Engineering Realistic Synthetic Insider Threat (Cyber-Social) Test Data

Kurt Wallnau  
Senior Member of Technical Staff

Dr. Kurt Wallnau joined the SEI in 1993. He joined CERT Science of Cyber-Security (SoCS) in 2012 and is currently PI of the DARPA/ADAMS Red Team. Kurt has led and contributed to many SEI efforts to advance the theory and practice of software engineering. From 2010-12 he contributed to NSF's XSEDE program, defining its software and systems engineering process and from 2011-12 was manager of software development. Prior to that Kurt was PI of SEI's Predictable Assembly from Certifiable Components (PACC) initiative, which integrated software model checking, real-time analysis, and program generation to create software with predictable runtime behavior backed by verifiable evidence.
My Topic Today

We need to protect ourselves from the “Insider Threat.”

- Part of solution: systems that monitor and control social behavior.

What engineering discipline is effective for assuring cyber-social systems?

- Those systems that produce value by exploiting the laws of human nature (as distinct from cyber-physical systems that exploit the laws of nature)
- But this is too broad a scope!

A narrower focus: How do we test systems whose dynamics are based in human nature that is (at best) partially understood?

- We are excluding from consideration simple “tripwire” systems
- Our concern is with the technology emerging from the intersection of
  - Big data machine learning analytics
  - Many forms of monitoring data
Key Takeaways

Cyber-social systems pose big challenges to systems engineering:

- Cyber-social: System Test → Human Subject Experiment?
- Human subject experiments are hard: we need synthetic social behavior

What is “real” in social behavior is a great question for philosophers, but for engineers “realistic” is – and should be – a practical matter

- Realistic “enough” for the problem at hand
- Subject to the same “tradeoffs” as any other engineered artifact

Engineers make use of the sciences where possible but never wait for the sciences when it social needs dictate that solutions be built…
DARPA ADAMS – Insider Threat Detection

Examples will be drawn from experiences in DARPA/ADAMS¹

- Anomaly Detection at Multiple Scales
- Connect The Dots technology
- Insider Threat demonstration domain
- Using host-based sensor data provided by an industry partner
  - Users are de-identified with strong protections on the use of data

CERT provides Red Team data

Detecting Insiders: the “Haystack” Metaphor

Metaphor has tremendous power in the “cyber” world

- This “outsized” impact is acquired from nature of software itself
- Seasoned designers choose governing metaphors very carefully!

The Haystack metaphor is apt, descriptive, but not operational

- There is lots and lots of (human/social) data being collected
- Almost all of this data is innocuous (all but “the needle”)
- A tiny faction of this data is important (for some purpose)
- There are many haystack/silos, many needles to correlate
The “Control Systems” Metaphor

Control systems provide a better operational metaphor for testing:

- Process
- Sensors
- Concentrator
- Analytics
- Decision Loop
- Actuators

However, social phenomena are real in a different way than physical ones

- They are real because we say they are: social reality is *constructed*\(^1\)
- We’ve decided what is “real” by choosing what it is we “observe”

This is not circular – it is how humans create their social systems

- and why “realistic” must be defined with respect to context of use

ADAMS Insider Threat Detection (Gross Level)

**Test Method:**
Produce test data by “acting like” insiders on host monitored computers.

**Just starting in Phase 2**

Black Box to Red Team
Process for Producing Insider Threat Data

Defining Valid Cases

- Write scenario
- Script
- Features

Select users
- Target users
- Truth: user traces

Finding Interesting Users

Creating Observable Insiders

- Enact scenario
- Synthetic evidence

Blend users
- Synthetically augmented users

Making “Realistic” Insiders

- Red data
- Test data
Defining Valid Cases

Validity is a kind of realism

- We assert there exists “insider behavior”
  - The insider threat community “constructs” this reality
- Validity is obtained by sampling these behaviors
  - Scenarios are a “judgment sampling” technique
- How do we validate the sample of a constructed reality?
  - That’s a hard question for science
  - It’s not a well-formed question for engineering

jects are a “judgment sampling” technique
- How do we validate the sample of a constructed reality?
  - That’s a hard question for science
  - It’s not a well-formed question for engineering
Observable Behavior and Sensors

- Are tests that produce no observable behavior useful?
- We can choose to make insider behavior more or less observable.
- We can choose different ways to make a behavior observable.
- In traditional testing we would expect as criteria, for example:
  - Sensor coverage
  - Signal strength per sensor
  - Code coverage (on analytics)
Validity and Observability: Science and Engineering Tradeoffs

- We are often confronted by this conundrum:
  - Would any real insider behave in the ways we require them to behave just so we can make their actions observable?
- Can a valid scenario be biased to ensure that it is observable?
  - The objective of test isn’t to establish that an insider who knows the collection policy could escape detection
  - Endowing insiders with “realistic tradecraft” is itself an engineering concern in the way we design scenarios
Realism and Artifacts

- The most concrete interpretation of realism is: can the synthetic data be distinguished from the real data it simulates?
- An indicator of synthetic origin is called an “artifact”
  - Intended: the “moral” of the scenario
  - Unintended: anything else

- Our technique of “augmenting” real users with synthetic behavior lets us “piggy-back” on real behavior and minimize the ratio of real-to-synthetic behavior in our data
  - But there are many subtle sources of artifact, e.g. email style
Realism and Validity Beyond Actions

- Detection is more than spotting late night USB (you all know that!)
- Personality traits, cognitive styles, interpersonal patterns…are the “context” for interpreting user actions.
- We select users for “blending” that are interesting in a variety of ways:
  - They typically do the things done by scenario characters
    - Realism – avoid artifacts
  - The do not typically do these things
    - Validity – change of behavior as indicator

Finding Interesting Users

Defining Valid Cases

- write scenario
- select users
- truth: user traces
- target users
- features
Closing Thoughts

I have tried to persuade you that realism in social test data:

- Requires in operational context which establishes “how much” and “what kind” of realism is required
  - a decision procedure in an operational setting
  - engineering or engineering research purpose such as sensitivity testing
- Is a product of engineering tradeoff, usually made with an incomplete understanding of the social theories underlying the systems being tested.

It is not the case that “realism” is an intrinsic quality of the data

Programs such as ADAMS, and the technology we produced to construct test data, offers a way for the insider community to:

- Define scenarios narratives and characters with specific “traits”
- Specify how traits are mapped to (site-specific) data
- Generate test scenarios that can be relocated across different sites