1. What changes, if any, need to be made to SSDF version 1.1 to accommodate secure development practices for generative AI and dual-use foundation models?
2. What AI-specific considerations should NIST capture in its companion resource?
3. What else should be captured in the SSDF Profiles?
4. Is there an alternative to an SSDF Profile that would be more effective at accomplishing the EO 14110 requirement, while also providing flexibility and technology neutrality for software producers?
5. What secure development resources specific to AI models do you find most valuable?
6. What is unique about developing code for generative AI and dual-use foundation models?
Achieve competitive advantage by enhancing customer experience, improving strategy & streamlining operations

AI could contribute up to $15.7 trillion to the global economy in 2030

Cybersecurity remains the ONLY risk that a majority of respondents say their organizations consider relevant

Through 2022, 30% of all AI cyberattacks will leverage training-data poisoning, AI model theft, or adversarial samples to attack AI-powered systems.

2 in 5 organizations have had an AI security or privacy breach. 1 in 4 were malicious attacks.
ML ADVERSARIAL ATTACKS ARE EXPLODING

REAL WORLD ATTACKS

1. Facebook
   Involved in 34 incidents, allegedly harming 60 entities.

2. Tesla
   Involved in 31 incidents, allegedly harming 41 entities.

3. Google
   Involved in 23 incidents, allegedly harming 31 entities.

ACCELERATING REGULATIONS


WEAPONIZED AML TOOLS

20+ free tools available online

Now is the time to protect
Models Need to be Scanned to Ensure Safety

Proprietary Model

Third-Party Models

Collect Training Data  
Training Process  
Trained Model

Models Need Real-Time Monitors & Detection for Attacks to Ensure Safety

Inference Process

API Boundary

Inputs

Outputs

MODEL TRAINING & DEVELOPMENT

PRODUCTION MODELS “AI ON THE EDGE”
SECURITY OPERATIONS FOR AI

DISCOVERY
- Where are all my AI models?
  - Model Registry
  - File Format Coverage

SAFETY & TRUST
- Are my AI models safe to use?
  - Malware
  - Vulnerabilities
  - Integrity Issues
  - Known Good State
  - Genealogy
  - Red Team Model Assessment

ATTACK MONITORING
- Are my AI models being attacked?
  - Adversarial ML
  - Poisoning
  - Model Evasion
  - Model Inversion
  - Model Theft
  - Prompt Injection
  - Confidential Data Leakage

RESPONSE
- What should I do about it?
  - AdvML Attack Remediations
  - Ticketing Systems
  - SecOps Tools

SITUATION AWARENESS
- Where is my AI security posture?
  - Detection Details Report
  - Security Health Dashboard
  - Risk Assessment Report
  - Response Dashboard
ML IS AN UNSECURED ATTACK VECTOR

Launchpad for lateral movement, deployment of malware, theft of IP/PII, and manipulation of the model output.
MITRE | ATLAS FRAMEWORK

Key Control Points

Run Time
Protection
MODEL SCANS
People & Process

RECONNAISSANCE
RESOURCE DEVELOPMENT
INITIAL ACCESS
ML MODEL ACCESS
EXECUTION
PERSISTENCE
DEFENSE EVASION
DISCOVERY
COLLECTION
ML ATTACK STAGING
EXFILTRATION
IMPACT

SEARCH FOR VICTIM'S
PROJECT AVAILABLE RESEARCH MATERIALS
SEARCH FOR PUBLICLY AVAILABLE VULNERABILITY ANALYSIS
SEARCH VICTIM-OWNED WEBSITES
SEARCH APPLICATION REPOSITORIES
ACTIVE SCANNING
ACQUIRE PUBLIC ML ARTIFACTS
OBTAIN CAPABILITIES
DEVELOP ADVERSARIAL ML ATTACK CAPABILITIES
ACQUIRE INFRASTRUCTURE
PUBLISH POISONED DATASETS
ACQUIRE ML ARTIFACTS
ML MODEL INCOMPROMISABLE
VALID ACCOUNTS
ML-ENABLED PRODUCT OR SERVICE
FULL MODEL ACCESS
USER EXECUTION
POISON TRAINING DATA
BACKDOOR ML MODEL
POISON ML MODEL
DISCOVER ML ARTIFACTS
ML MODEL FAMILY
DISCOVER ML ARTIFACTS
DISCOVER ML ONTOLOGY
DATA FROM INFORMATION REPOSITORIES
ML ARTIFACT COLLECTION
CREATE PROXY ML MODEL
EXFILTRATION VIA ML INFERENCE API
BACKDOOR ML MODEL
EXFILTRATION VIA CYBER MEANS
ML ATTACK STAGING
EXFILTRATION
DENIAL OF ML SERVICE
SPANNING ML SYSTEM WITH CHEAT DATA
ERODE ML MODEL INTEGRITY
COST HARVESTING
ML INTELLECTUAL PROPERTY THEFT
IMPACT
# AI Risk Requires Mitigation

<table>
<thead>
<tr>
<th>AI USE CASE</th>
<th>GENERATIVE AI</th>
<th>FRAUD MANAGEMENT</th>
<th>SUPPLY CHAIN ATTACKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOST COMMON ATTACK TYPES</td>
<td>Inference</td>
<td>Inference</td>
<td>Ransomware Attack</td>
</tr>
<tr>
<td></td>
<td>Data Poisoning</td>
<td>Bypass</td>
<td>Data Poisoning</td>
</tr>
<tr>
<td></td>
<td>Prompt Injection</td>
<td>Ransomware Attack</td>
<td>Backdoored Models</td>
</tr>
<tr>
<td>KEY CONTROLS NEEDED</td>
<td>Threat Modeling</td>
<td>Threat Modeling</td>
<td>Threat Modeling</td>
</tr>
<tr>
<td></td>
<td>Red Teaming</td>
<td>Red Teaming</td>
<td>Red Teaming</td>
</tr>
<tr>
<td></td>
<td>Model Scans</td>
<td>Model Scans</td>
<td>Model Scans</td>
</tr>
<tr>
<td></td>
<td>Real time/Run time protection</td>
<td>Real time/Run time protection</td>
<td>Real time / Run time protection</td>
</tr>
</tbody>
</table>
NOTES REGARDING NIST SP 800-218 AND AI/ML

- PO 1.1, 2.1, 2.3, 5.1 (ID + Document Sec. Req., et al.): Include ML Dev, ML Ops, DatSci; **Protect Training Corpora**

- PO 3.2: (Rec. Sec. Pract.) Scan 3rd party models used by org, ML Ops Protection (EDR), **Provenance of Models** (used and created), Data Curation & Training, Build/Supply Chain Security

- PS 1.1: (Src Storage) Extend to include Models and the **Data used for training**

- PS 2.1: (Publish SW Integrity Info) **Sign models**, Train/Build/Ops pipeline libs/vers (BOM), Secure Scoring APIs, Provenance, Need Standards around bias checks?

- PS 3.1: (Archiving) **Archive all models**, training data used per, meta data?

- PS 3.2: (Provenance) **Include 3rd party models** included in system, provenance of models derived from 3rd party models

- PW 1.1: (**Risk Modelling**) Back Doors/activation, Real Time Manifold Exploration/activation, Unusual Categorizations, Malware Infections, Vulnerabilities in models themselves

- PW 2.1: (SW Design Sec Req. Review) Each item has an analogous AI aspect

- PW 4.1, 4.4, 6.1: (Use Existing Secured Tools, et al.) Incl. **ML build/supply chain** (libs, frameworks); provenance of model and data

- PW 6.2: (predetermine comp/tools) Ban inherently insecure formats (Pickle, Cloud Pickle, etc.)

- PW 7.2: (Src Review/analysis) Incl. Model scanning backdoors, **exercise attacks against models**, bias checks

- PW 8.x: (Test Exe Code) Models can (mostly do) **have executable code embedded within** them

- PW 9.2: (Specify Settings) IAC for MLOps should be included

- RV 1.1: (Market info on vulns): Provenance of models, training/ops frameworks, file formats as part of BOMs

- RV 1.3: (Vuln. Discrl.): **include AI**: biased models, poisoned training, vuln. train/dev/ops frameworks, discovered bypasses

- RV 2.2: (Risk Responses) Industry doesn't understand the need to secure AI, nor what that undertaking involves - MITRE ATLAS framework to begin with; how can you even tell?

- RV 3.2: (RCAs) AI needs to be included in secure coding practices to be followed: **Must include the data** (corpora) used to train the model, APIs can assist in gradient-type attacks

- Runtime Controls missing

- Responsibility gaps/definitions amongst DatSci, MLOps, Product