Comparing reproduced cyber experimentation studies across different emulation testbeds

Sandia National Laboratories: Tom Tarman (tdtarma@sandia.gov), Laura Swiler, Eric Vugrin, Trevor Rollins, Jerry Cruz

Texas A&M University: Hao Huang, Abhijeet Sahu, Patrick Wlazlo, Ana Goulart, Kate Davis

CSET'21, August 9, 2021

Supported by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

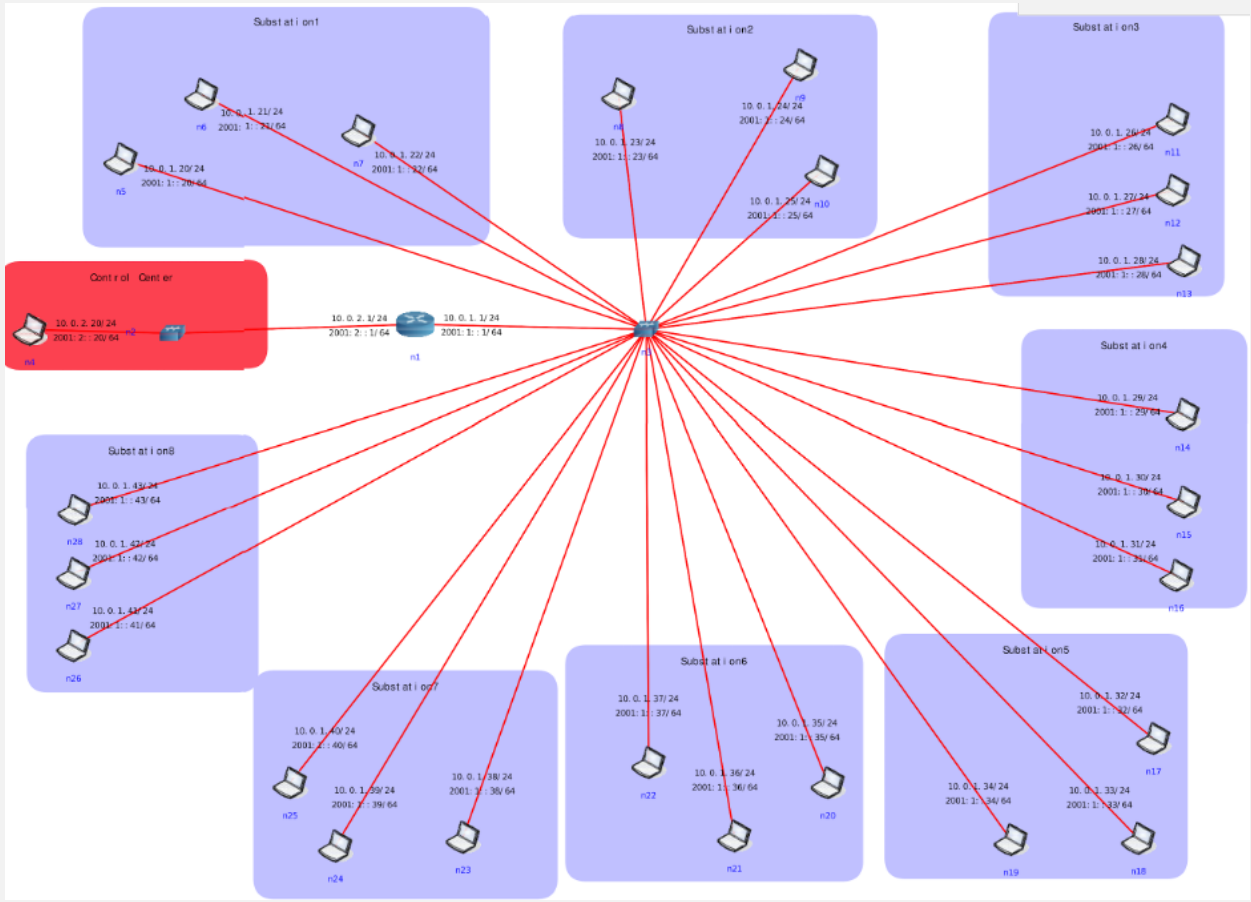
SAND2021-9485 C

Outline

- Scenario description and background
- Motivation for this study – is the original experiment reproducible, and how do we *quantitatively* compare both experiments?
- Data analysis and comparison metrics
- Lessons learned in reproducing this experiment
- Conclusions and future work

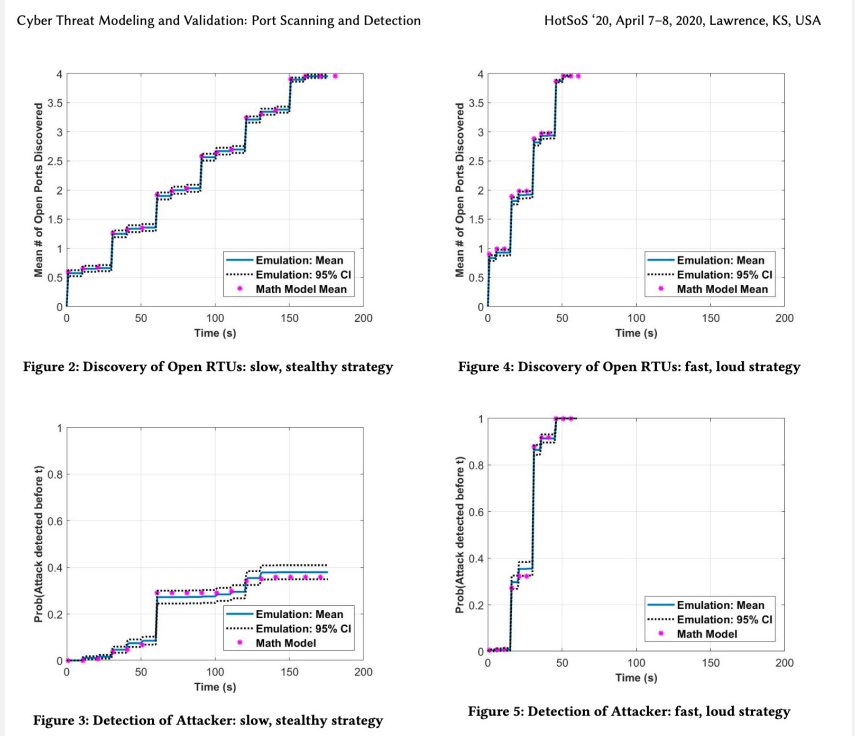
SCADA network scanning/detection study

- SCADA – Supervisory Control and Data Acquisition
- In this study, a SCADA network is used to control portions of a power grid
- The attacker has a presence in the control network, and uses Nmap to scan for vulnerable control devices
  - Modeled as open ssh port
- Defender uses Snort intrusion detection to detect scanning
- Attacker's objective is to identify as many vulnerable devices as possible without detection, using two strategies: Fast, and slow
- Sources of randomness
  - Attacker scan sequence
  - Network packet drop



Scanning/detection – mathematical model and its validation

- The original study[1] developed a mathematical stochastic model of the attacker's port discovery progress and the defender's ability to detect scanning
- This work used the minimega emulation testbed environment[2] to validate the mathematical model
- The original study showed good agreement between the models, relative to 95% confidence intervals



Comparison of mathematical and emulation results[1]

[1] Eric Vugrin, Jerry Cruz, Christian Reedy, Thomas Tarman, and Ali Pinar. 2020. Cyber threat modeling and validation: port scanning and detection. In Proceedings of the 7th Symposium on Hot Topics in the Science of Security. Association for Computing Machinery.
[2] https://minimega.org

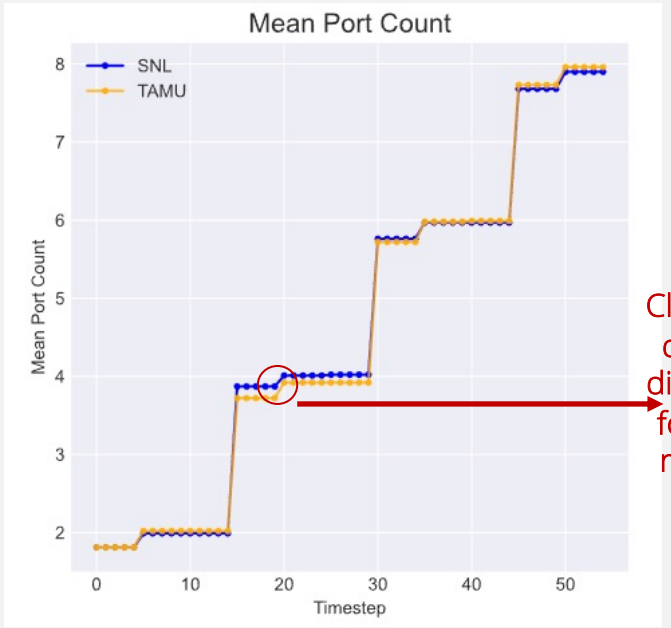
Can a different group reproduce original results on their testbed, and how do we compare both experiments?

- Reproducibility is key to ensuring results are correct, well documented, and unbiased
- Did paper describe the emulation experiment sufficiently well to be reproduced?
- Texas A&M used CORE testbed[1]
- Differences between minimega and CORE
  - Testbed technologies (minimega used kvm VMs on one physical machine, CORE used FreeBSD jails containers in one VM)
  - Experiment orchestration (minimega used SCenario ORCHestrator [SCORCH], CORE used custom scripts)
- Both testbeds used the same topology, mechanism for packet drop, Nmap and Snort versions, and port for "vulnerable" services
- Same experiment design: four scenarios {fast, slow} X {random, deterministic}
  - Random – random ordering of scanned host IPs, random packet drop
  - Deterministic – fixed scan order, no packet drop

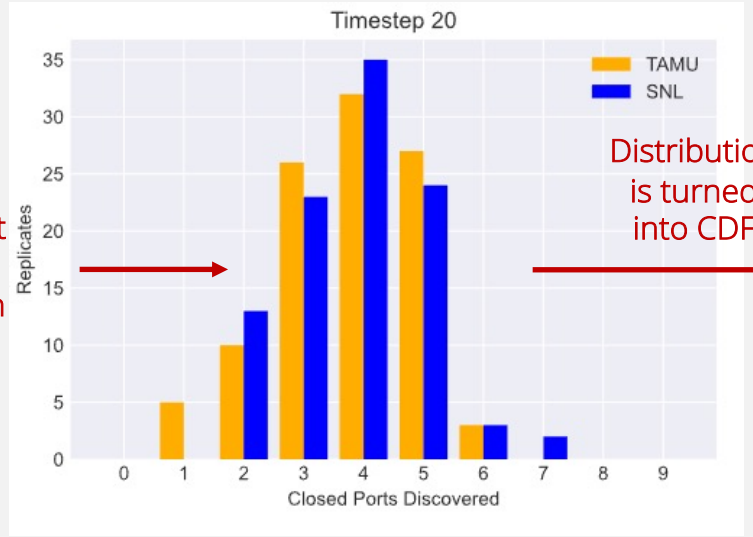
[1] Jeff Ahrenholz, Claudiu Danilov, Thomas R Henderson, and Jae H Kim. 2008. CORE: A real-time network emulator. In MILCOM 2008-2008 IEEE Military Communications Conference. IEEE, 1–7.

How do we compare ensembles of results?

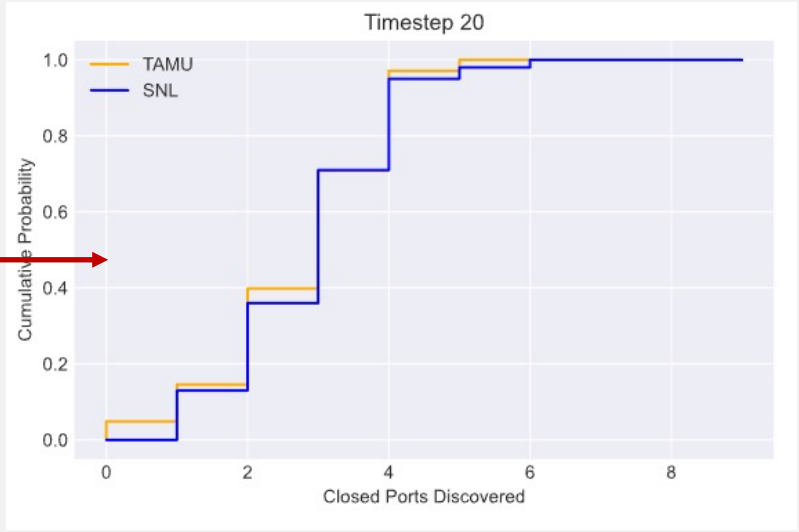
- There is inherent variability in cyber network results due to system timing, resources, operating systems, kernels, etc.
- We run a number of replicates (100 in this experiment) on both experimental platforms
- We want to **compare distributions** from the 100 TAMU CORE results vs. 100 SNL minimega results.



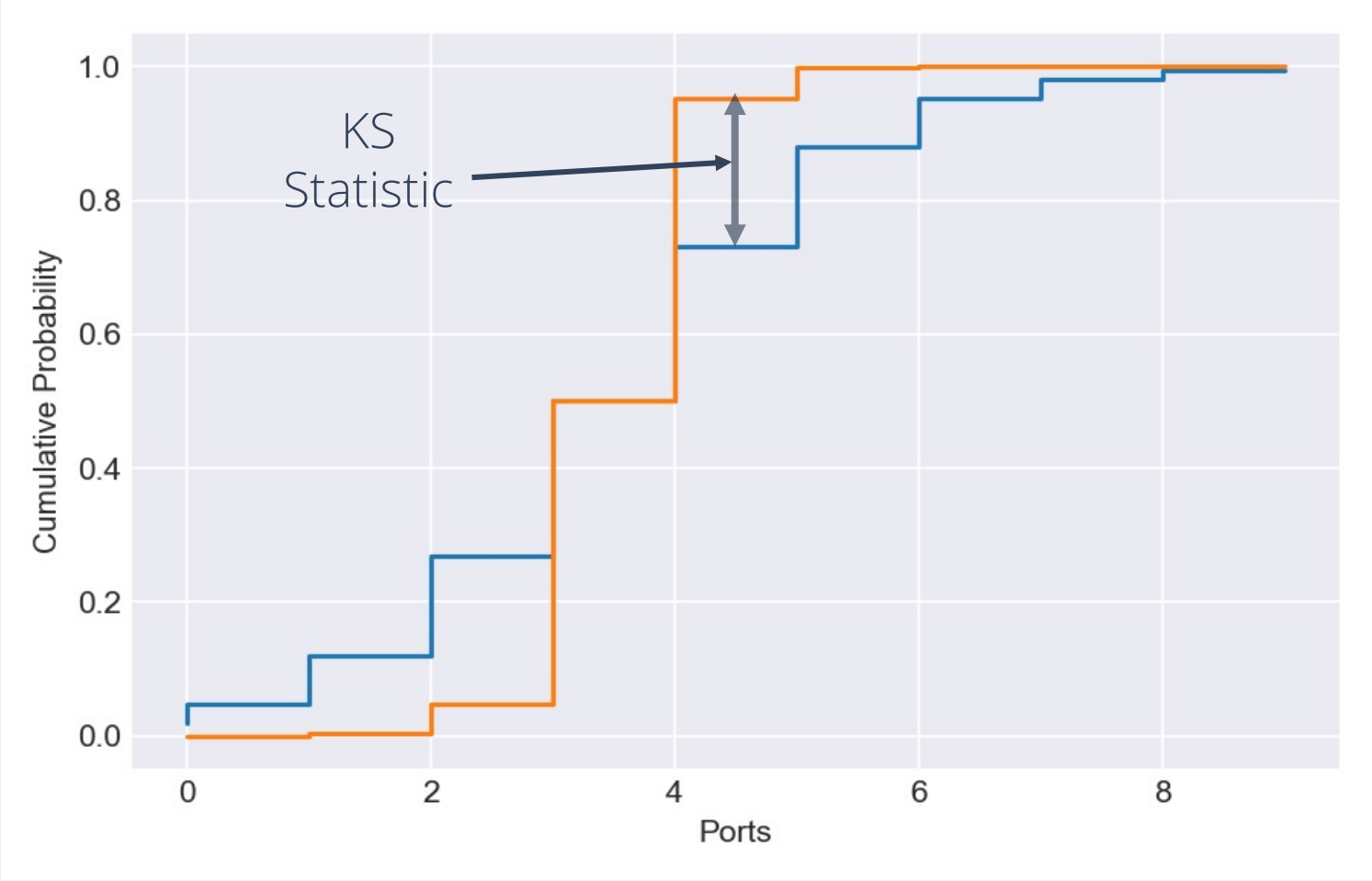
Closed port discovery distribution for all 100 replicates



Distribution is turned into CDF



How do we compare ensembles of results?



Distribution comparison can be performed by the Kolmogorov-Smirnov (KS) test

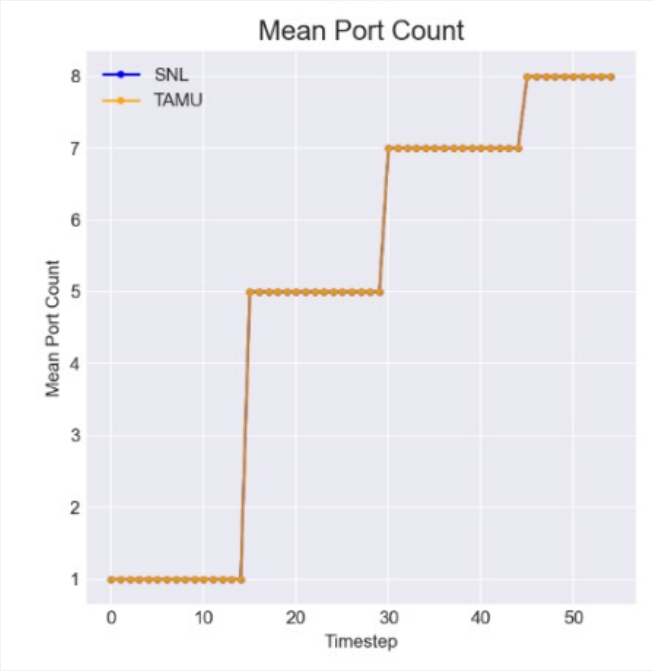
- Function of the max difference between CDFs
- p-value = 1 indicates complete agreement
- p-value less than 0.05 indicates one would not accept the hypothesis the distributions are the same.

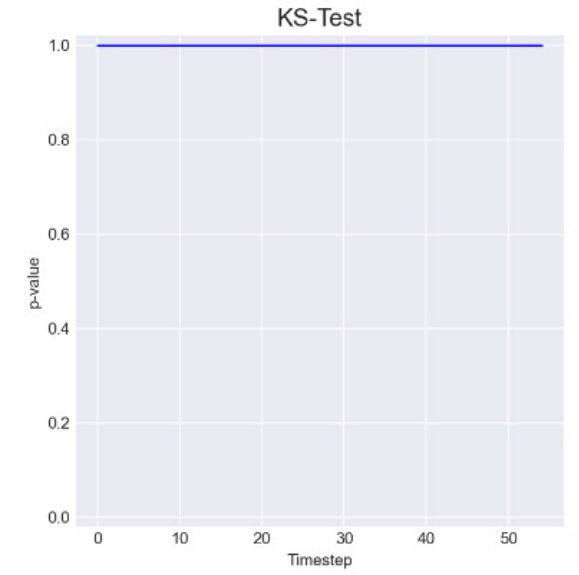
We looked at other comparison metrics: please see paper for more details.

[1] K.A. Maupin, L.P. Swiler, N.W. Porter, "Validation Metrics for Deterministic and Probabilistic Data," *Journal of Verification, Validation and Uncertainty Quantification*, Vol. 3, September 2018.

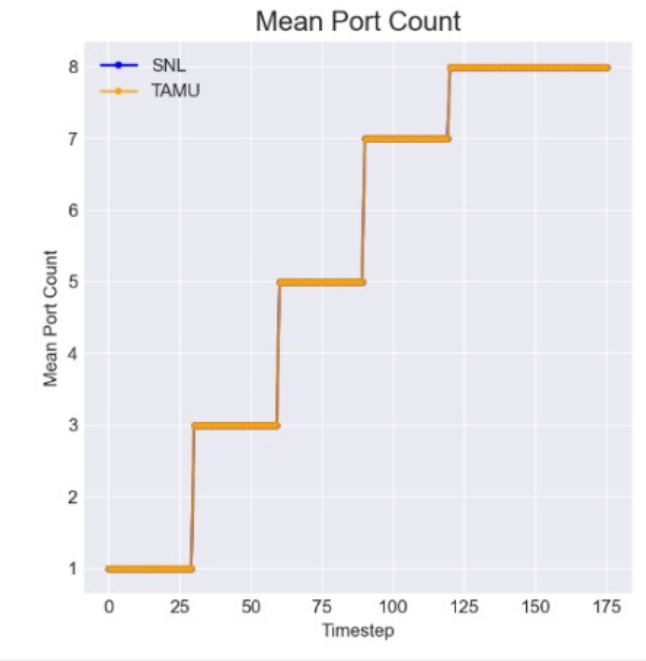
Results: No drops, fixed port order
No replicate variability: deterministic results match

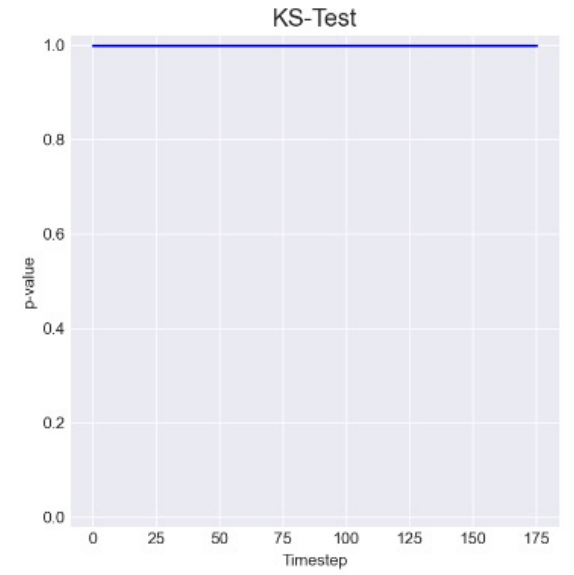
Fast





Slow

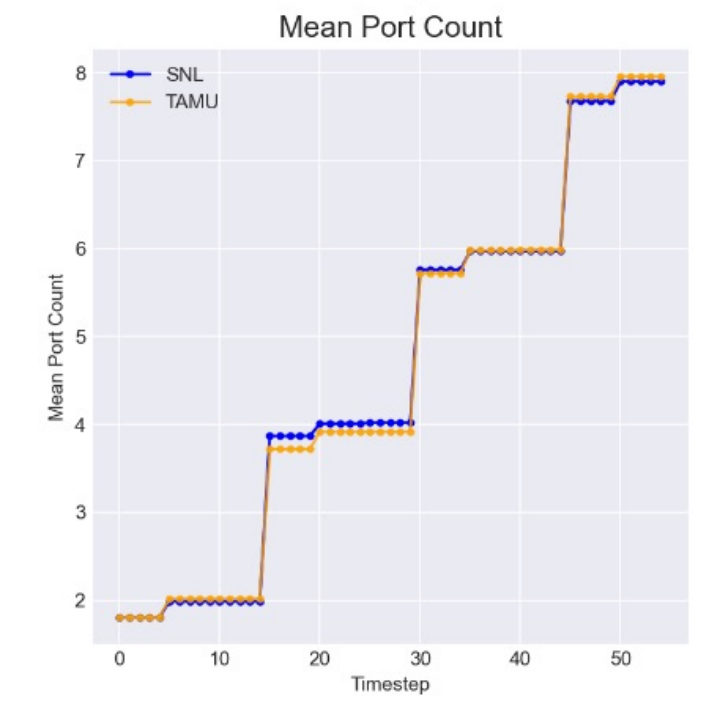


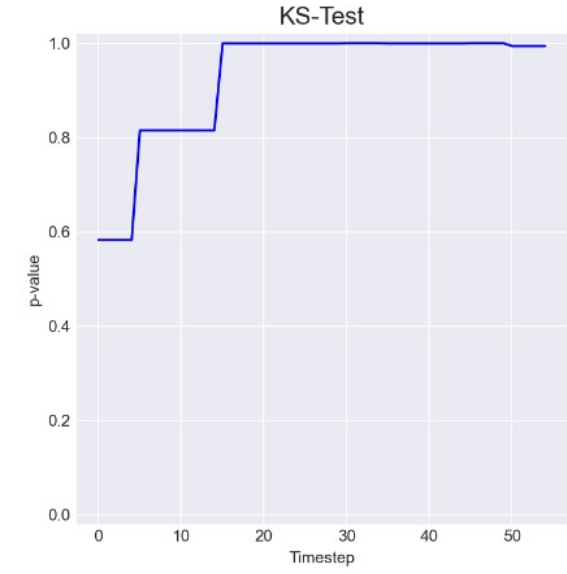


Results: with packet drops and random port ordering

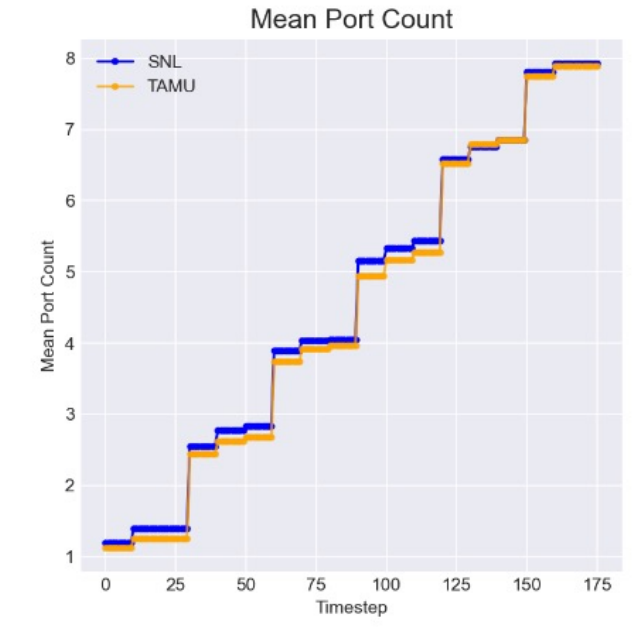
* Fast scenario more consistent than slow under stochastic conditions
* Statistical tests indicate two sets of data would not be considered different

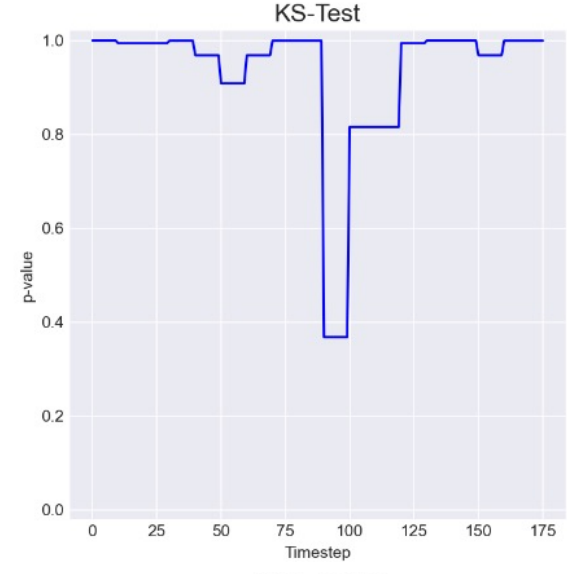
Fast





Slow





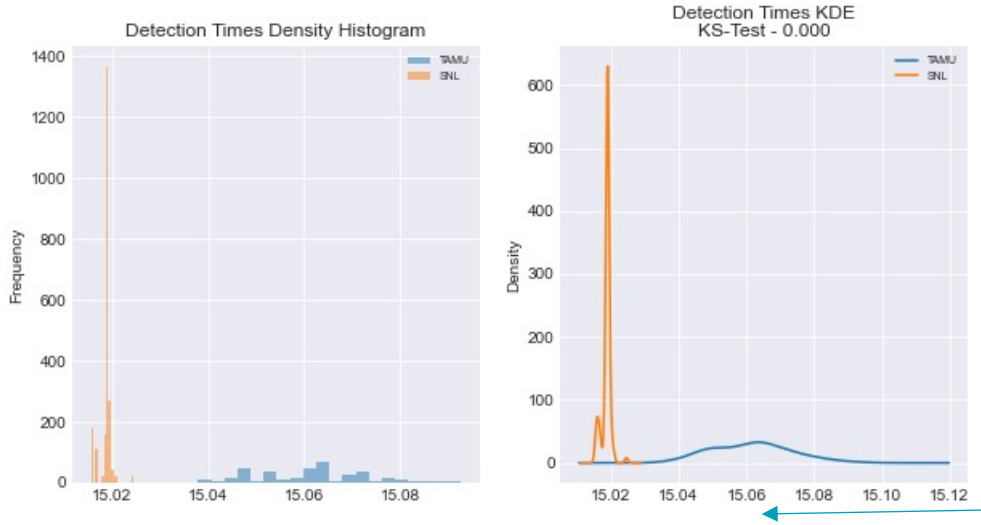
How well was the experiment reproduced?

We may care about the differences in magnitude and not care about distributional differences.

Alert detection times

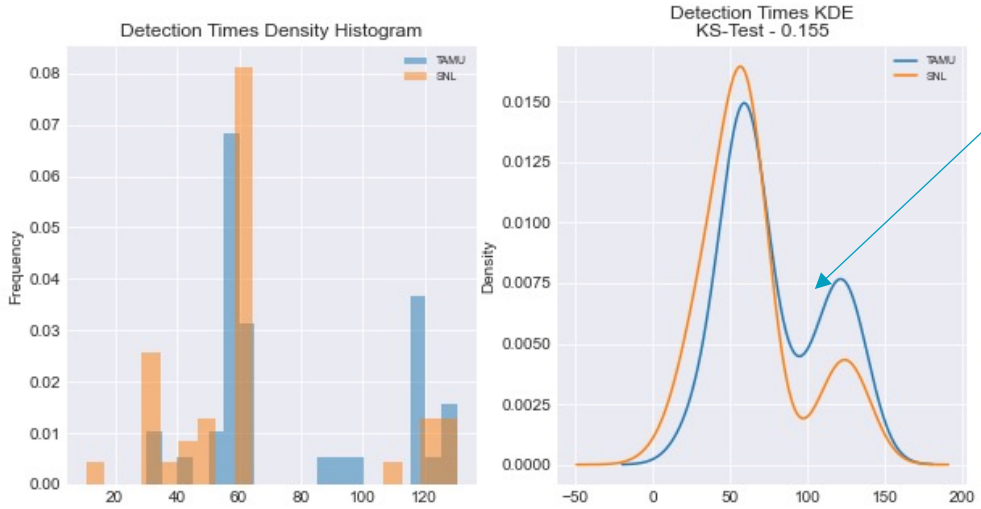
Fast – No Drop – Fixed Nmap order

- KS-test: 0.000
- Least variable experiment



Slow – Drop – Random Nmap order

- KS-test: 0.155
- Most variable experiment



The validation metrics depend on the question being asked:

Are these differences significant due to differing hypervisors, time synchronization, and experiment orchestration? Are they acceptable to be used in a larger attack model?

What did we learn about reproducing emulation experiments?

- Even after providing a comprehensive writeup and details of the experiment, both teams still required significant coordination to reproduce the experiment.
- It can be challenging to determine if small differences are due to differences in the hardware/emulation platform OR due to an implementation detail that is not correctly reproduced.
  - Subject matter expertise is critical
- Statistical tests and ensembles of replicate results can help in this comparison as they provide some estimate of the uncertainty inherent in the results on one platform.

Recommendations - what is needed to facilitate reproducibility?

- Public repositories for experimental artifacts
  - Topologies, applications, orchestration files
  - Github, SEARCCH[1], etc.
- Need consensus in artifacts and how testbed technologies use them
- Understand differences between common cyber experimentation platforms
  - Virtualization technologies (CPUs, network interfaces, switching, etc.)
- Appropriate metrics, depending on experiment question/objective
  - Distance measures between experimental results
  - Metrics to determine effects of platform differences on results

[1] Sharing Expertise and Artifacts for Reuse through Cybersecurity Community Hub (SEARCCH) project. 2021. https://searcch.cyberexperimentation.org/

Thank You

Thomas D. Tarman
tdtarma@sandia.gov