Testing and Evaluation of AI Cyber Defense Systems

21 Apr 2022

Shing-hon Lau, PhD

Grant Deffenbaugh, PhD

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Document Markings

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM22-0358

Growing prominence of AI cyber defenses

- Shortfall of ~600k cybersecurity staff in US in 2021
  (https://www.cyberseek.org/heatmap.html)
  - AI can act as significant force multiplier for existing staff
- Concern over near-machine speed and low-and-slow attacks
  - NotPetya attack took down an entire Ukrainian bank in 45 seconds, making human responses near-impossible
  - Pattern changes from SolarWinds-style attacks can be slow and hard to notice
- Market share of AI in cybersecurity market was estimated at $8.8B in 2019, expected to grow to $38.2B by 2026
  (https://www.marketsandmarkets.com/Market-Reports/artificial-intelligence-security-market-220634996.html)

https://www.wired.com/story/notpetya-cyberattack-ukraine-russia-code-crashed-the-world/

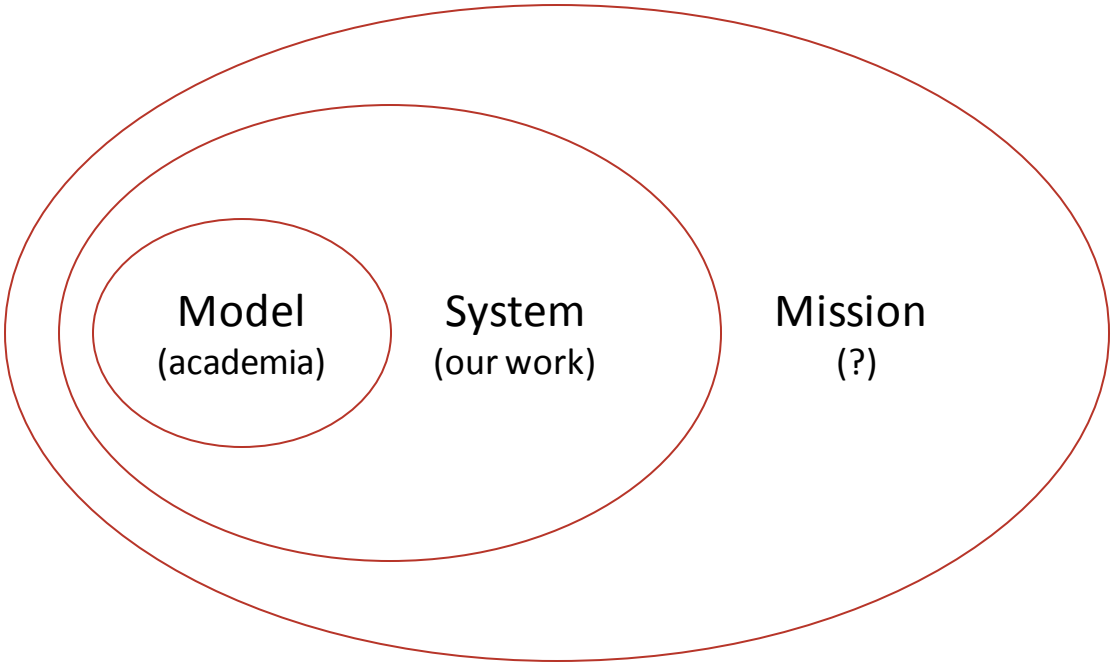
How do you know if an AI cyber defense system works?

- AI cyber defenses promises to offer a significant competitive advantage to any nation that can use it to protect their networks

- No USG department or agency can rely on an AI network defenses to protect their network without establishing that the defense works
- Establishing that a defense works requires a capability to test and evaluate

Testing and evaluation of AI systems

- Consider three levels of evaluation:
  - At the model level
  - At the system level
  - At the mission level
- Note that AI systems typically consist of multiple ML models in conjunction with hard-coded rules
- More comprehensive evaluations provide more realistic results but are considerably more difficult to execute



Challenges of testing an AI system

Environment Need to create a sufficiently realistic test environment where the AI can operate	Minimal access to models ML models are generally proprietary and preclude direct access
Learning AIs are generally designed to learn over time and may also be arbitrarily updated by humans	Adversarial AI Adversary may attempt to evade the AI through obfuscation or data poisoning

Scoping the problem

Problem:

- How can we test and evaluate the capabilities of AI **network** cyber defenses to detect malicious **network** activity, including in the presence of adversarial attempts to evade detection?

Approach:

- Develop a comprehensive testing methodology for AI network cyber defenses that involves testing an AI:
  - On an actual network
  - Without requiring direct access to the underlying ML models
  - While accommodating learning
  - In the presence of an evasive adversary

Completed vs. current work

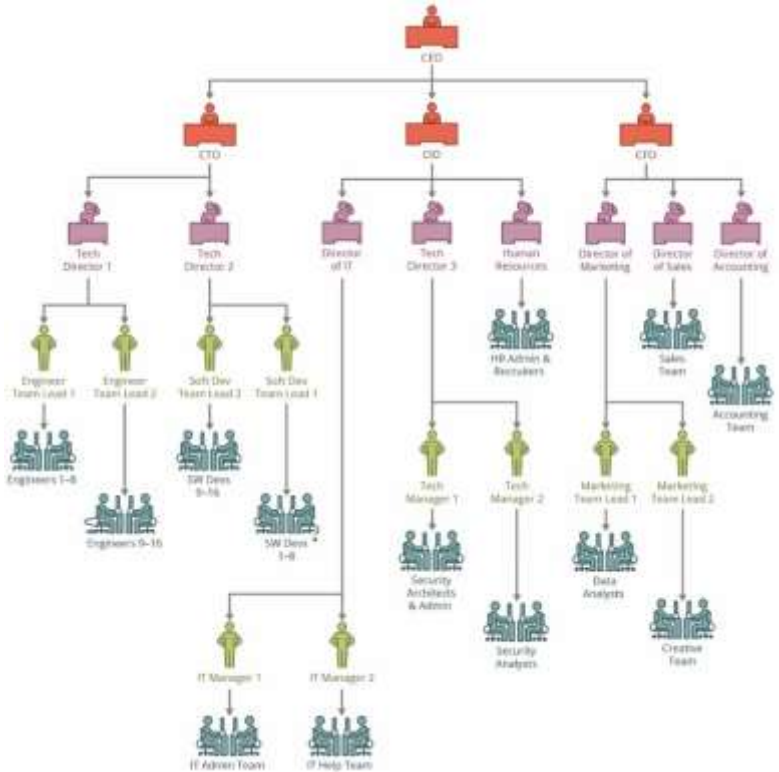
- Initial proof-of-concept work completed for DHS CISA
  - Question: Is it possible to evaluate the capabilities of an AI-based network defense on an actual network?
  - Goal: Demonstrate that evaluation is possible by testing two complete attack paths

- Current work is funded by Congressional LINE funding
  - Question: How do we effectively evaluate the capabilities of an AI-based network defense on an actual network, against a broad array of cyberattacks, and do so in a reasonable time frame?
  - Goal: Utilize a mix of existing tooling and new tooling to cover a large swath of the cyberattacks categorized in the MITRE ATT&CK framework

Challenges of testing an AI system

Environment **Need to create a sufficiently realistic test environment where the AI can operate**	Minimal access to models ML models are generally proprietary and preclude direct access
Learning AIs are generally designed to learn over time and may be arbitrarily changed by humans	Adversarial AI Adversary may attempt to evade the AI through obfuscation or data poisoning

Fictional company organization chart



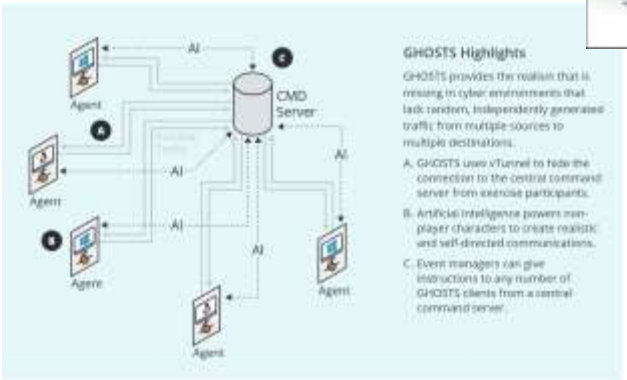
- ~ 100 employees
- 5 divisions
- 3 levels of management

- Each user is provided a unique behavior
  - Customized work schedules
  - Role-specific work tasks
  - Hobbies that influence personal use
- Privileges and access set by role

GHOSTS — realistic NPC orchestration

- GHOSTS is a SEI-developed Cyber Defense Training Tool to create realistic network traffic
- NPC's (Non-player characters) perform preassigned random tasks within VMs to generate actual network traffic
- The traffic is not simulated the users are
- Utilized in both completed and current work

- Each with separate profiles
- Realistic network traffic
- No PII
- Can simulate multiple network types



Challenges of testing an AI system

Environment

Need to create a sufficiently realistic test environment where the AI can operate

Minimal access to models

ML models are generally proprietary and preclude direct access

Learning

AIs are generally designed to learn over time and may be arbitrarily changed by humans

Adversarial AI

Adversary may attempt to evade the AI through obfuscation or data poisoning

Varying levels of sophistication

Simple	**Complex**
Counting	Markov-chain models
Rule parameter tuning	Deep neural nets
Naïve Bayes	Combinations of multiple ML models
Logistic regression	

- Devices generally do not reveal what model(s) are employed or even differentiate between hard-coded rules and ML-based approaches

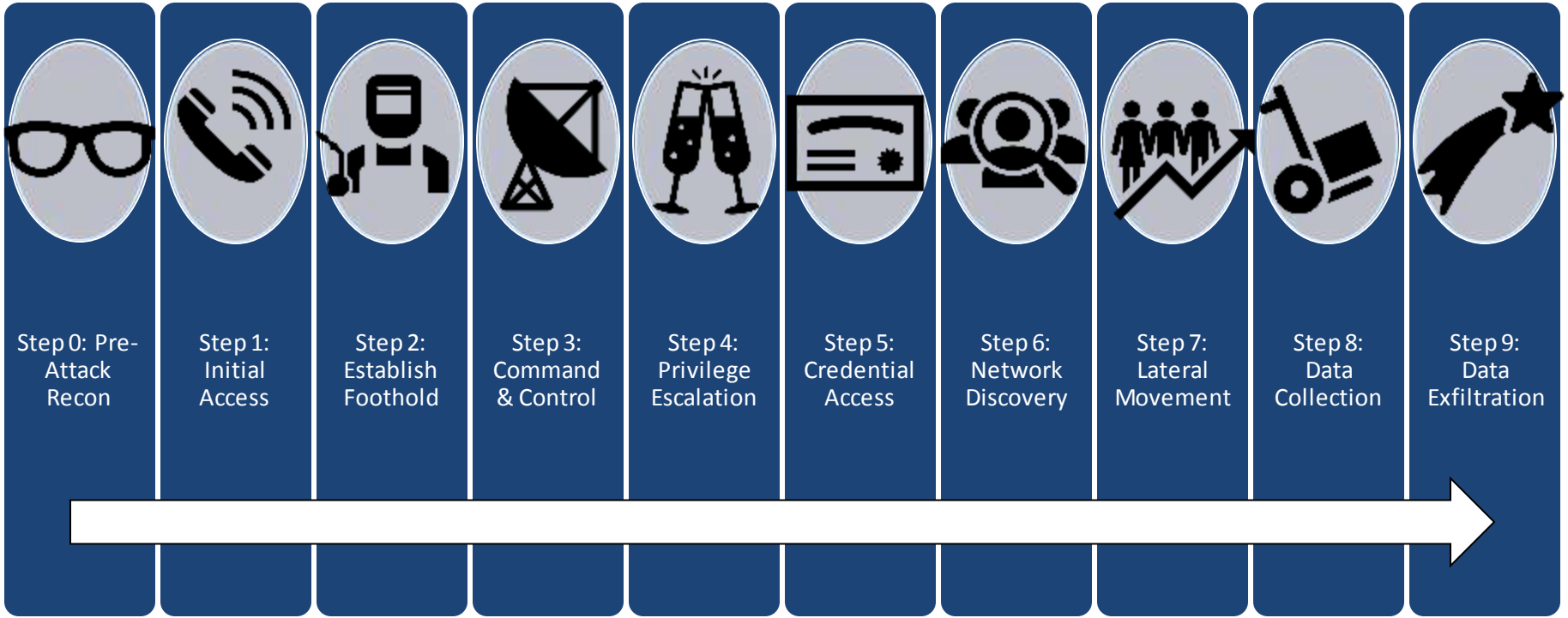
AI network cyber defenses

- Evaluate devices from leading vendors in the AI network cyber defense space
  - Previous DHS work evaluated two products
  - Current LINE work evaluates three products

- Devices are network-based devices that were installed on-prem and run air-gapped to avoid manual updates to the AI pushed by the vendor
  - Devices are still capable of learning from observed traffic
  - Want to eventually permit manual updates or cloud-based devices

- Devices are run in “passive” mode to detect any perceived threats but without automatic remediation

Baseline evaluation



Challenges of testing an AI system

Environment

Need to create a sufficiently realistic test environment where the AI can operate

Minimal access to models

ML models are generally proprietary and preclude direct access

Learning

AIs are generally designed to learn over time and may be arbitrarily changed by humans

Adversarial AI

Adversary may attempt to evade the AI through obfuscation or data poisoning

Training / testing AIs

- Initial training of AIs on normal background network traffic for vendor-specified duration
- Data is “pristine” – no malicious activities are present on the network
  - This represents a best-case scenario and polluted data can easily be substituted, if desired
- Previous evaluations were focused on “single point in time” evaluations
- Evaluations in current project are designed to cover a time period, providing insight into how defense capabilities change over time
- Developing tools will permit semi- or fully-automated evaluation capabilities, allowing us to repeatedly evaluate AI performance to investigate performance under changing background traffic conditions

Automating evaluations

- No access to underlying ML models, so we use repeated black-box evaluations
- We are utilizing MITRE CALDERA as the foundation for our tools to perform repeated evaluations
- Compose any applicable attack steps into many attack paths that can be tested in a semi- or fully-automated fashion
- Deliberate introduction of vulnerable hosts with known vulnerabilities maximizes the number of attack paths / steps that are applicable

Challenges of testing an AI system

Environment Need to create a sufficiently realistic test environment where the AI can operate	Minimal access to models ML models are generally proprietary and preclude direct access
Learning AIs are generally designed to learn over time and may be arbitrarily changed by humans	**Adversarial AI** **Adversary may attempt to evade the AI through obfuscation or data poisoning**

Obfuscating attack paths

- Obfuscations include:
  - Utilizing different tools or techniques that may not generate the same type(s) of traffic
  - Going sufficiently “low and slow”
  - Perform attack steps using machines of users that may perform similar activities as part of their job duties

- Define a set of parameters for obfuscations

- Tunable knobs can be introduced for “low and slow” rate, along with substitutable attack steps that generate different types of traffic, to determine how robust AIs are to each type of traffic

Data poisoning to enable attack paths

- Slowly introduce “poisoned data” by performing benign activities that generate traffic similar to the traffic that will be generated during attack
- Poisoned training data should cause AI to regard attack as normal background traffic
  - Note, the goal here is just to change the decision boundary of the AI without necessarily introducing a backdoor
- Data poisoning rates can be controlled through tunable knobs and durations + rates can be carefully controlled
- Quantify amount of poisoning required to change AI decision

Future directions

- Continued iteration on the realism of network traffic generated
- Expand environment to permit testing of joint endpoint and network products
- Expand methodology to include testing cloud / SaaS products
- Create capability for injecting attacks into replay traffic to permit more realistic testing of products in network environment of interest to an organization
- Consider other types (i.e., non-network) of AI cyber defenses

How you can help

- Our current LINE project is fully funded through FY22, but we could use a transition partner for the methodology and tools we are developing
- Connections with T&E directorate / lab to help us better understand current concerns
- We are open to potential collaborations in FY23:
  - Extending our methodology to both endpoint and cloud-based network cyber defense systems
  - Validating methodology on a larger network
  - Expanding methodology to include other cyber defenses

Dr. Shing-hon Lau
slau@sei.cmu.edu

Dr. Grant Deffenbaugh
grant@sei.cmu.edu

U.S. Mail
Software Engineering Institute
4500 Fifth Avenue
Pittsburgh, PA 15213-2612 USA

Website
https://www.sei.cmu.edu/contact-us/index.cfm