ...
     1098. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 151:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
     383. + request.getParameter(name) + "\"\n";
     ...
     387. return report;
```

```
1178. private void processGetFileFormat(Context context, //submitservlet.java
     ...
     1218. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

```
1178. private void processGetFileFormat(Context context, //submitservlet.java
```

```
...
1217. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 152:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1315. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1314. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 153:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
```

```
...
1346. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1345. log.warn(LogManager.getHeader(context, "integrity_error",
```

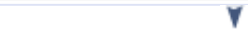
[Table of Contents] [Description]

Path 154:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```



```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1390. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
1235. private void processFileList(Context context, HttpServletRequest
request, //submitservlet.java
      ...
      1389. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 155:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
```

```
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
1448. private void processChangeFileDescription(Context
context, //submitservlet.java
...
1473. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1448. private void processChangeFileDescription(Context
context, //submitservlet.java
...
1472. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 156:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
...
1784. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
1733. private void doStepJump(Context context, HttpServletRequest
request, //submitservlet.java
...
1783. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 157:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
     383. + request.getParameter(name) + "\"\n";
     ...
     387. return report;
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
     ...
     1884. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
     ...
     88. + action + ":" + extrainfo);
     ...
     90. return result;
```

```
1811. private void doStep(Context context, HttpServletRequest
request, //submitservlet.java
     ...
     1883. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 158:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
     ...
     383. + request.getParameter(name) + "\"\n";
     ...
     387. return report;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
     ...
     168. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
     66. String extrainfo)
```

```
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
167. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 159:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
201. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
200. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 160:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
```

```
387. return report;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
230. UIUtil.getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
130. protected void doDSPost(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
229. log.warn(LogManager.getHeader(context, "integrity_error",
```

[Table of Contents] [Description]

Path 161:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
...
383. + request.getParameter(name) + "\"\n";
...
387. return report;
```

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
462. .getRequestLogInfo(wrapper)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
461. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 162:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      289. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      288. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 163:

Query Name - **Log_Forgery**

Severity - **Warning**

```
358. public static String getRequestLogInfo(HttpServletRequest
request) //uiutil.java
      ...
      383. + request.getParameter(name) + "\"\n";
      ...
      387. return report;
```

```
171. protected void doDSPost(Context context, HttpServletRequest
request, //edititemservlet.java
      ...
      276. .getRequestLogInfo(request)));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
```

```
90. return result;
```

```
171. protected void doDSPost(Context context, HttpServletRequest
request, //edititemservlet.java
...
275. log.warn(LogManager.getHeader(context, "integrity_error", UIUtil
```

[Table of Contents] [Description]

Path 164:

Query Name - **Log_Forgery**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
...
403. String val = request.getParameter(param);
...
407. return Integer.parseInt(val.trim());
```

```
90. protected void doDSPost(Context c, HttpServletRequest
request, //authorizeadminservlet.java
...
695. Collection collection = Collection.find(c, UIUtil.getIntParameter(
...
699. request.setAttribute("collection", collection);
700. request.setAttribute("policies", policies);
701. JSPManager.showJSP(request, response,
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
...
87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
...
197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
...
80. contextExtraInfo = context.getExtraLogInfo();
...
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
```

```
...
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 165:

Query Name - **Log_Forgery**

Severity - **Warning**

```
488. public static String getSubmitButton(HttpServletRequest request, String
def) //uiutil.java
```

```
...
490. Enumeration e = request.getParameterNames();
...
494. String parameterName = (String) e.nextElement();
...
498. return parameterName;
```

```
515. private void processInitialQuestions(Context context, //submitservlet.java
```

```
...
631. request.setAttribute("button.pressed", UIUtil.getSubmitButton(
...
634. JSPManager.showJSP(request, response, "/submit/verify-prune.jsp");
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
```

```
...
87. .getAttribute("dspace.context"), "view_jsp", jsp));
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
...
80. contextExtraInfo = context.getExtraLogInfo();
```

```
195. public String getExtraLogInfo() //context.java
```

```
...
197. return extraLogInfo;
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
```

```
...
80. contextExtraInfo = context.getExtraLogInfo();
...
87. String result = new String(email + ":" + contextExtraInfo + ":"
...
90. return result;
```

```
80. public static void showJSP(HttpServletRequest request, //jspmanager.java
```

```
...
86. log.debug(LogManager.getHeader((Context) request
```

[Table of Contents] [Description]

Path 166:

Query Name - **Log_Forgery**

Severity - **Warning**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
      ...
      177. request.getParameter("conjunction"+i):"AND");
      ...
      183. Iterator iconj = conjunction.iterator();
      ...
      193. conj_curr = " " + (String)iconj.next() + " ";
      ...
      197. newquery = newquery + ")";
      198. return (newquery);
```

```
149. private void doSearch(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
      ...
      181. query = qArgs.buildQuery(request);
      ...
      322. + query + "\",results=(" + resultsCommunities.length + ","
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
      66. String extrainfo)
      ...
      88. + action + ":" + extrainfo);
      ...
      90. return result;
```

```
149. private void doSearch(Context context, HttpServletRequest
request, //controlledvocabularysearchservlet.java
      ...
      321. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Path 167:

Query Name - **Log_Forgery**

Severity - **Warning**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
      ...
      177. request.getParameter("conjunction"+i):"AND");
      ...
      183. Iterator iconj = conjunction.iterator();
      ...
      193. conj_curr = " " + (String)iconj.next() + " ";
      ...
      197. newquery = newquery + ")";
      198. return (newquery);
```

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
      ...
```

```
124. query = qArgs.buildQuery(request);
...
293. + query + "\",results=(" + resultsCommunities.length + ","
```

```
65. public static String getHeader(Context context, String action, //logmanager.java
66. String extrainfo)
...
88. + action + ":" + extrainfo);
...
90. return result;
```

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
...
292. log.info(LogManager.getHeader(context, "search", logInfo + "query=\""
```

[Table of Contents] [Description]

Improper Resource Shutdown or Release

CWE ID	404
Description	The program can potentially fail to release or incorrectly release a system resource. A resource is not properly cleared and made available for re-use.
Functional Area	Non-specific
Potential Mitigations	It is good practice to be responsible for freeing all resources you allocate. Memory should be allocated/freed using matching functions such as malloc/free, new/delete, and new[]/delete[].
Demonstrative Examples	The following method never closes the file handle it opens. The Finalize() method for StreamReader eventually calls Close (), but there is no guarantee as to how long it will take before the Finalize() method is invoked. In fact, there is no guarantee that Finalize() will ever be invoked. In a busy environment, this can result in the VM using up all of its available file handles.

```
private void processFile(string fName) {
  StreamWriter sw = new
  StreamWriter(fName);
  string line;
  while ((line = sr.ReadLine()) != null) processLine(line);
}
```

If an exception occurs after establishing the database connection and before the same connection closes, the pool of database connections may become exhausted. If the number of available connections is exceeded, other users cannot access this resource, effectively denying access to the application. Using the following database connection pattern

will ensure that all opened connections are closed. The con.close() call should be the first executable statement in the finally block.

```
try {
  Connection con = DriverManager.getConnection
(some_connection_string)
}
catch ( Exception e ) {
  log( e )
}
finally {
  con.close()
}
```

Under normal conditions the following C# code executes a database query, processes the results returned by the database, and closes the allocated SqlConnection object. But if an exception occurs while executing the SQL or processing the results, the SqlConnection object is not closed. If this happens often enough, the database will run out of available cursors and not be able to execute any more SQL queries.

C# Example:

```
...
SqlConnection conn = new SqlConnection(connString);
SqlCommand cmd = new SqlCommand(queryString);
cmd.Connection = conn; conn.Open();
SqlDataReader rdr = cmd.ExecuteReader();
HarvestResults(rdr);
conn.Connection.Close();
...
```

The following C function does not close the file handle it opens if an error occurs. If the process is long-lived, the process can run out of file handles.

C Example:

```
int decodeFile(char* fName) {
  char buf[BUF_SZ];
  FILE* f = fopen(fName, "r");
  if (!f) {
    printf("cannot open %s\n", fName);
    return DECODE_FAIL;
  }
  else {
```

```
    while (fgets(buf, BUF_SZ, f)) {
      if (!checkChecksum(buf)) {
        return DECODE_FAIL;
      }
      else {
        decodeBlock(buf);
      }
    }
  }
  fclose(f);
  return DECODE_SUCCESS;
}
```

In this example, the program fails to use matching functions such as malloc/free, new/delete, and new[]/delete[] to allocate/deallocate the resource.

C++ Example:

```
class A{
  void foo();
};
void A::foo(){
  int *ptr;
  ptr = (int*)malloc(sizeof(int));
  delete ptr;
}
```

In this example, the program calls the delete[] function on non-heap memory.

C++ Example:

```
class A{
  void foo(bool);
};
void A::foo(bool heap) {
  int localArray[2] = {11,22};
  int *p = localArray;
  if (heap){
    p = new int[2];
  }
  delete[] p;
}
```

Observed Examples

CVE-1999-1127 - Does not shut down named pipe connections if malformed data is sent.

CVE-2001-0830 - Sockets not properly closed when attacker repeatedly connects and disconnects from server.

CVE-2002-1372 - Return values of file/socket operations not checked, allowing resultant consumption of file descriptors.

Context Notes

It is important to be consistent with how and where you free memory in a function. If you allocate memory that you intend to free upon completion of the function, you must be sure to free the memory at all exit points for that function including error conditions.

Before freeing a pointer, the programmer should make sure that the pointer was previously allocated and that the memory belongs to the programmer. Freeing an unallocated pointer will cause undefined behavior in the program.

Can be resultant from improper error handling or insufficient resource tracking.

Overlaps memory leaks, asymmetric resource consumption, malformed input errors.

Most unreleased resource issues result in general software reliability problems, but if an attacker can intentionally trigger a resource leak, the attacker might be able to launch a denial of service attack by depleting the resource pool. Resource leaks have at least two common causes: - Error conditions and other exceptional circumstances. - Confusion over which part of the program is responsible for releasing the resource.

Node Relationships

ChildOf - Resource Management Errors (399)
Peer - Memory Leak (401)
Peer - Asymmetric Resource Consumption (Amplification) (405)

Source Taxonomies

PLOVER - Improper resource shutdown or release

7 Pernicious Kingdoms - Unreleased Resource

Applicable Platforms

All

Path 1:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
68. public static void main(String[] argv) //createadministrator.java
      ...
71. BufferedReader input = null;
```

[Table of Contents] [Description]

Path 2:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
      ...
```

```
258. FileReader fr = null;
```

[Table of Contents] [Description]

Path 3:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
247. public static void processReport(Context context, String
myFormat, //reportgenerator.java
     ...
259. BufferedReader br = null;
```

[Table of Contents] [Description]

Path 4:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
547. public static void readMap(String map) //reportgenerator.java
     ...
550. FileReader fr = null;
```

[Table of Contents] [Description]

Path 5:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
547. public static void readMap(String map) //reportgenerator.java
     ...
551. BufferedReader br = null;
```

[Table of Contents] [Description]

Path 6:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
121. private void showStatistics(Context context, //statisticsservlet.java
     ...
135. FileInputStream fir = null;
```

[Table of Contents] [Description]

Path 7:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
121. private void showStatistics(Context context, //statisticsservlet.java
```

```
...
136. InputStreamReader ir = null;
```

[Table of Contents] [Description]

Path 8:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
121. private void showStatistics(Context context, //statisticsservlet.java
...
137. BufferedReader br = null;
```

[Table of Contents] [Description]

Path 9:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
103. public static ResultsPruner getPruner(String propsFile) //resultspruner.java
...
107. FileInputStream fin = null;
```

[Table of Contents] [Description]

Path 10:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
564. public static void checkConfiguration() //pluginmanager.java
...
582. BufferedReader cr = new BufferedReader(new FileReader(config));
```

[Table of Contents] [Description]

Path 11:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
1031. private static String getTool() //historymanager.java
...
1038. BufferedReader reader = new BufferedReader(new StringReader(
```

[Table of Contents] [Description]

Path 12:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
867. private static Document buildDocument(Context context, Item
item) //dsindexer.java
        ...
1050. InputStreamReader is = new InputStreamReader(
```

[Table of Contents] [Description]

Path 13:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
1049. public static void loadSql(Reader r) throws SQLException,
IOException //databasemanager.java
        ...
1051. BufferedReader reader = new BufferedReader(r);
```

[Table of Contents] [Description]

Unsafe URL Redirection

CWE ID	601
Description	An http parameter may contain a URL value and could cause the web application to redirect the request to the specified URL. By modifying the URL value to a malicious site, an attacker may successfully launch a phishing scam and steal user credentials. The fact that the server name in the modified link is identical to the original site helps the attacker by giving his phishing attempts a more reliable appearance.
Observed Examples	- CVE-2005-4206 - Blackboard Learning and Community Portal System in Academic Suite 6.3.1.424, 6.2.3.23, and other versions before 6 allows remote attackers to redirect users to other URLs and conduct phishing attacks via a modified url parameter to frameset.jsp, which loads the URL into a frame and causes it to appear to be part of a valid page.
Context Notes	Phishing is a general term for deceptive attempts to coerce private information from users that will be used for identity theft.
Node Relationships	ChildOf - API Abuse (227)
Source Taxonomies	Anonymous Tool Vendor (under NDA) -

Path 1:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
57. protected void doDSGet(Context context, HttpServletRequest
request, //controlledvocabularyservlet.java
        ...
64. String callerUrl = request.getParameter("callerUrl");
...
75. response.sendRedirect(callerUrl);
```

[Table of Contents] [Description]

Path 2:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
553. private boolean handleButton(HttpServletRequest request, //handleservlet.java
        ...
        557. String location = request.getParameter("location");
        ...
        573. prefix = "/handle/" + location + "/";
        ...
        579. url = request.getContextPath() + prefix + "browse-title";
        ...
        614. response.sendRedirect(response.encodeRedirectURL(url));
```

[Table of Contents] [Description]

Path 3:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
148. protected void processMainPage(Context context, HttpServletRequest
request, //mydspaceservlet.java
        ...
        200. String wsID = request.getParameter("workspace_id");
        ...
        203. + "/submit?resume=" + wsID));
```

[Table of Contents] [Description]

Path 4:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
        ...
        73. String title = request.getParameter("title");
        ...
        81. query = query + " " + title;
        ...
        102. + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 5:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
        ...
```

```
96. String query = request.getParameter("query");
...
151. query = URLEncoder.encode(query, Constants.DEFAULT_ENCODING);
...
155. query = query + "&from_advanced=true&" + advancedQuery;
...
161. + url + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 6:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
...
403. String val = request.getParameter(param);
...
407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
...
153. .getIntParameter(request, "collection_id"));
...
158. request.setAttribute("collection", collection);
...
221. .getContextPath()
```

[Table of Contents] [Description]

Path 7:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
...
403. String val = request.getParameter(param);
...
407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
...
153. .getIntParameter(request, "collection_id"));
...
158. request.setAttribute("collection", collection);
...
246. .getContextPath()
```

[Table of Contents] [Description]

Path 8:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
      ...
      403. String val = request.getParameter(param);
      ...
      407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      148. Community community = Community.find(context, UIUtil.getIntParameter(
```

```
133. public static Community find(Context context, int id) throws
SQLException //community.java
      ...
      137. .fromCache(Community.class, id);
```

```
289. public Object fromCache(Class objectClass, int id) //context.java
      ...
      291. String key = objectClass.getName() + id;
      ...
      293. return objectCache.get(key);
```

```
133. public static Community find(Context context, int id) throws
SQLException //community.java
      ...
      137. .fromCache(Community.class, id);
      ...
      141. return fromCache;
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      148. Community community = Community.find(context, UIUtil.getIntParameter(
      ...
      237. Community parent = community.getParentCommunity();
```

```
544. public Community getParentCommunity() throws SQLException //community.java
      ...
      571. parentCommunity = new Community(ourContext, row);
```

```
100. Community(Context context, TableRow row) throws SQLException //community.java
      ...
      102. ourContext = context;
```

```
544. public Community getParentCommunity() throws SQLException //community.java
      ...
      571. parentCommunity = new Community(ourContext, row);
      ...
      577. return parentCommunity;
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      237. Community parent = community.getParentCommunity();
      ...
      254. + "/handle/" + parent.getHandle()));
```

```
309. public String getHandle() //community.java
      ...
      311. return handle;
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      254. + "/handle/" + parent.getHandle()));
```

[Table of Contents] [Description]

Path 9:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
      ...
      403. String val = request.getParameter(param);
      ...
      407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      153. .getIntParameter(request, "collection_id"));
      ...
      158. request.setAttribute("collection", collection);
      ...
      166. showControls(context, request, response);
```

```
304. private void showControls(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      316. .getContextPath()
```

[Table of Contents] [Description]

Path 10:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
      ...
      403. String val = request.getParameter(param);
      ...
      407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      153. .getIntParameter(request, "collection_id"));
      ...
      158. request.setAttribute("collection", collection);
      ...
      166. showControls(context, request, response);
```

```
304. private void showControls(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      327. .getContextPath()
```

[Table of Contents] [Description]

Path 11:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
401. public static int getIntParameter(HttpServletRequest request, String
param) //uiutil.java
      ...
      403. String val = request.getParameter(param);
      ...
      407. return Integer.parseInt(val.trim());
```

```
121. protected void doDSPost(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      153. .getIntParameter(request, "collection_id"));
      ...
      158. request.setAttribute("collection", collection);
      ...
      166. showControls(context, request, response);
```

```
304. private void showControls(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
      ...
      334. .getContextPath()
```

[Table of Contents] [Description]

Path 12:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
93. protected void doDSPost(Context context, HttpServletRequest
request, //metadatafieldregistryservlet.java
    ...
    209. .getParameter("dc_dest_schema_id"));
    ...
    234. + "/dspace-admin/metadata-schema-registry?dc_schema_id=" + schemaID);
```

[Table of Contents] [Description]

Path 13:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
553. private boolean handleButton(HttpServletRequest request, //handleservlet.java
    ...
    607. + URLEncoder.encode(request.getParameter("query"),
    ...
    614. response.sendRedirect(response.encodeRedirectURL(url));
```

[Table of Contents] [Description]

Path 14:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    74. String authorFirst = request.getParameter("aufirst");
    ...
    87. query = query + " " + authorFirst;
    ...
    102. + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 15:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
66. protected void doDSGet(Context context, HttpServletRequest
request, //openurlservlet.java
    ...
    75. String authorLast = request.getParameter("aulast");
    ...
    93. query = query + " " + authorLast;
    ...
```

```
102. + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 16:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
    ...
135. String location = request.getParameter("location");
    ...
147. url = "/handle/" + location;
    ...
161. + url + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 17:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
142. public String buildQuery(HttpServletRequest request) //queryargs.java
    ...
177. request.getParameter("conjunction"+i):"AND");
    ...
183. Iterator iconj = conjunction.iterator();
    ...
193. conj_curr = " " + (String)iconj.next() + " ";
    ...
197. newquery = newquery + ")";
198. return (newquery);
```

▼

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
    ...
124. query = qArgs.buildQuery(request);
    ...
151. query = URLEncoder.encode(query, Constants.DEFAULT_ENCODING);
    ...
155. query = query + "&from_advanced=true&" + advancedQuery;
    ...
161. + url + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 18:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
240. public HashMap buildQueryHash(HttpServletRequest request) //queryargs.java
    ...
264. : request.getParameter("field"+numField));
```

```
...
266. return (queryHash);
```

```
291. public String buildHTTPQuery(HttpServletRequest request) //queryargs.java
...
295. HashMap queryHash = buildQueryHash(request);
...
297. Iterator i = queryHash.keySet().iterator();
...
301. String key = (String) i.next();
...
304. querystring = querystring + "&" + key + "="
...
309. querystring = querystring + "&num_search_field="+request.getParameter
("num_search_field");
...
313. return (querystring.substring(1));
```

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
...
125. advancedQuery = qArgs.buildHTTPQuery(request);
...
155. query = query + "&from_advanced=true&" + advancedQuery;
...
161. + url + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Path 19:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
291. public String buildHTTPQuery(HttpServletRequest request) //queryargs.java
...
309. querystring = querystring + "&num_search_field="+request.getParameter
("num_search_field");
...
313. return (querystring.substring(1));
```

```
91. protected void doDSGet(Context context, HttpServletRequest
request, //simplesearchservlet.java
...
125. advancedQuery = qArgs.buildHTTPQuery(request);
...
155. query = query + "&from_advanced=true&" + advancedQuery;
...
161. + url + "/simple-search?query=" + query));
```

[Table of Contents] [Description]

Object Model Violation: Just One of Equals and Haschode Defined

CWE ID	581
Description	Software fails to maintain equal hashcodes for equal

	objects.
Common Consequences	Failure to uphold this invariant is likely to cause trouble if objects of this class are stored in a collection. If the objects of the class in question are used as a key in a Hashtable or if they are inserted into a Map or Set, it is critical that equal objects have equal hashcodes.
Context Notes	Java objects are expected to obey a number of invariants related to equality. One of these invariants is that equal objects must have equal hashcodes. In other words, if a.equals(b) == true then a.hashCode() == b.hashCode().
Node Relationships	ChildOf - Failure to Follow Specification (573)
Applicable Platforms	Java

Path 1:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
82. public class Collection extends DSpaceObject //collection.java
      |
```

[Table of Contents] [Description]

Path 2:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
75. public class Community extends DSpaceObject //community.java
      |
```

[Table of Contents] [Description]

Path 3:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
87. public class Item extends DSpaceObject //item.java
      |
```

[Table of Contents] [Description]

Path 4:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
61. public class ItemComparator implements Comparator //itemcomparator.java
      |
```

[Table of Contents] [Description]

Path 5:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
69. public class Group extends DSpaceObject //group.java
```

[Table of Contents] [Description]

	Improper Transaction Handling
CWE ID	10016
Description	If an exception occurred during a database transaction, a rollback should be committed.
Likelihood of Exploit	Low
Common Consequences	Database lock.
Potential Mitigations	Perform rollback in the catch clause.
Applicable Platforms	All

Path 1:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
177. public void update(BitstreamInfo info) //bitstreaminfodao.java
       ...
188. try
```

[Table of Contents] [Description]

Path 2:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
278. public void updateMissingBitstreams() //bitstreaminfodao.java
       ...
283. try
```

[Table of Contents] [Description]

Path 3:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
354. public int deleteBitstreamInfoWithHistory(int id) //bitstreaminfodao.java
       ...
359. try
```

[Table of Contents] [Description]

Path 4:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
75. public void insertHistory(BitstreamInfo info) //checksumhistorydao.java
      |...
      86. try
```

[Table of Contents] [Description]

Path 5:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
121. protected int deleteHistoryForBitstreamInfo(int id, Connection
conn) //checksumhistorydao.java
      |...
      127. try
```

[Table of Contents] [Description]

Path 6:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
216. public int prune(Map interests) //checksumhistorydao.java
      |...
      219. try
```

[Table of Contents] [Description]

Path 7:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
136. public WorkspaceItem ingest(Context context, Collection
collection, //pdfpackager.java
      |...
      171. try
```

[Table of Contents] [Description]

Path 8:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
209. public void complete() throws SQLException //context.java
      |...
      213. try
```

[Table of Contents] [Description]

Path 9:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
142. public int authenticate(Context context, //ldapauthentication.java
       ...
195. try
```

[Table of Contents] [Description]

Path 10:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
142. public int authenticate(Context context, //ldapauthentication.java
       ...
215. try
```

[Table of Contents] [Description]

Path 11:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
380. public int authenticate(Context context, //x509authentication.java
       ...
398. try
```

[Table of Contents] [Description]

Path 12:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
594. public static void cleanup(boolean deleteDbRecords) throws SQLException,
IOException //bitstreamstoragemanager.java
       ...
600. try
```

[Table of Contents] [Description]

Error Message Information Leaks

CWE ID	209
Description	Server messages need to be parsed before being passed on to the user.
Likelihood of Exploit	High
Common Consequences	Confidentiality: Often this will either reveal sensitive information which may be used for a later attack or private

	information stored in the server.
Potential Mitigations	Design: When an application detects illegal input, error messages should only provide generic feedback, such as "Illegal characters were detected." Messages should provide few, if any, implementation details. Implementation: Any error should be parsed for dangerous revelations. Build: Debugging information should not make its way into a production release. Handle exceptions internally and do not display errors containing potentially sensitive information to a user. Create default error pages if necessary.
Demonstrative Examples	**Java Example:** try { /.../ } catch (Exception e) { System.out.println(e); }
Context Notes	Here you are passing much more data than is needed. Another example is passing the SQL exceptions to a WebUser without filtering. Error messages should not provide attackers with any implementation details when the application detects an illegal action. This includes indicating exactly what is allowable, or exactly what was illegal about the user input. Such detailed information can help an attacker craft another attack that now will pass through the validation filters. The first thing an attacker may use -- once an attack has failed -- to stage the next attack is the error information provided by the server. SQL Injection attacks generally probe the server for information in order to stage a successful attack.
Node Relationships	ChildOf - Information Leak (Information Disclosure) (200) ParentOf - Product-Generated Error Message Information Leak (210) ParentOf - Product-External Error Message Information Leak (211)
Source Taxonomies	CLASP - Accidental leaking of sensitive information through error messages
Applicable Platforms	All

Path 1:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
68. public static void main(String[] argv) //createadministrator.java
      ...
180. e.printStackTrace();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
63. public static void main(String[] argv) //upgrade101to11.java
      ...
119. e.printStackTrace();
```

[Table of Contents] [Description]

Path 3:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
108. public static void main(String[] args) //checksumchecker.java
       ...
169. e.printStackTrace();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
139. public static void main(String[] argv) throws Exception //itemimport.java
       ...
440. e.printStackTrace();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
312. public static boolean filterBitstream(Context c, Item
myItem, //mediafiltermanager.java
       ...
338. e.printStackTrace();
```

[Table of Contents] [Description]

Path 6:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
323. public static void writeMETS(Context context, Item item, OutputStream os,
boolean fullURL) //metsexport.java
       |
```

```
...
538. e.printStackTrace();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
...
244. ex.printStackTrace();
```

[Table of Contents] [Description]

Path 8:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
...
264. System.err.println("Caught exception:"+sqle.getCause());
265. sqle.printStackTrace();
```

[Table of Contents] [Description]

Path 9:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
174. public String createMetadata(Object nativeItem) //plugincrosswalk.java
...
190. e.toString());
...
193. e.printStackTrace(new PrintWriter(sw));
```

[Table of Contents] [Description]

Path 10:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
130. public static void main(String[] argv) throws Exception //packager.java
...
308. e.printStackTrace();
```

[Table of Contents] [Description]

Path 11:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
229. public String renderVocabularyAsHTML(Document
vocabulary, //controlledvocabularytag.java
    ...
249. e.printStackTrace();
```

[Table of Contents] [Description]

Path 12:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //index.jsp
    ...
88. se.toString()), se);
...
91. UIUtil.sendAlert(request, se);
```

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
    ...
549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1887. public static void updateIndexData(BrowseScope key) //browse.java
    ...
1925. e.printStackTrace();
```

[Table of Contents] [Description]

Path 14:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
65. public static void main(String[] argv) //initializebrowse.java
    ...
87. sqle.printStackTrace();
```

[Table of Contents] [Description]

Path 15:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
781. public static void main(String[] argv) //configurationmanager.java
      |...
      |793. warn("Error installing configuration files", ie);
```



```
832. private static void warn(String string, Exception
e) //configurationmanager.java
      |...
      |837. e.printStackTrace();
```

[Table of Contents] [Description]

Path 16:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
457. public static void loadConfig(String configFile) //configurationmanager.java
      |...
      |567. fatal("Can't load configuration", e);
```



```
869. private static void fatal(String string, Exception
e) //configurationmanager.java
      |...
      |874. e.printStackTrace();
```

[Table of Contents] [Description]

Path 17:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
247. public void abort() //context.java
      |...
      |255. log.error(se.getMessage());
      |256. se.printStackTrace();
```

[Table of Contents] [Description]

Path 18:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1115. public static void main(String[] argv) //historymanager.java
      |...
      |1174. e.printStackTrace();
```

[Table of Contents] [Description]

Path 19:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
296. public static void unIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      305. log.error(exception.getMessage(),exception);
      306. emailException(exception);
```

```
668. private static void emailException(Exception exception) //dsindexer.java
      ...
      686. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
727. public static void freeConnection(Connection c) //databasemanager.java
      ...
      738. log.warn(e.getMessage());
      739. e.printStackTrace();
```

[Table of Contents] [Description]

Path 21:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //index.jsp
      ...
      91. UIUtil.sendAlert(request, se);
```

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
      549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 22:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //index.jsp
      ...
      91. se.toString()), se);
      ...
      94. UIUtil.sendAlert(request, se);
```

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
      549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //index.jsp
      ...
      94. UIUtil.sendAlert(request, se);
```



```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
      549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 24:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //formats.jsp
      ...
      83. se.toString()), se);
      ...
      86. UIUtil.sendAlert(request, se);
```



```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
      549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 25:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1. <%-- //formats.jsp
      ...
      86. UIUtil.sendAlert(request, se);
```



```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
```

```
...
549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 26:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
83. public String createMetadata(Object nativeItem) //didlcrosswalk.java
...
265. sqle.printStackTrace();
```

[Table of Contents] [Description]

Path 27:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
174. public String createMetadata(Object nativeItem) //plugincrosswalk.java
...
193. e.printStackTrace(new PrintWriter(sw));
```

[Table of Contents] [Description]

Path 28:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
...
158. log.warn(LogManager.getHeader(context, "database_error", se
...
162. UIUtil.sendAlert(request, se);
```

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
...
549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 29:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
116. private void processRequest(HttpServletRequest request, //dspaceservlet.java
...
162. UIUtil.sendAlert(request, se);
```

```
523. public static void sendAlert(HttpServletRequest request, Exception
exception) //uiutil.java
      ...
      549. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 30:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
247. public void abort() //context.java
      ...
      256. se.printStackTrace();
```

[Table of Contents] [Description]

Path 31:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
296. public static void unIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      306. emailException(exception);
```

```
668. private static void emailException(Exception exception) //dsindexer.java
      ...
      686. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 32:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
      364. log.error(exception.getMessage(),exception);
      365. emailException(exception);
```

```
668. private static void emailException(Exception exception) //dsindexer.java
      ...
      686. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 33:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
355. public static void reIndexContent(Context context, DSpaceObject
dso) //dsindexer.java
      ...
365. emailException(exception);
```

```
668. private static void emailException(Exception exception) //dsindexer.java
      ...
686. exception.printStackTrace(pw);
```

[Table of Contents] [Description]

Path 34:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
727. public static void freeConnection(Connection c) //databasemanager.java
      ...
739. e.printStackTrace();
```

[Table of Contents] [Description]

Resource Injection

CWE ID	99
Description	Allowing user input to control resource identifiers may enable an attacker to access or modify otherwise protected system resources.
Likelihood of Exploit	High
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Demonstrative Examples	The following Java code uses input from an HTTP request to create a file name. The programmer has not considered the possibility that an attacker could provide a file name such as "../../tomcat/conf/server.xml", which causes the application to delete one of its own configuration files.

Java Example:

```
String rName = request.getParameter("reportName");
File rFile = new File("/usr/local/apfr/reports/" + rName);
...
rFile.delete();
```

The following code uses input from the command line to determine which file to open and echo back to the user. If the program runs with privileges and malicious users can create soft links to the file, they can use the program to read the first part of any file on the system.

C++ Example:

```
ifstream ifs(argv[0]);
string s;
ifs >> s;
cout << s;
```

The kind of resource the data affects indicates the kind of content that may be dangerous. For example, data containing special characters like period, slash, and backslash, are risky when used in methods that interact with the file system. (Resource injection, when it is related to file system resources, sometimes goes by the name "path manipulation.") Similarly, data that contains URLs and URIs is risky for functions that create remote connections.

Context Notes

A resource injection issue occurs when the following two conditions are met: 1. An attacker can specify the identifier used to access a system resource. For example, an attacker might be able to specify part of the name of a file to be opened or a port number to be used. 2. By specifying the resource, the attacker gains a capability that would not otherwise be permitted. For example, the program may give the attacker the ability to overwrite the specified file, run with a configuration controlled by the attacker, or transmit sensitive information to a third-party server. Note: Resource injection that involves resources stored on the filesystem goes by the name path manipulation and is reported in separate category. See the path manipulation description for further details of this vulnerability.

Node Relationships

ChildOf - Injection (74)
CanAlsoBe - Path Manipulation (73)
ChildOf - Weaknesses Examined by SAMATE (630)

Source Taxonomies 7 Pernicious Kingdoms - Resource Injection

Applicable Platforms All

Path 1:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
998. private void processChooseFile(Context context, HttpServletRequest
request, //submitservlet.java
     |
     |...
     1028. temp = wrapper.getFile("file");
```

```
...
1030. if (temp != null && temp.length() > 0)
...
1034. new FileInputStream(temp));
```

[Table of Contents] [Description]

Path 2:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
449. private void processBasicInfo(Context context, HttpServletRequest
request, //collectionwizardservlet.java
...
490. File temp = wrapper.getFile("file");
...
495. InputStream is = new BufferedInputStream(new FileInputStream(temp));
```

[Table of Contents] [Description]

Path 3:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
684. private void processUploadLogo(Context context, HttpServletRequest
request, //editcommunitiesservlet.java
...
696. File temp = wrapper.getFile("file");
...
699. InputStream is = new BufferedInputStream(new FileInputStream(temp));
```

[Table of Contents] [Description]

Path 4:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
618. private void processUploadBitstream(Context context, //edititemservlet.java
...
630. File temp = wrapper.getFile("file");
...
633. InputStream is = new BufferedInputStream(new FileInputStream(temp));
```

[Table of Contents] [Description]

Web Parameter Tampering

CWE ID	472
Description	A web product does not properly protect assumed-immutable values from modification in hidden form fields, parameters, cookies, or URLs, which lead to modification of critical data.
Alternate Terms	Assumed-Immutable Parameter Tampering

Potential Mitigations Do not pass hidden fields or cookies (apart from a session cookie) to the server. And, do not use parameters that are not from immediate user input.

Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.

Observed Examples CVE-2002-0108 - Forum product allows spoofed messages of other users via hidden form fields for name and e-mail address.

CVE-2000-0253 - Shopping cart allows price modification via hidden form field.

CVE-2000-0254 - Shopping cart allows price modification via hidden form field.

CVE-2000-0926 - Shopping cart allows price modification via hidden form field.

CVE-2000-0101 - Shopping cart allows price modification via hidden form field.

CVE-2000-0102 - Shopping cart allows price modification via hidden form field.

CVE-2000-0758 - Allows admin access by modifying value of form field.

CVE-2002-1880 - Read messages by modifying message ID parameter.

CVE-2000-1234 - Send email to arbitrary users by modifying email parameter.

CVE-2005-1652 - Authentication bypass by setting a parameter.

CVE-2005-1784 - Product does not check authorization for configuration change admnin script, leading to password theft via modified e-mail address field.

CVE-2005-2314 - Logic error leads to password disclosure.

CVE-2005-1682 - Modification of message number parameter allows attackers to read other people's messages.

Context Notes This is a primary weakness for many other weaknesses and functional consequences, including XSS, SQL injection, path disclosure, and file inclusion. For example, custom cookies commonly store session data or persistent data across sessions. This kind of session data is normally involved in security related decisions on the server side, such as user authentication and access control. Thus, the cookies might contain sensitive data such as user credentials and privileges. This is a dangerous practice, as it can often lead to improper

reliance on the value of the client-provided cookie by the server side application. Without appropriate protection mechanisms, the client can easily tamper with cookies. Reliance on the cookies without detailed validation can lead to problems such as SQL injection. If you use cookie values for security related decisions on the server side, manipulating the cookies might lead to violations of security policies such as authentication bypassing, user impersonation and privilege escalation. In addition, storing sensitive data in the cookie without appropriate protection can also lead to disclosure of sensitive user data, especially data stored in persistent cookies.

Hidden fields should not be trusted as secure parameters. An attacker can intercept and alter hidden fields in a post to the server as easily as user input fields. An attacker can simply parse the HTML for the substring < input type "hidden" or even just "hidden". Hidden field values displayed later in the session, such as on the following page, can open a site up to cross-site scripting attacks.

This is a technology-specific MAID problem.

Node Relationships ChildOf - Modification of Assumed-Immutable Data (MAID) (471)
ChildOf - Weaknesses in OWASP Top Ten (629)
ParentOf - Unchecked Array Indexing (129)

Source Taxonomies PLOVER - Web Parameter Tampering

Applicable Platforms All

Path 1:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      133. String focus = request.getParameter("focus");
      ...
      182. scope.setFocus(focus);
```

```
202. public void setFocus(String value) //browsescope.java
      ...
      204. focus = value.toLowerCase();
```

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      182. scope.setFocus(focus);
      ...
      357. browseInfo = Browse.getSubjects(scope);
```

```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
```

```
SQLException //browse.java
    ...
147. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
    ...
717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
    ...
952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
    ...
293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
    ...
952. statement.setString(1, (String) scope.getFocus());
...
960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
    ...
1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
133. String focus = request.getParameter("focus");
...
182. scope.setFocus(focus);
```

```
202. public void setFocus(String value) //browsescope.java
    ...
204. focus = value.toLowerCase();
```

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
182. scope.setFocus(focus);
```

```
...
357. browseInfo = Browse.getSubjects(scope);
```

```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
...
147. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
...
710. results.addAll(getResultsBeforeFocus(scope, itemValue));
```

```
813. protected static List getResultsBeforeFocus(BrowseScope scope, //browse.java
...
837. PreparedStatement statement = createSql(scope, itemValue, false, false);
```

```
1132. private static PreparedStatement createSql(BrowseScope scope, //browse.java
...
1146. value = (String)scope.getFocus();
```

```
291. public Object getFocus() //browsescope.java
...
293. return focus;
```

```
1132. private static PreparedStatement createSql(BrowseScope scope, //browse.java
...
1146. value = (String)scope.getFocus();
...
1149. statement.setString(1, value);
...
1163. return statement;
```

```
813. protected static List getResultsBeforeFocus(BrowseScope scope, //browse.java
...
837. PreparedStatement statement = createSql(scope, itemValue, false, false);
...
839. List qresults = DatabaseManager.queryPrepared(statement).toList();
```

```
351. public static TableRowIterator queryPrepared(PreparedStatement
statement) //databasemanager.java
...
354. TableRowIterator retTRI = new TableRowIterator(statement.executeQuery());
```

[Table of Contents] [Description]

Path 3:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
    134. String startsWith = request.getParameter("starts_with");
    ...
    255. scope.setFocus(startsWith);
```

```
202. public void setFocus(String value) //browsescope.java
    ...
    204. focus = value.toLowerCase();
```

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
    ...
    255. scope.setFocus(startsWith);
    ...
    357. browseInfo = Browse.getSubjects(scope);
```

```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
    ...
    147. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
    ...
    717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
    ...
    952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
    ...
    293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
    ...
    952. statement.setString(1, (String) scope.getFocus());
    ...
    960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
    ...
    1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      135. String top = request.getParameter("top");
      ...
      270. val = top;
      ...
      277. scope.setFocus(val);
```

```
202. public void setFocus(String value) //browsescope.java
      ...
      204. focus = value.toLowerCase();
```

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
      ...
      277. scope.setFocus(val);
      ...
      298. scope.setNumberBefore(isTop ? 0 : 20);
      ...
      357. browseInfo = Browse.getSubjects(scope);
```

```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
      ...
      147. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
      ...
      717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
      ...
      952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
      ...
      293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
      ...
      952. statement.setString(1, (String) scope.getFocus());
      ...
      960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
        ...
1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
        ...
136. String bottom = request.getParameter("bottom");
...
265. String val = bottom;
...
277. scope.setFocus(val);
```

▼

```
202. public void setFocus(String value) //browsescope.java
        ...
204. focus = value.toLowerCase();
```

▼

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
        ...
277. scope.setFocus(val);
...
298. scope.setNumberBefore(isTop ? 0 : 20);
...
357. browseInfo = Browse.getSubjects(scope);
```

▼

```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
        ...
147. return doBrowse(scope);
```

▼

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
        ...
717. int total = countTotalInIndex(scope, results.size());
```

▼

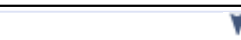
```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
        ...
952. statement.setString(1, (String) scope.getFocus());
```

▼

```
291. public Object getFocus() //browsescope.java
        ...
293. return focus;
```

▼

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
        ...
        952. statement.setString(1, (String) scope.getFocus());
        ...
        960. return getIntValue(statement);
```



```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
        ...
        1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 6:

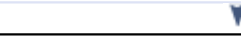
Query Name - **Parameter_Tampering**

Severity - **Serious**

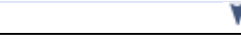
```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
        ...
        139. String month = request.getParameter("month");
        ...
        166. month = "0" + month;
        ...
        169. startsWith = year + "-" + month;
        ...
        255. scope.setFocus(startsWith);
```

```
202. public void setFocus(String value) //browsescope.java
        ...
        204. focus = value.toLowerCase();
```

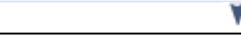
```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
        ...
        255. scope.setFocus(startsWith);
        ...
        357. browseInfo = Browse.getSubjects(scope);
```



```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
        ...
        147. return doBrowse(scope);
```



```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
        ...
        717. int total = countTotalInIndex(scope, results.size());
```



```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
        ...
```

```
952. statement.setString(1, (String) scope.getFocus());
```



```
291. public Object getFocus() //browsescope.java
     ...
     293. return focus;
```



```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
     ...
     952. statement.setString(1, (String) scope.getFocus());
     ...
     960. return getIntValue(statement);
```



```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
      ...
      1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
     ...
     140. String year = request.getParameter("year");
     ...
     158. startsWith = year;
     ...
     255. scope.setFocus(startsWith);
```



```
202. public void setFocus(String value) //browsescope.java
     ...
     204. focus = value.toLowerCase();
```



```
114. protected void doDSGet(Context context, HttpServletRequest
request, //browseservlet.java
     ...
     255. scope.setFocus(startsWith);
     ...
     357. browseInfo = Browse.getSubjects(scope);
```



```
141. public static BrowseInfo getSubjects(BrowseScope scope) throws
SQLException //browse.java
     ...
     147. return doBrowse(scope);
```



```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
     |
```

```
...
717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
...
952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
...
293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
...
952. statement.setString(1, (String) scope.getFocus());
...
960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
...
1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 8:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbyauthorservlet.java
...
83. String author = request.getParameter("author");
...
122. scope.setFocus(author);
```

```
202. public void setFocus(String value) //browsescope.java
...
204. focus = value.toLowerCase();
```

```
72. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbyauthorservlet.java
...
122. scope.setFocus(author);
...
124. BrowseInfo browseInfo = Browse.getItemsByAuthor(scope, orderByTitle);
```

```
240. public static BrowseInfo getItemsByAuthor(BrowseScope scope, //browse.java
...
260. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
      ...
      717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
      ...
      952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
      ...
      293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
      ...
      952. statement.setString(1, (String) scope.getFocus());
      ...
      960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
      ...
      1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Path 9:

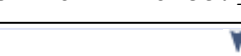
Query Name - **Parameter_Tampering**

Severity - **Serious**

```
71. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbysubjectservlet.java
      ...
      82. String subject = request.getParameter("subject");
      ...
      121. scope.setFocus(subject);
```

```
202. public void setFocus(String value) //browsescope.java
      ...
      204. focus = value.toLowerCase();
```

```
71. protected void doDSGet(Context context, HttpServletRequest
request, //itemsbysubjectservlet.java
      ...
      121. scope.setFocus(subject);
      ...
      123. BrowseInfo browseInfo = Browse.getItemsBySubject(scope, orderByTitle);
```



```
284. public static BrowseInfo getItemsBySubject(BrowseScope scope, //browse.java
```

```
...
304. return doBrowse(scope);
```

```
696. private static BrowseInfo doBrowse(BrowseScope scope) throws
SQLException //browse.java
...
717. int total = countTotalInIndex(scope, results.size());
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
...
952. statement.setString(1, (String) scope.getFocus());
```

```
291. public Object getFocus() //browsescope.java
...
293. return focus;
```

```
899. protected static int countTotalInIndex(BrowseScope scope, //browse.java
...
952. statement.setString(1, (String) scope.getFocus());
...
960. return getIntValue(statement);
```

```
1501. private static int getIntValue(PreparedStatement statement) //browse.java
...
1508. results = statement.executeQuery();
```

[Table of Contents] [Description]

Copyright © 2008 Checkmarx
Description materials copyrighted by the National Cyber Security Division of the U.S. Department of Homeland Security and the MITRE Corporation.