

**NIST/ITL CSD Biometric Conformance Test Software on Apache™
Hadoop®**

September 2014

Dylan Yaga
NIST/ITL CSD Lead Software Designer

Fernando Podio
NIST/ITL CSD Project Manager

National Institute of Standards and Technology (NIST)

Information Technology Laboratory (ITL)

Computer Security Division (CSD)

Contents

Contents.....	2
1. Disclaimer.....	3
2. NIST/ITL Computer Security Division Overview	4
3. BioCTS Overview.....	4
4. Overview	4
4.1. Requirements	6
4.2. Quick Start	6
5. Conformance Test Suites	7
6. Guide.....	7
6.1. Download and Installation.....	7
6.2. Running the Conformance Test Architecture	8
6.3. Screen Shot of BioCTS on Apache™ Hadoop®	9
7. Appendix A – Additional Information	10
7.1. Problems Encountered During Development Overview	10
7.1.1. Platform Incompatibilities	10
7.1.2. Platform Incompatibilities – Solved.....	10
7.1.3. Small File Problem	11
7.1.4. Small File Problem – Solved	11

1. Disclaimer

NIST/ITL BioCTS on

Apache™ Hadoop®

October 2010

The software was developed by the National Institute of Standards and Technology (NIST), an agency of the Federal Government. Pursuant to Title 15 United States Code Section 105, works of NIST are not subject to copyright protection in the United States and are considered to be in the public domain. Thus, the software may be freely reproduced and used. Please explicitly acknowledge the National Institute of Standards and Technology as the source of the software.

This software is released by NIST as a service and is expressly provided "AS IS." NIST MAKES NO WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT AND DATA ACCURACY. NIST DOES NOT REPRESENT OR WARRANT THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT ANY DEFECTS WILL BE CORRECTED.

NIST does not warrant or make any representations regarding the use of the software or the results thereof, including but not limited to the correctness, accuracy, reliability or usefulness of the software. By using this software or by incorporating this software into another product, you agree to hold harmless the United States Government for any and all damages or liabilities that arise out of such use.

Certain trade names and company products are mentioned in the text or identified. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose. With the exception of material marked as copyrighted, information presented in this document is considered public information and may be distributed or copied. Use of appropriate byline/photo/image credits is requested.

2. NIST/ITL CSD Support for Biometrics Standards and Conformance Testing

NIST/ITL Computer Security Division supports the development of biometric conformance testing methodology standards and other conformity assessment efforts through active technical participation in the development of these standards and the development of associated conformance test architectures and test suites. These test tools are developed to promote adoption of these standards and to support users that require conformance to selected biometric standards, product developers and testing labs.

3. BioCTS Overview

BioCTS is a traditional desktop based application developed in Microsoft® C# used as either the installer Graphical User Interface, or Command Line to test conformance to Biometric Data Interchange Records. There are two Conformance Test Architectures (CTAs):

- Tests Implementations of ANSI/NIST-ITL 1-2011 and ANSI/NIST-ITL 1-2011 Update: 2013
- Tests Implementations of select ISO/IEC 19794-X Generation 1 & 2 Data Formats, ANSI/INCITS, and several PIV Profiles of Standards

The software tests 1000s+ of files in a single Batch Test, allows editing of files, and provides charts, and detailed test results in text and XML formats.

4. Overview

Apache™ Hadoop® is an Open Source Framework for creating scalable distributed computing system (also known as a cluster) which runs on commodity computers (not specialized hardware) known as Nodes.

Apache™ Hadoop® processes data using the MapReduce Programming model created by Google. The Map function “Maps” portions of data out to Nodes, and processes it, and the Reduce function “Reduces” the portions of processed data results into an aggregation.

BioCTS on Apache™ Hadoop® is a multiphase MapReduce job that will process a set of Implementations Under Test (IUT) for Conformance Testing to a specific Biometric Data Interchange Record format. The problems and solutions to the reasons behind this multiphase approach can be found in Appendix A.

1. A collection of IUTs must be on the Computer that will send them to the Apache™ Hadoop® cluster
2. The IUTs must be converted to base64 and uploaded to Apache™ Hadoop®
3. The SmallFilesToSequenceFile MapReduce Job must be run on the base64 files to generate a single Sequence File; using the Apache™ Hadoop® Filesystem Path of the file as the Key, and the base64 data as the Value
4. The Sequence File must be passed to the BioCTS MapReduce Job, which performs the Conformance Tests on the decoded base64 data

5. The BioCTS output files must be retrieved from Apache™ Hadoop® and split for analysis
6. These steps are displayed below, in figure 2-1.

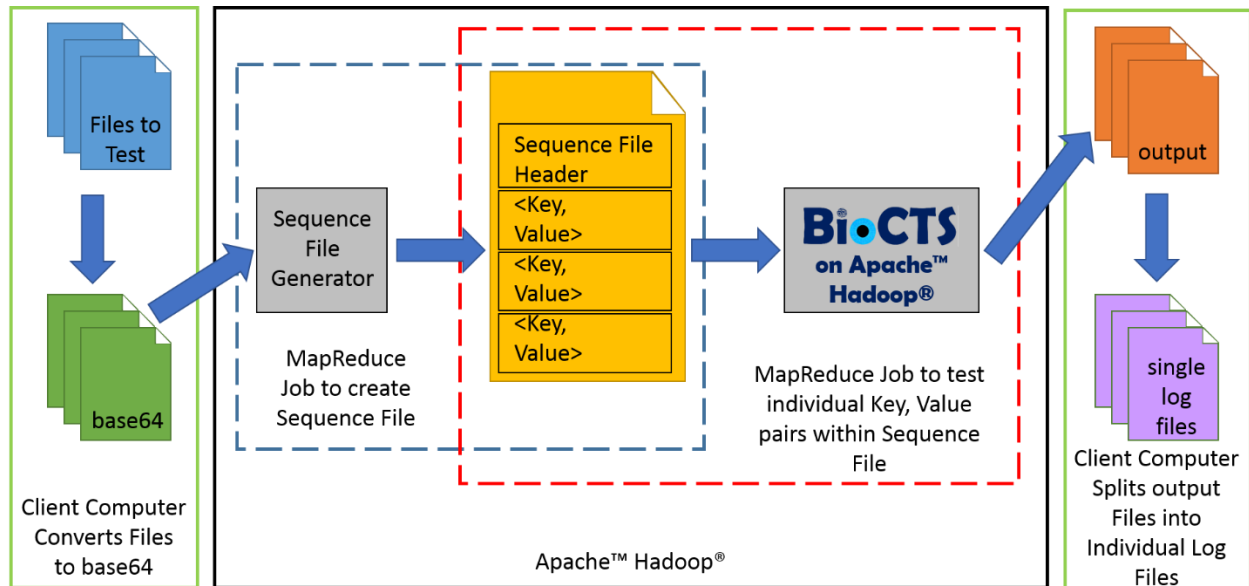


Figure 4-1 - All Phases of BioCTS on Apache™ Hadoop®

All of the above steps are automated by a script that is provided within the download zip file.

4.1. Requirements

The following versions of software were used during the development of BioCTS on Apache™ Hadoop®. Newer versions of the software may work – but have not yet been tested.

- Linux Operating System (e.g. CentOS version 6.2)
- Apache™ Hadoop® version 2.0.0
- Java version 1.6.0_31
- Mono version 3.0.7
- MonoDevelop version 3.1.1

4.2. Quick Start

To use BioCTS on Apache™ Hadoop® as quickly as possible, a pre-configured Virtual Machine (VM) system image may be used. The Quick Start VM image used for development of BioCTS on Apache™ Hadoop® was Cloudera, Inc.'s *QuickStart VM for CDH version 4.7.x* located:

http://www.cloudera.com/content/support/en/downloads/quickstart_vms/cdh-4-7-x.html

However, any Linux installation with the proper versions of the required software should work – adjustment of the paths within the script files may be required to meet each individuals' needs.

5. Conformance Test Suites

Currently, BioCTS on Apache™ Hadoop® supports a Conformance Test Suite for ANSI/NIST-ITL 1-2011. The BioCTS on Apache™ Hadoop® software can be adapted to any of the BioCTS Conformance Test Suite Module files (which are Dynamically Linked Library files, DLLs) by editing the BioCTS_Hadoop_Map source code project.

6. Guide

6.1. Download and Installation

Pre Setup

1. Have a Linux machine running Apache™ Hadoop® version 2.0.0, with the Hadoop setup into the directory `/usr/lib/`
 - a. For the purposes of these instructions, the username used throughout will be 'cloudera'
 - b. For the purposes of these instructions, the text editor "gedit" is used throughout, however, any text editor may be used
2. Boot up the Linux machine running Hadoop
3. Download BioCTS Hadoop Package
4. Extract the BioCTS Hadoop Package, the default folder name is "BioCTS_Hadoop_Complete" to the user folder, in this case `"/user/cloudera/"`, the full path should be `"/user/cloudera/BioCTS_Hadoop_Complete"`
5. Launch a Terminal (Command Line)
6. Run the command `'sudo gedit ~/.bashrc'`
 - a. Add to the end of it the following lines:

```
PKG_CONFIG_PATH=/usr/lib/pkgconfig
export PKG_CONFIG_PATH
HADOOP_HOME=/usr/lib/hadoop-0.20-mapreduce/
export HADOOP_HOME
```
 - b. Save the file
 - c. Close gedit
7. Close and Relaunch the Terminal to reload the `.bashrc` session
8. Run the command `'sudo gedit /usr/lib/hadoop-0.20-mapreduce/conf/mapred-site.xml'`
 - a. Locate the Property with the name `"mapred.child.java.opts"`
 - b. Change the value to `"-Xmx4096m"`
 - c. The Property should look like this:

```
<property>
    <name>mapred.child.java.opts</name>
    <value>-Xmx4096m</value>
</property>
```
 - d. Save the file

- e. Close gedit

Setup Apache™ Hadoop® Folders

1. Launch the Terminal
2. Run the command `'sudo su hdfs'`
 - a. As superuser hdfs run the command `'hadoop fs -mkdir /user/cloudera'`
 - b. After the file is created, change the owner to cloudera by running the command `'hadoop fs -chown cloudera /user/cloudera'`
 - c. Exit the superuser hdfs by running the command `'exit'`
3. Run the command `'hadoop fs -mkdir /user/cloudera/an/base64'`

Setup Mono

1. Launch a Terminal
 2. Change directories to 'BioCTS Hadoop Map' by running the command `'cd /home/cloudera/BioCTS_Hadoop_Complete/'`
 3. Enter superuser mode by running the command `'su'` and entering the default password of `'cloudera'`
 4. Change the mode of the file "SetupMonoAndDepends" to 777 by running `'chmod 777 SetupMonoAndDepends'`
 5. Execute the SetupMonoAndDepends script by running the command `'./SetupMonoAndDepends'`
 6. Wait for the script to finish downloading, extracting, and building Mono, MonoDevelop, and their dependencies – maybe get a coffee; it can take a while
- If the script executed successfully, the program "MonoDevelop" should be present in "Applications > Programming"

6.2. Running the Conformance Test Architecture

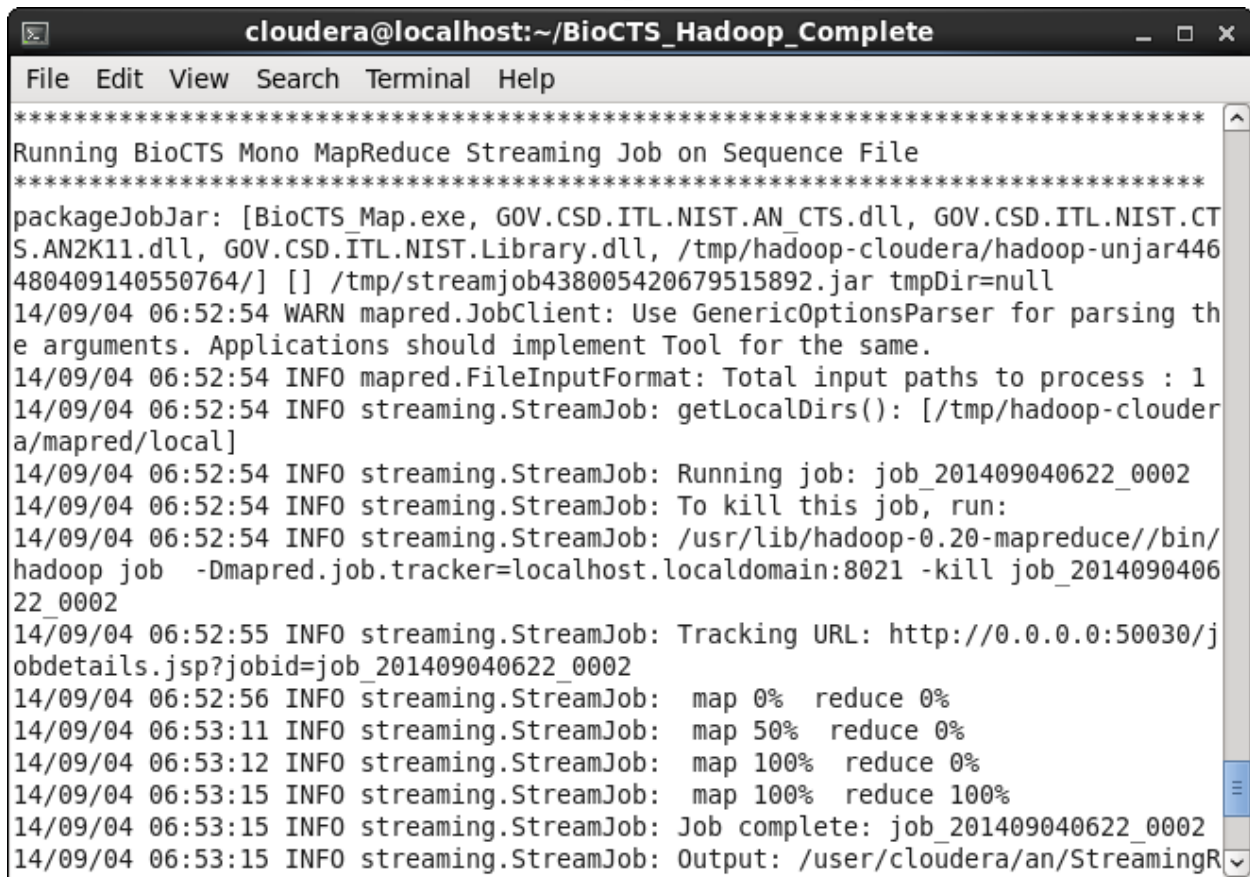
Within the **BioCTS_Hadoop_Complete** folder that was extracted, there is a script file **CompleteBioCTSONHadoop** that must get execution permission to run.

Run the command `"chmod 777 CompleteBioCTSONHadoop"`

Place some IUTs within the folder "Input_Files_To_Test", where they will be converted to base64 as part of the process of the script CompleteBioCTSONHadoop.

The script can be invoked by the command `"./CompleteBioCTSONHadoop"`

6.3. Screen Shot of BioCTS on Apache™ Hadoop®



```
cloudera@localhost:~/BioCTS_Hadoop_Complete
File Edit View Search Terminal Help
*****
Running BioCTS Mono MapReduce Streaming Job on Sequence File
*****
packageJobJar: [BioCTS_Map.exe, GOV.CSD.ITL.NIST.AN_CTS.dll, GOV.CSD.ITL.NIST.CT
S.AN2K11.dll, GOV.CSD.ITL.NIST.Library.dll, /tmp/hadoop-cloudera/hadoop-unjar446
480409140550764/] [] /tmp/streamjob438005420679515892.jar tmpDir=null
14/09/04 06:52:54 WARN mapred.JobClient: Use GenericOptionsParser for parsing th
e arguments. Applications should implement Tool for the same.
14/09/04 06:52:54 INFO mapred.FileInputFormat: Total input paths to process : 1
14/09/04 06:52:54 INFO streaming.StreamJob: getLocalDirs(): [/tmp/hadoop-clouder
a/mapred/local]
14/09/04 06:52:54 INFO streaming.StreamJob: Running job: job_201409040622_0002
14/09/04 06:52:54 INFO streaming.StreamJob: To kill this job, run:
14/09/04 06:52:54 INFO streaming.StreamJob: /usr/lib/hadoop-0.20-mapreduce//bin/
hadoop job -Dmapred.job.tracker=localhost.localdomain:8021 -kill job_2014090406
22_0002
14/09/04 06:52:55 INFO streaming.StreamJob: Tracking URL: http://0.0.0.0:50030/j
obdetails.jsp?jobid=job_201409040622_0002
14/09/04 06:52:56 INFO streaming.StreamJob: map 0% reduce 0%
14/09/04 06:53:11 INFO streaming.StreamJob: map 50% reduce 0%
14/09/04 06:53:12 INFO streaming.StreamJob: map 100% reduce 0%
14/09/04 06:53:15 INFO streaming.StreamJob: map 100% reduce 100%
14/09/04 06:53:15 INFO streaming.StreamJob: Job complete: job_201409040622_0002
14/09/04 06:53:15 INFO streaming.StreamJob: Output: /user/cloudera/an/StreamingR
```

Figure 6-1 - BioCTS on Apache™ Hadoop® in Linux Terminal

7. Appendix A – Additional Information

7.1. Problems Encountered During Development Overview

- Platform Incompatibilities
 - Linux vs. Microsoft® C#
 - Microsoft® within a Java MapReduce Job
- How Apache™ Hadoop® Processes Data vs. How BioCTS Processes Data
 - Large Files, many splits vs. Small Discrete Files, no splitting
 - Linefeed Characters within Files

7.1.1. Platform Incompatibilities

- How can the Linux and Java-based Apache™ Hadoop® Framework work with the Microsoft® C# BioCTS Conformance Test Suites?
- Possible Solutions:
 - Rewrite the Software in Java
 - No: Long process, could introduce bugs, may not provide 100% exact compatibility with existing Conformance Test Suite
 - Change Platforms
 - Not yet: Initial implementation targets existing systems, which were already running Linux & Apache™ Hadoop®
 - Possible Future Work: Expanding to other Platforms
 - Investigate a method for running Microsoft® C# Under Linux
 - Quickest method, if possible

7.1.2. Platform Incompatibilities – Solved

- Step 1: Find a way for Microsoft® C# to Run under Linux
 - The open source implementation of C#, known as Mono, is cross-platform, and runs well under Linux
 - The BioCTS software compiles perfectly with Mono and can then run in Linux
- Step 2: Find a way for the Java-Based Apache™ Hadoop® MapReduce to run C#
 - A method called “Hadoop Streaming” enables Apache™ Hadoop® to use any program/programming language that Linux supports to be embedded within a MapReduce Job
 - This is where the information was sparse, the API is very informative, but literature and publications simply mention the capability in passing
 - During research, was unable to locate a real world example of doing this exact procedure
- What is Hadoop Streaming?
 - Hadoop Streaming is a JAR (Java Archive) file that is distributed with Apache™ Hadoop®
 - This JAR file allows a developer to create MapReduce jobs with any script or executable file, allowing the specification of

- Program for the Map function
- Program for the Reduce Function
- Input File Type
- Additional Files needed for distribution
- Allows Apache™ Hadoop® to use any program that can run on Linux

7.1.3. Small File Problem

- Apache™ Hadoop® processes large amounts of data by splitting it up, and often times from a single file
- BioCTS processes data by testing a large number of discrete files, as a whole – with many inter-relationships needed to be tested within a file – therefore it CANNOT be split
- A simple solution would be to concatenate the files – but Apache™ Hadoop® could still split them arbitrarily and not on file boundaries
- So, how to process many files at once, but have Apache™ Hadoop® think it is one giant file?

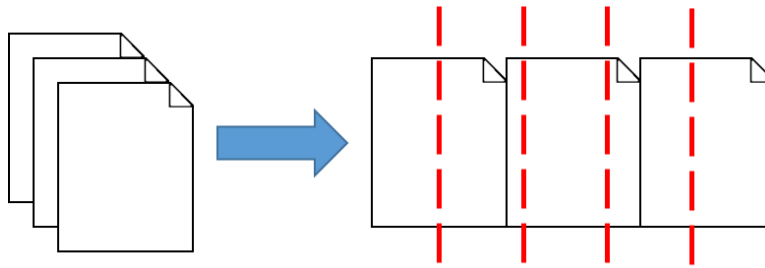


Figure 7-1 - Many Files Concatenated

7.1.4. Small File Problem – Solved

- Sequence Files – a file that is comprised of many key-value pairs
- The Key is the file path
- The value is the Data for that file
- Apache™ Hadoop® accepts Sequence Files as input, and knows to split the file based on key-values
- Each Key, Value pair is recorded on its own line
- So the data does not get arbitrarily split!

7.1.4.1. Sub Problem 2: Converting Files with Internal Line Feed Characters

- When converting the individual files into a Sequence File a problem was found:
 - When the files-to-convert contained internal new line characters, the process would not work
 - The Line Feed characters were placed in the Sequence File
- The solution? Eliminate the new line characters, but preserve the data – because the complete file is needed for testing
- The Implementation: A quick conversion of the files-to-convert to a base-64 encoded file, would preserve the data, but not contain new line characters

- The base-64 encoded data would then have to be decoded by the BioCTS software before processing

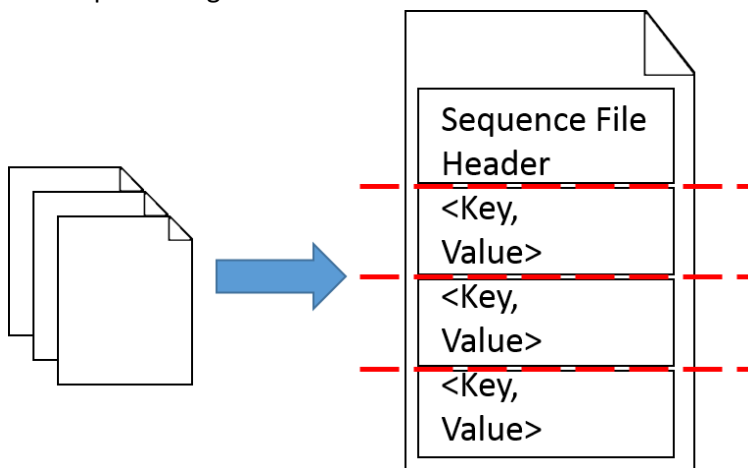


Figure 7-2 - Many Files Converted to a Sequence File