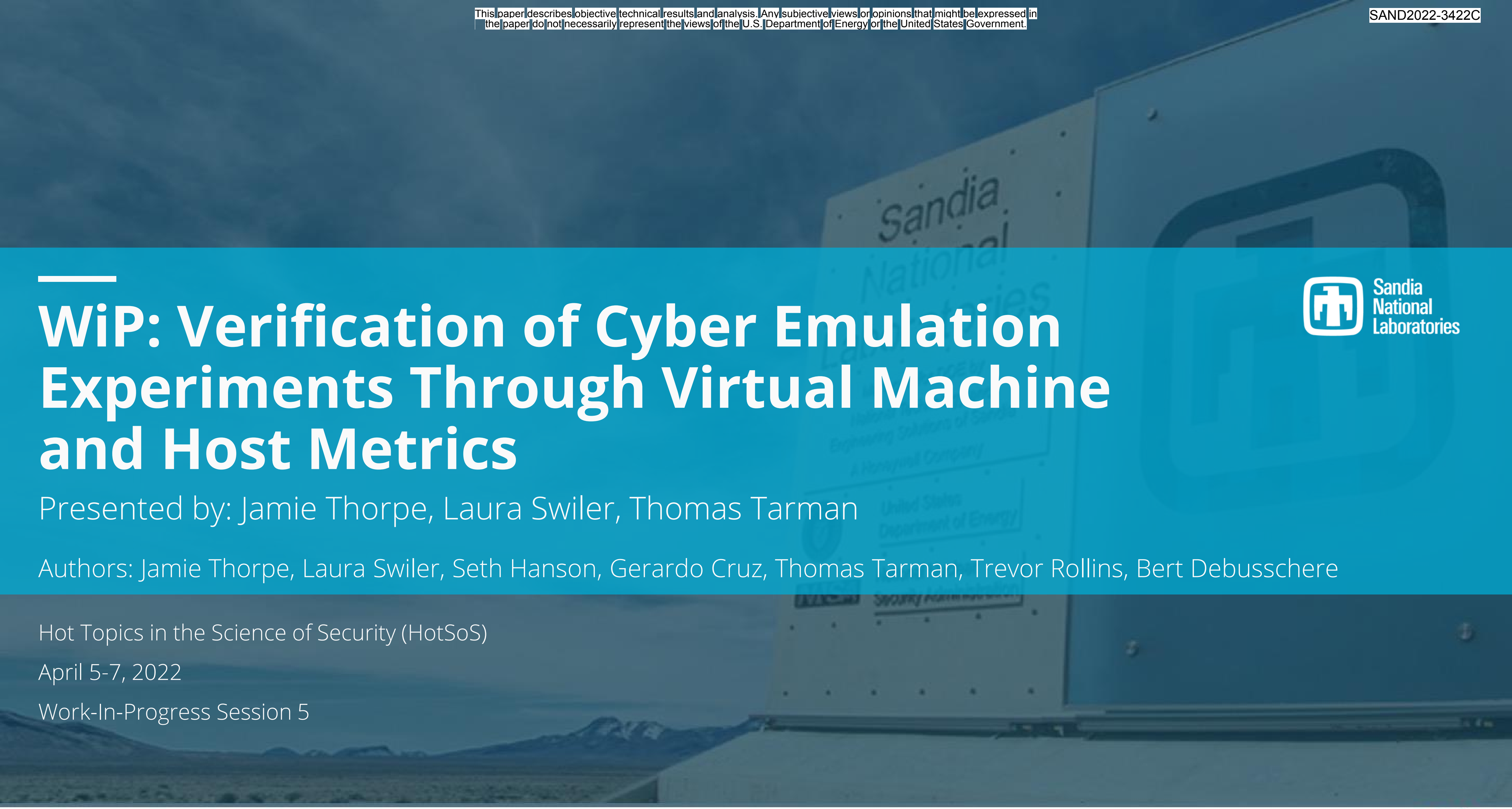
This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2022-3422C



WiP: Verification of Cyber Emulation Experiments Through Virtual Machine and Host Metrics

Presented by: Jamie Thorpe, Laura Swiler, Thomas Tarman

Authors: Jamie Thorpe, Laura Swiler, Seth Hanson, Gerardo Cruz, Thomas Tarman, Trevor Rollins, Bert Debusschere

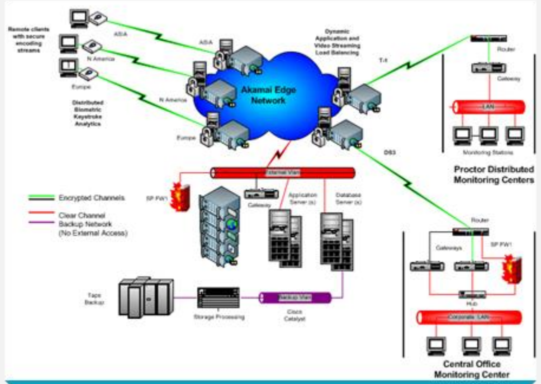
Hot Topics in the Science of Security (HotSoS)

April 5-7, 2022

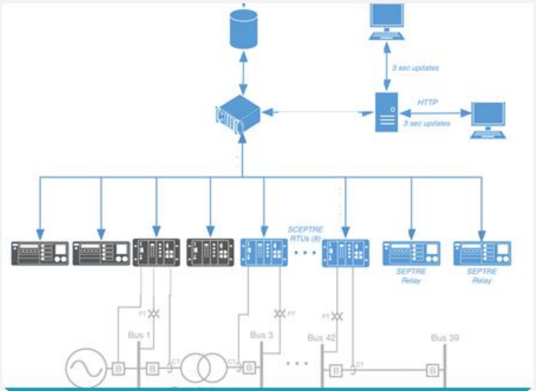
Work-In-Progress Session 5

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & 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.

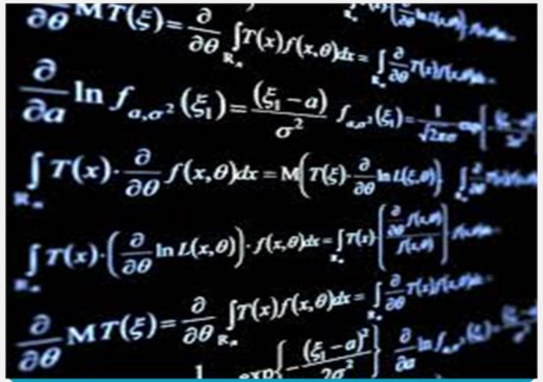
What is Cyber Experimentation?



ACTUAL SYSTEM



VIRTUALIZED TESTBED



SIMULATION

“BAD DAY” BRAINSTORMING

Increasing Realism
Decreasing Flexibility
Increasing Cost
Increasing Time

Increasing Abstraction
Increasing Flexibility
Decreasing Cost
Decreasing Time

Why Do We Need Cyber Experimentation?

To study complex cyber systems with rigor -

- "How resilient is my system to Threat X?"
- "How does Tool Y affect the cyber security of my system?"
- "How confident am I in these results?"

Challenge: Can we trust this approach for high consequence systems?

Rigorous Cyber Experimentation should be a Pillar of the Science of Cyber Security

Verification

Is the experimental environment working as intended?

- If so, results can be used to better understand the system modeled
- If not, experiment results may not be reliable

Different Types of Verification

- Timing Realism – Processes and network traffic occur at expected rate
- Traffic Realism – Network traffic contains expected fields/data
- Resource Realism – Physical host has enough resources to support experiment

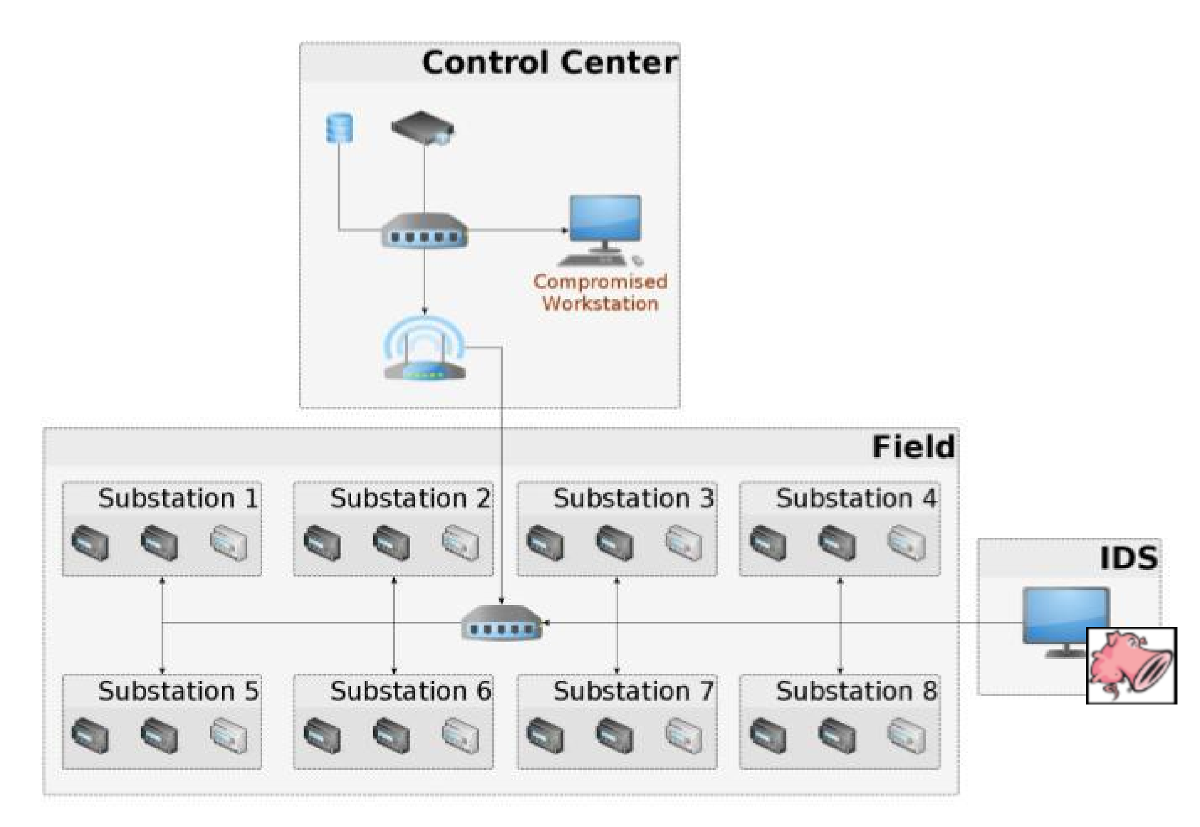
Approach

1. Devise mechanism for increasingly stressing physical host resources
   - Run more experiments (replicates) in parallel
2. Run multiple replicates in each resource setting
3. Collect key telemetry and results data from each replicate
   - Physical host load (telemetry)
   - In-experiment virtual machine functionality (telemetry)
   - In-experiment results
4. Compare telemetry from replicates under different resource settings with experiment results

Can a Telemetry-Based Metric be Used to Determine if the Results of a Replicate are Unreliable?

Scenario 1 – Scanning and Detection

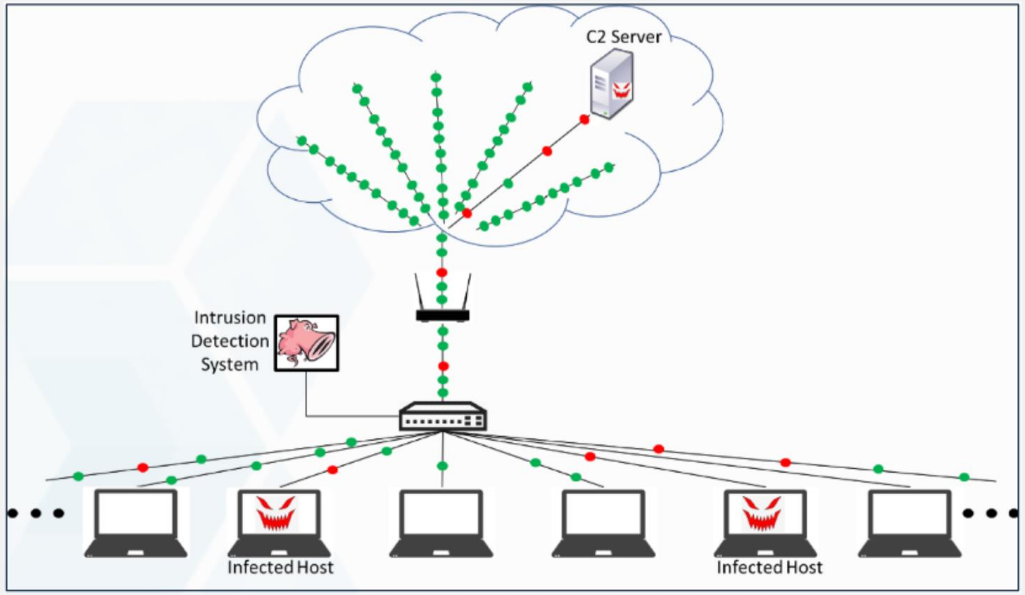
Detect adversary running port scan on 24 nodes



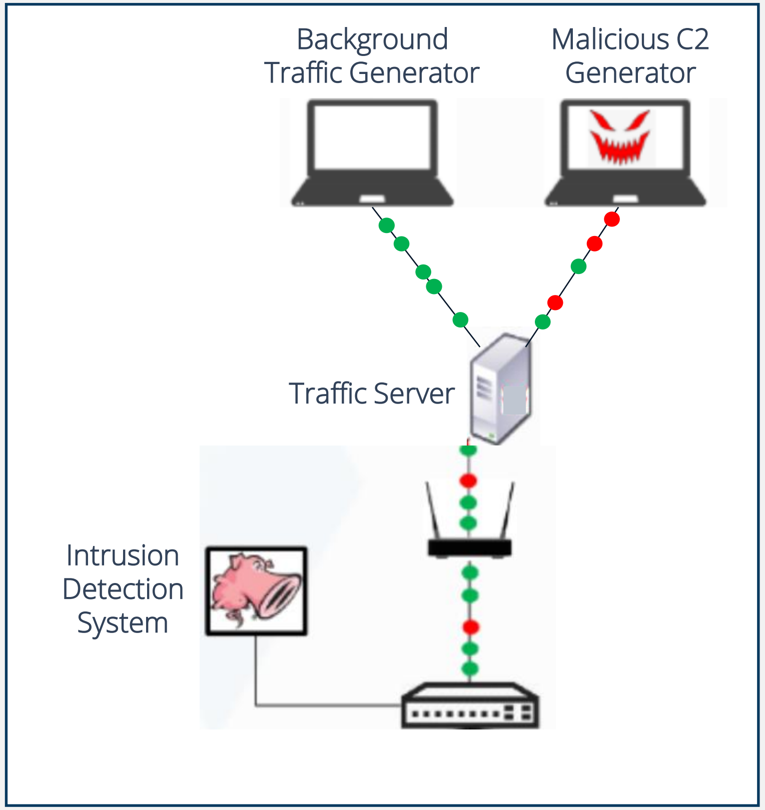
- Quantity of Interest: Detection Time
- Deterministic Scan Order
- No Packet Loss Assumed

Scenario 2 – Command and Control (C2)

Detect malicious traffic between host(s) and C2 server



Scenario as Described



Scenario as Modeled

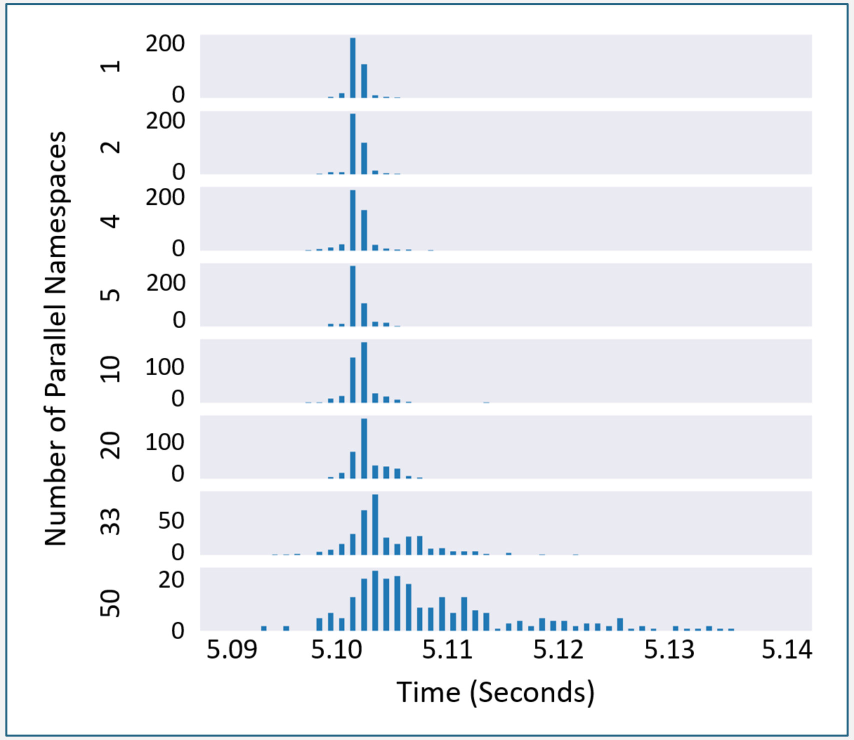
- Quantity of Interest: Number of Alerts at Certain Timestamps
- No Packet Loss Assumed

Results – Scenario 1 (Scanning and Detection)

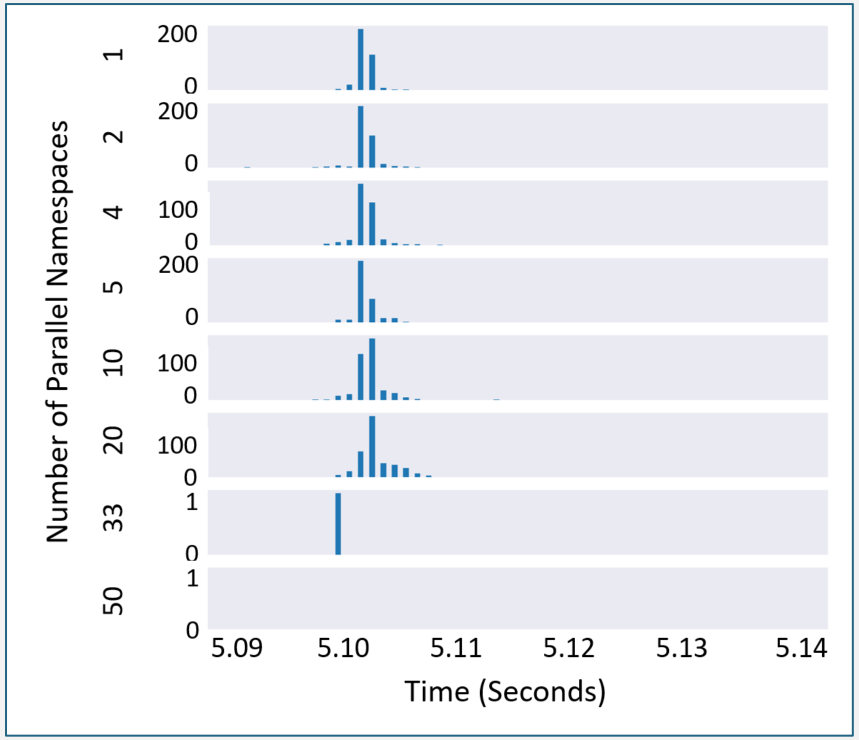
Example Metrics:

- Stolen Cycles = 0
- Load ≤ 64 Processes
- Throughput ≥ 250k bps

All replicates

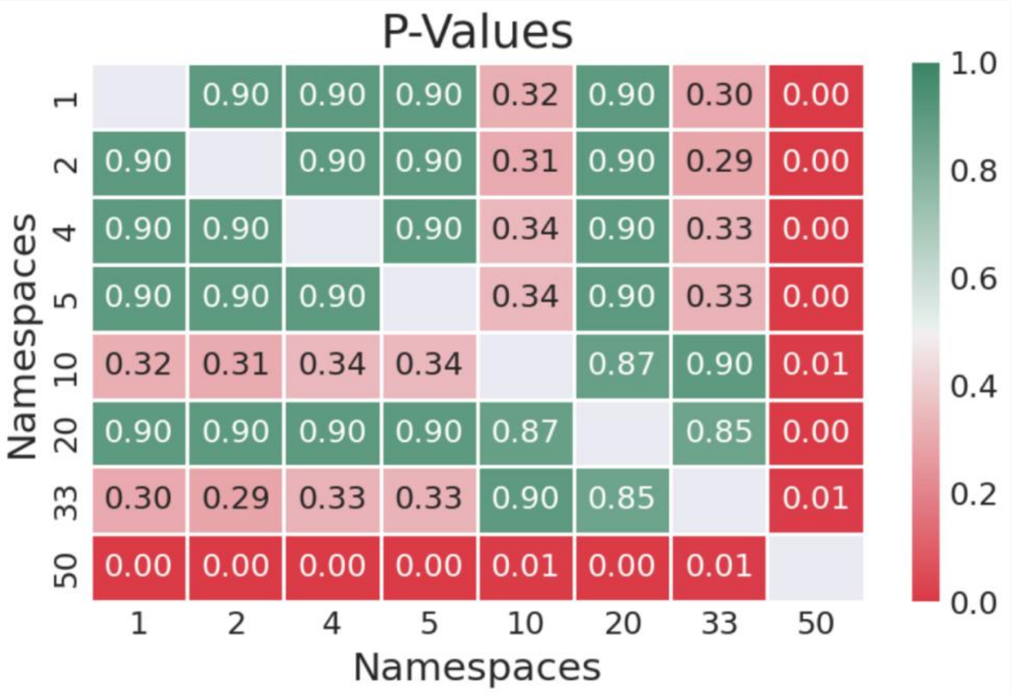


No stolen cycles



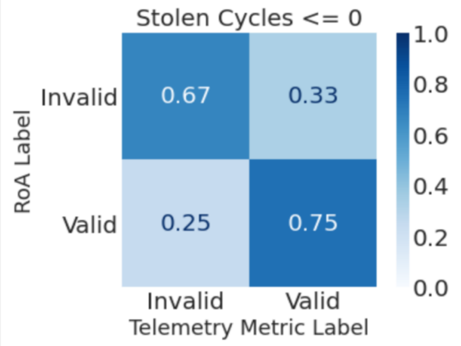
Results – Scenario 1 (Scanning and Detection)

All replicates

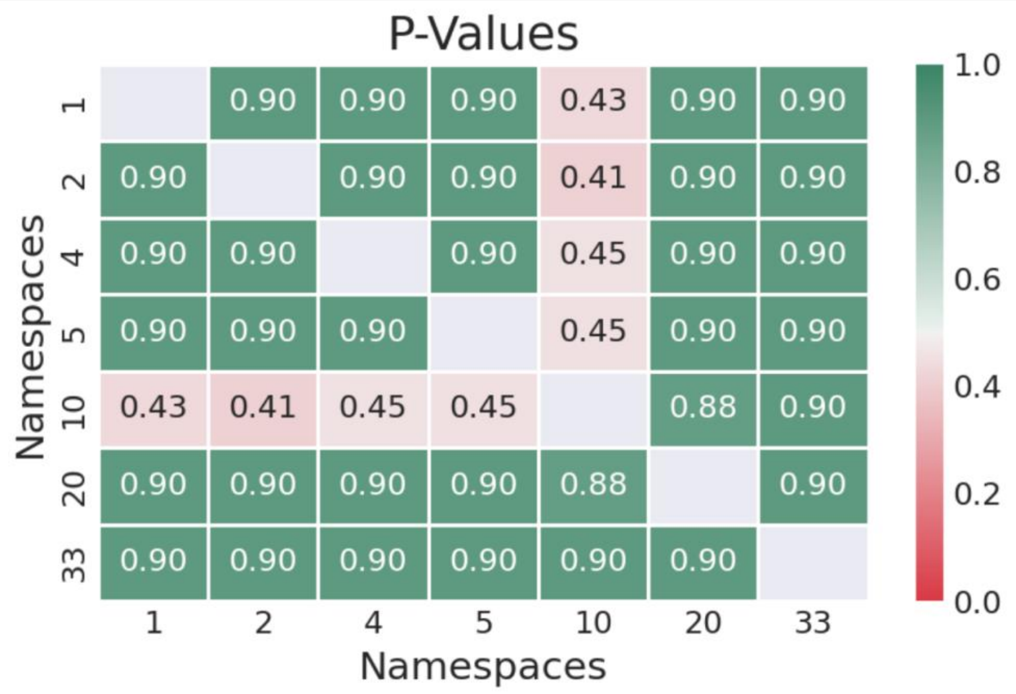


	1	2	4	5	10	20	33	50
1		0.90	0.90	0.90	0.32	0.90	0.30	0.00
2	0.90		0.90	0.90	0.31	0.90	0.29	0.00
4	0.90	0.90		0.90	0.34	0.90	0.33	0.00
5	0.90	0.90	0.90		0.34	0.90	0.33	0.00
10	0.32	0.31	0.34	0.34		0.87	0.90	0.01
20	0.90	0.90	0.90	0.90	0.87		0.85	0.00
33	0.30	0.29	0.33	0.33	0.90	0.85		0.01
50	0.00	0.00	0.00	0.00	0.01	0.00	0.01	

No stolen cycles



	Invalid	Valid
Invalid	0.67	0.33
Valid	0.25	0.75



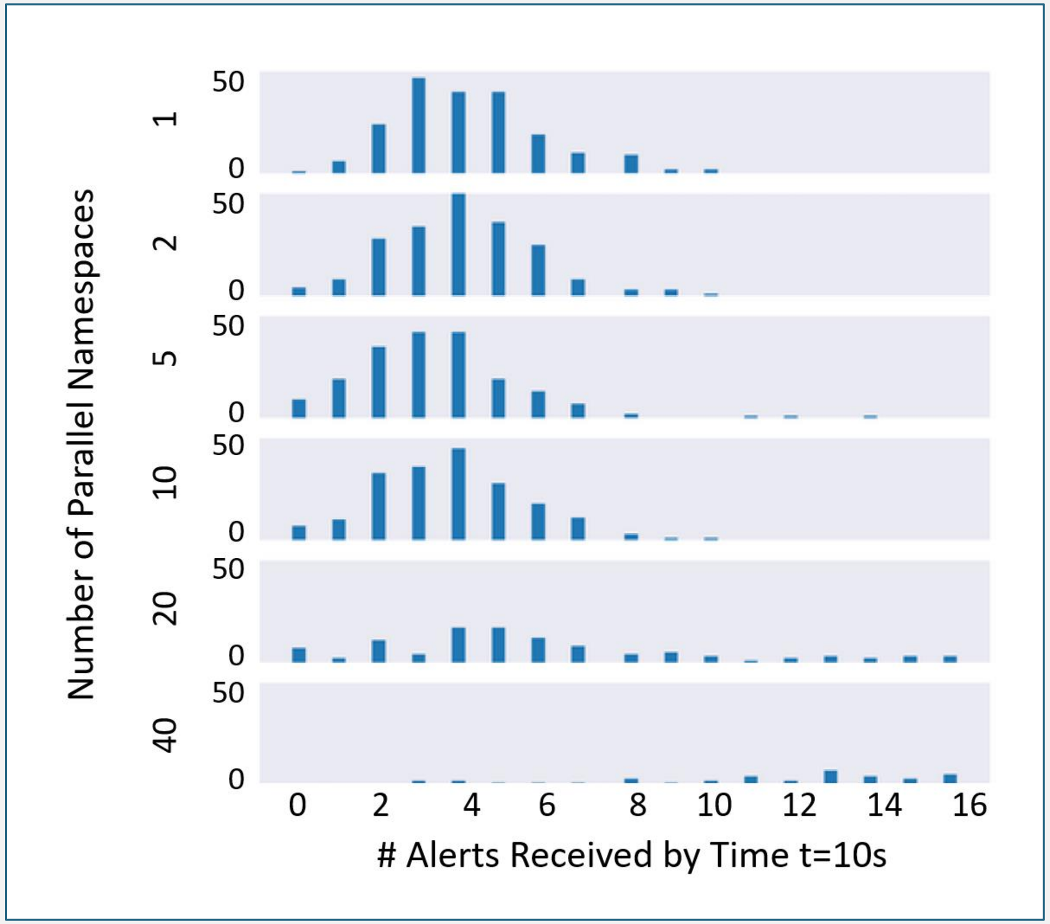
	1	2	4	5	10	20	33
1		0.90	0.90	0.90	0.43	0.90	0.90
2	0.90		0.90	0.90	0.41	0.90	0.90
4	0.90	0.90		0.90	0.45	0.90	0.90
5	0.90	0.90	0.90		0.45	0.90	0.90
10	0.43	0.41	0.45	0.45		0.88	0.90
20	0.90	0.90	0.90	0.90	0.88		0.90
33	0.90	0.90	0.90	0.90	0.90	0.90	

Results – Scenario 2 (Command and Control)

Example Metrics:

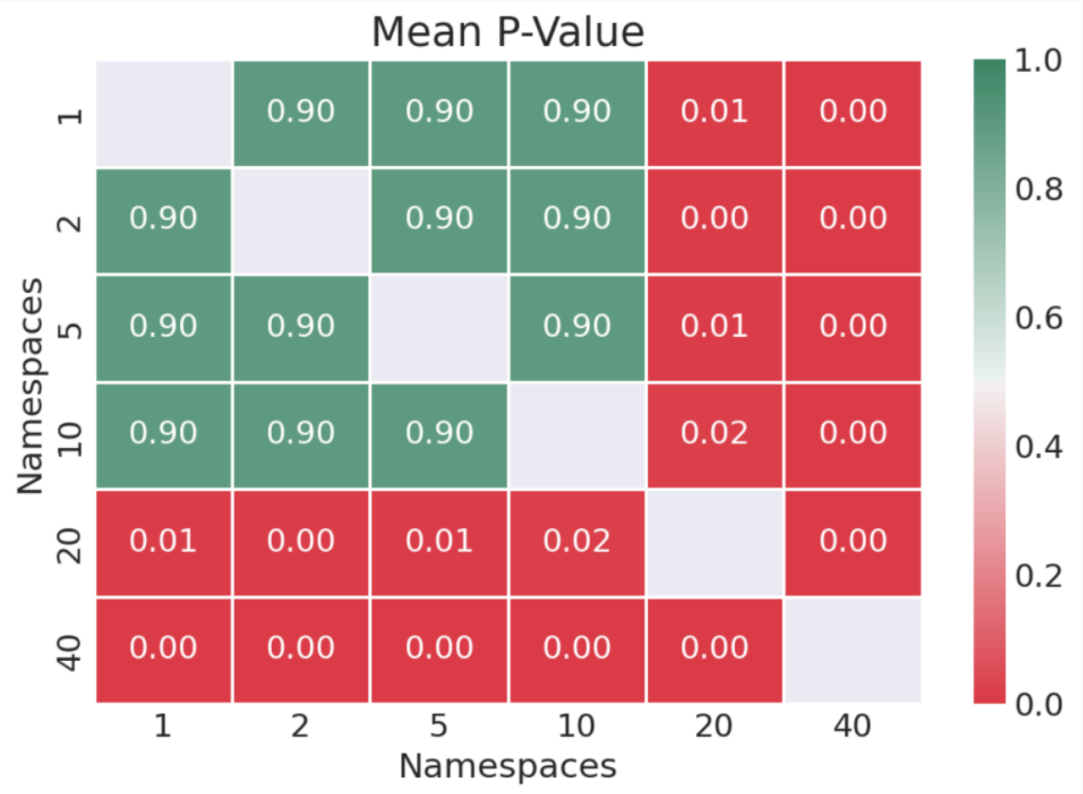
- Stolen Cycles ≤ 1
- Load ≤ 14 Processes
- Interrupts ≤ 2250/s

All replicates

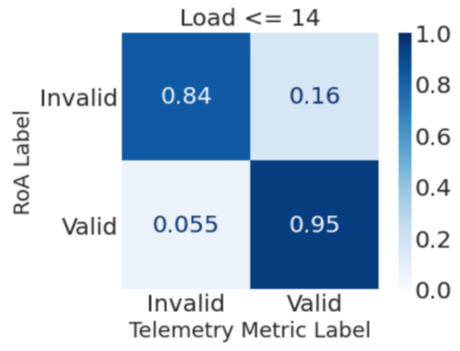


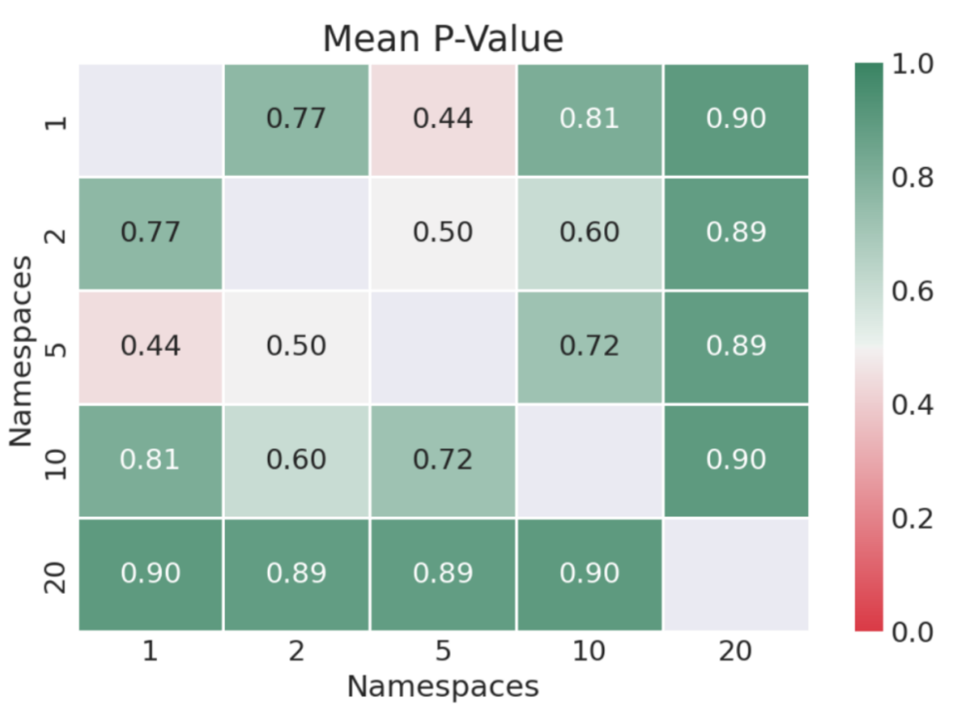
Results – Scenario 2 (Command and Control)

All replicates



Load ≤ 14 Processes





Outcome

Verification helps ensure cyber experiment results can be used to accurately understand real cyber systems

Failure to reproduce cyber experiment results could be due to emulation environment rather than faulty experiment design – the **emulation environment should be verified**

This work successfully demonstrates a generalizable process for resource verification

Thank You!

Discussion Topics

1. Are there other platforms, metrics, and software tools available to perform **verification** of emulation frameworks? (NOT validation)

2. What is suggested for timing or traffic realism and verification of these aspects?

3. How does the nature of the scenario/experiment affect the selection of metrics?

4. Are there other approaches to push resource utilization besides ramping up the number of parallel namespaces?

5. How do we define “Ground Truth”? Is it always the lowest resource usage case?

Discussion Topics

5. What is the best way to identify thresholds? If we take thresholds from the data itself, we are pre-supposing we know when the resources are becoming overutilized. Thoughts on this?

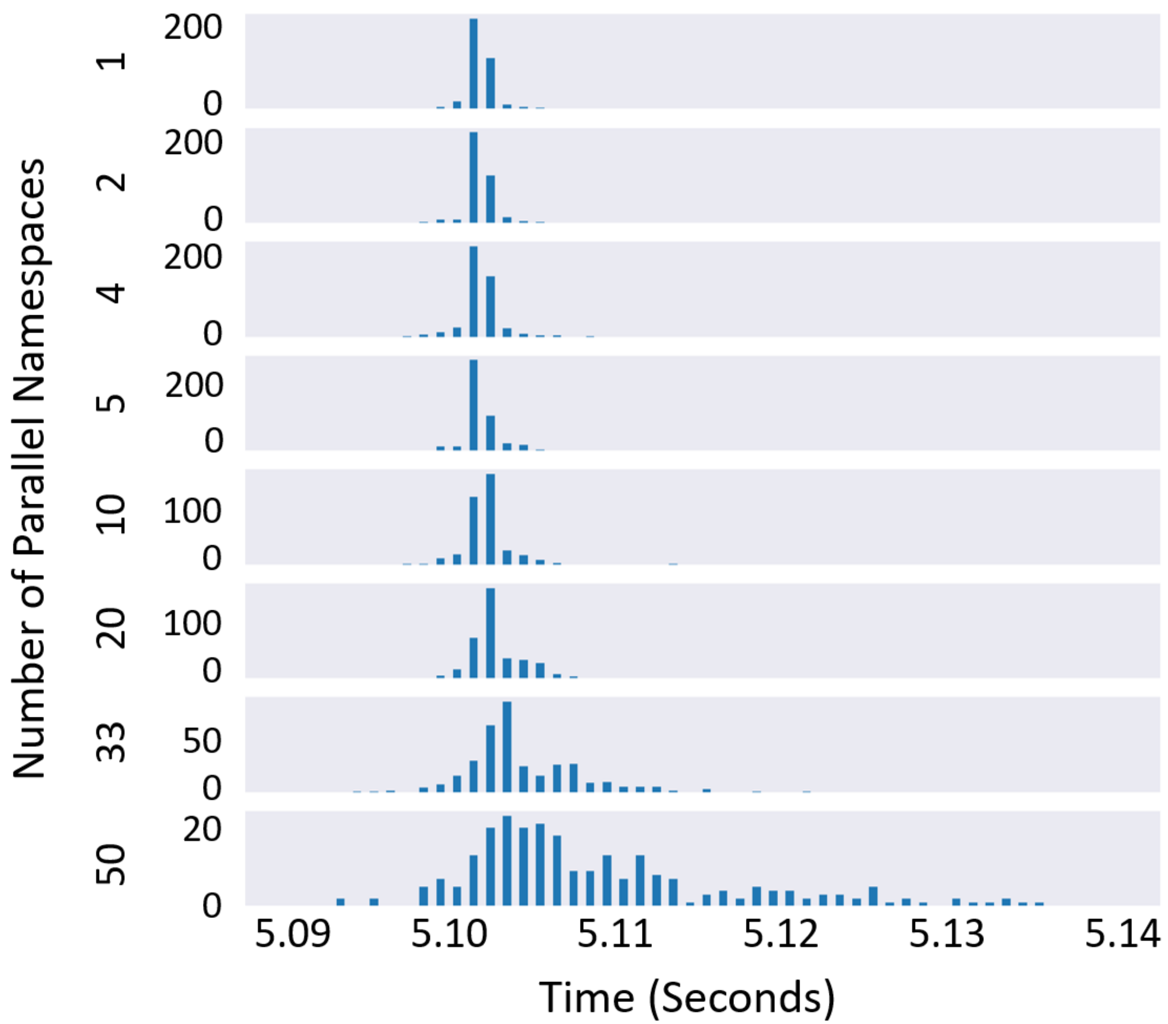
6. We strongly believe in running multiple replicates because there is so much inherent stochasticity in emulated system behavior. This then necessitates the need for statistical comparison across the different test conditions or configurations.
   - Is K-S the best test statistic?
   - Are there other statistical comparisons which should be performed?
   - What if the data is discrete?

7. There are several potential approaches to making a multi-telemetry metric, including various machine learning models. Are there any examples of this of which people are aware?

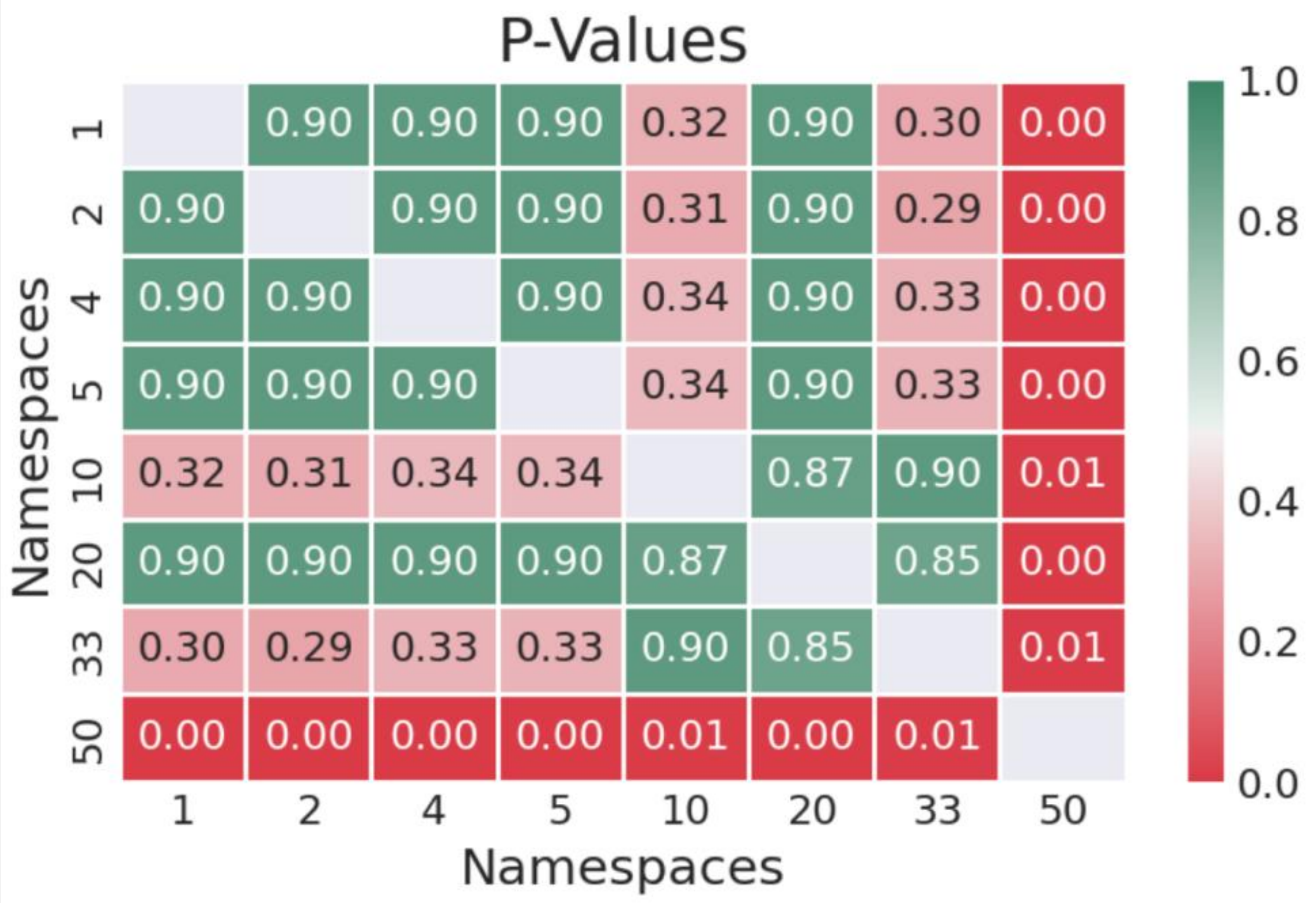
Backup Slides

Images from Paper

Scenario 1 – All Replicates



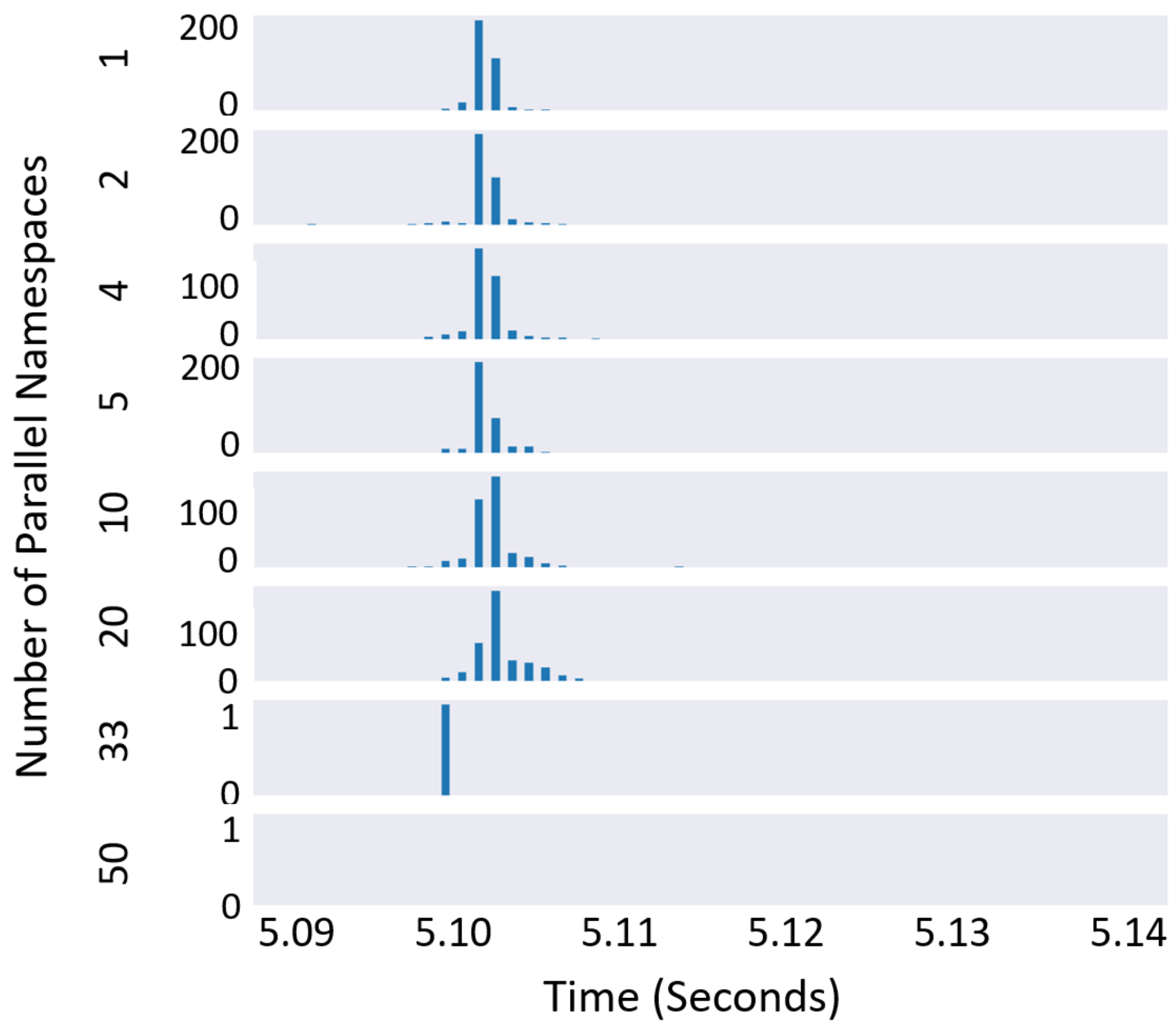
Scenario 1 – All Replicates



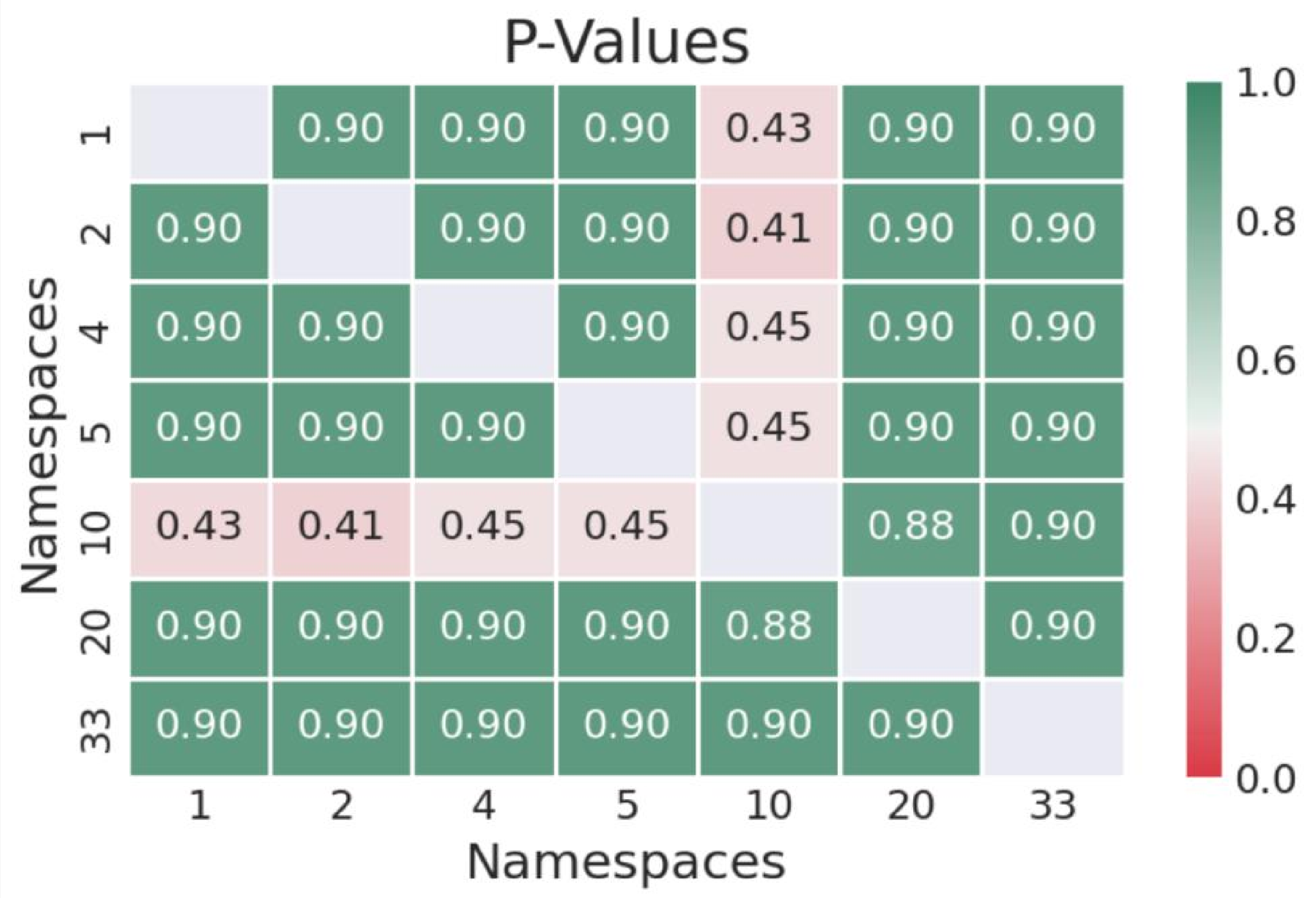
P-Values

Namespaces	1	2	4	5	10	20	33	50
1		0.90	0.90	0.90	0.32	0.90	0.30	0.00
2	0.90		0.90	0.90	0.31	0.90	0.29	0.00
4	0.90	0.90		0.90	0.34	0.90	0.33	0.00
5	0.90	0.90	0.90		0.34	0.90	0.33	0.00
10	0.32	0.31	0.34	0.34		0.87	0.90	0.01
20	0.90	0.90	0.90	0.90	0.87		0.85	0.00
33	0.30	0.29	0.33	0.33	0.90	0.85		0.01
50	0.00	0.00	0.00	0.00	0.01	0.00	0.01	

Scenario 1 – No Stolen Cycles

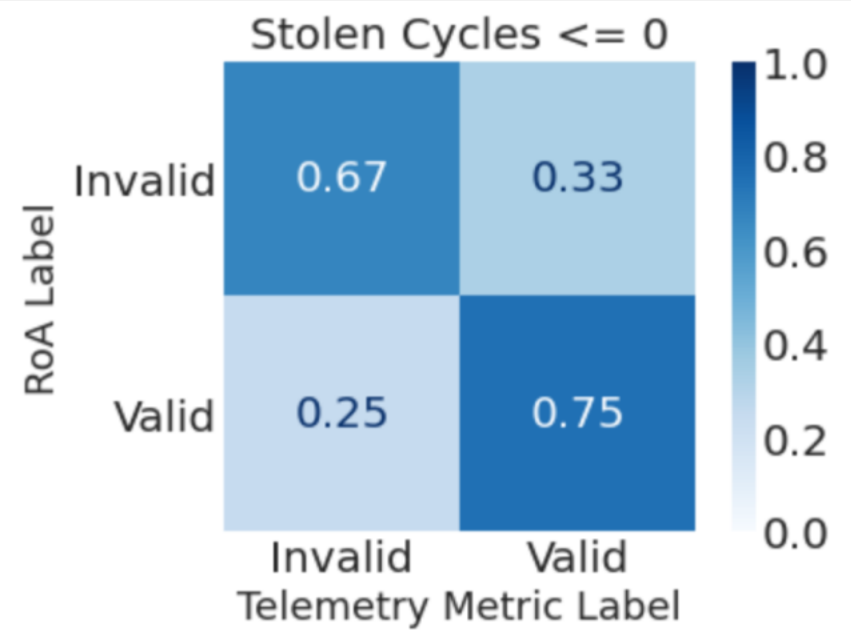


Scenario 1 – No Stolen Cycles



P-Values

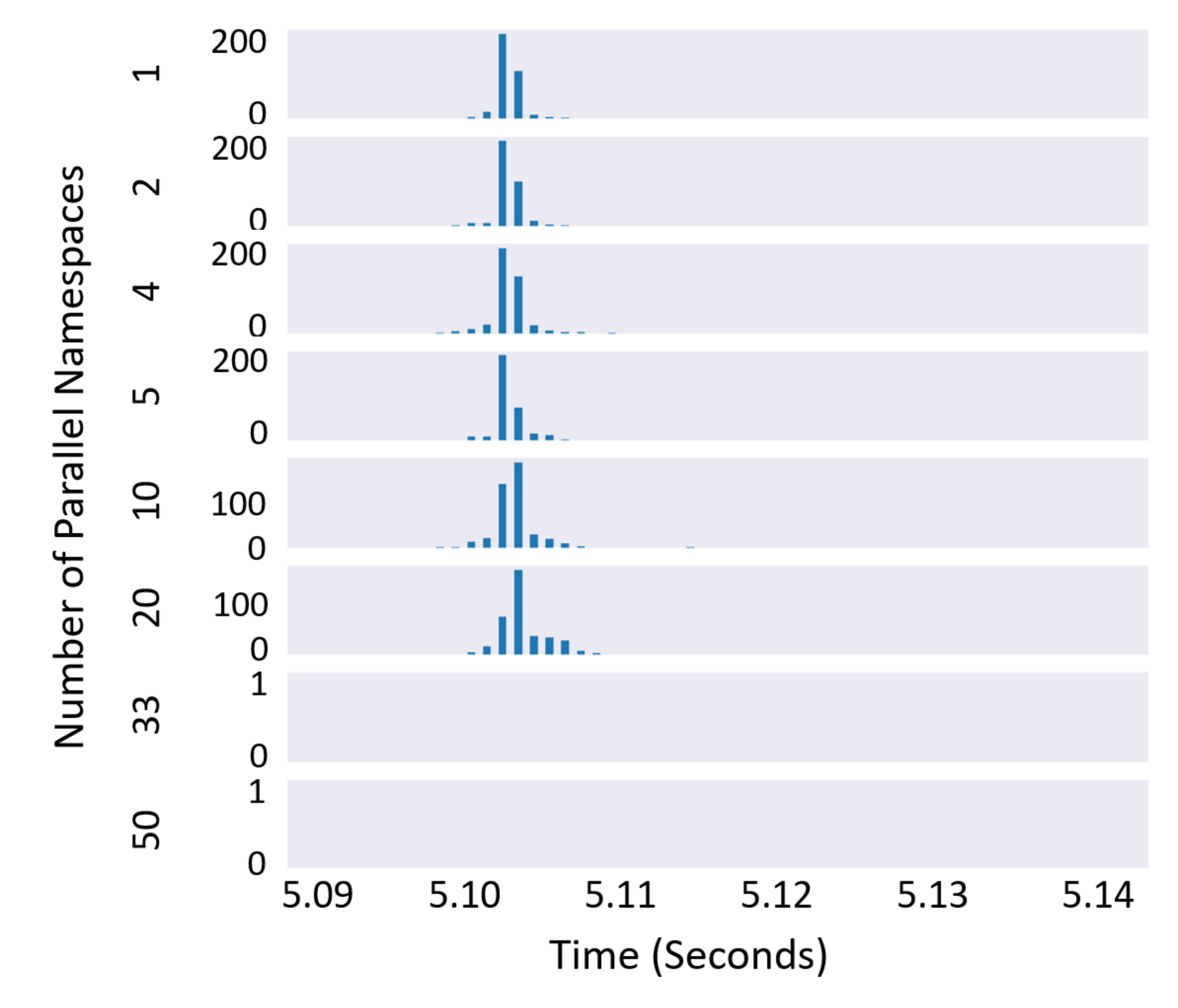
Namespaces	1	2	4	5	10	20	33
1		0.90	0.90	0.90	0.43	0.90	0.90
2	0.90		0.90	0.90	0.41	0.90	0.90
4	0.90	0.90		0.90	0.45	0.90	0.90
5	0.90	0.90	0.90		0.45	0.90	0.90
10	0.43	0.41	0.45	0.45		0.88	0.90
20	0.90	0.90	0.90	0.90	0.88		0.90
33	0.90	0.90	0.90	0.90	0.90	0.90	



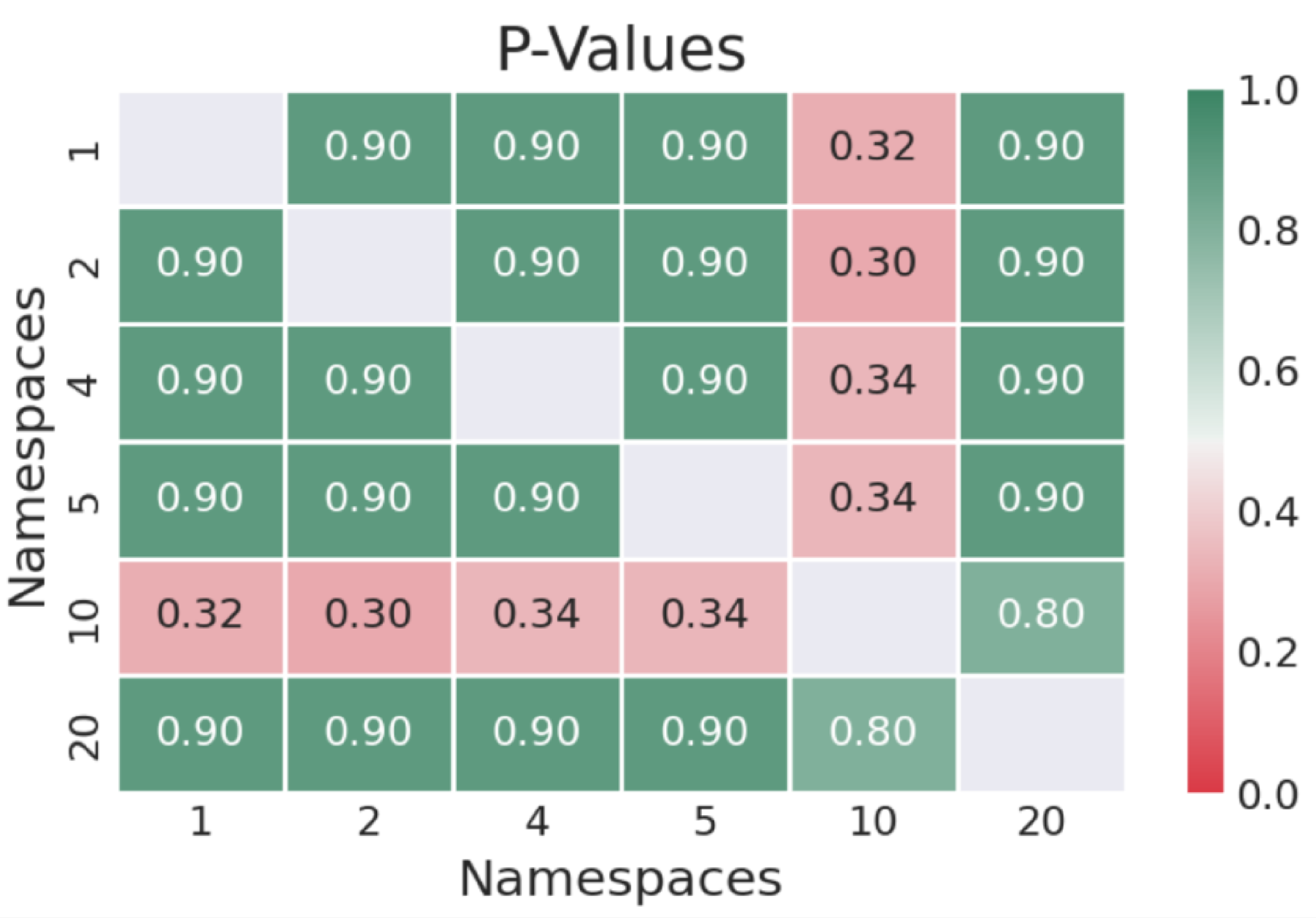
Stolen Cycles <= 0

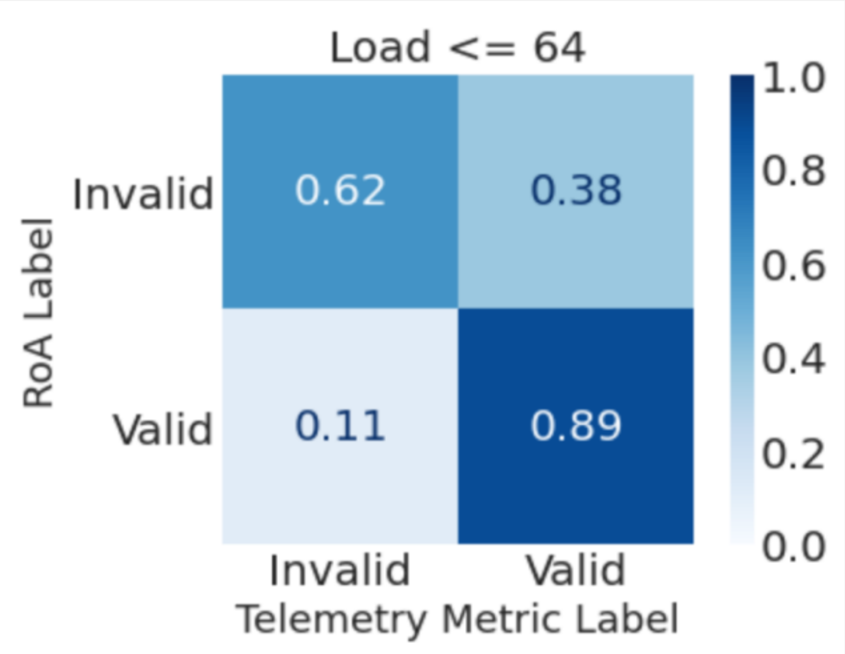
RoA Label \ Telemetry Metric Label	Invalid	Valid
Invalid	0.67	0.33
Valid	0.25	0.75

Scenario 1 – Load ≤ 64 Processes

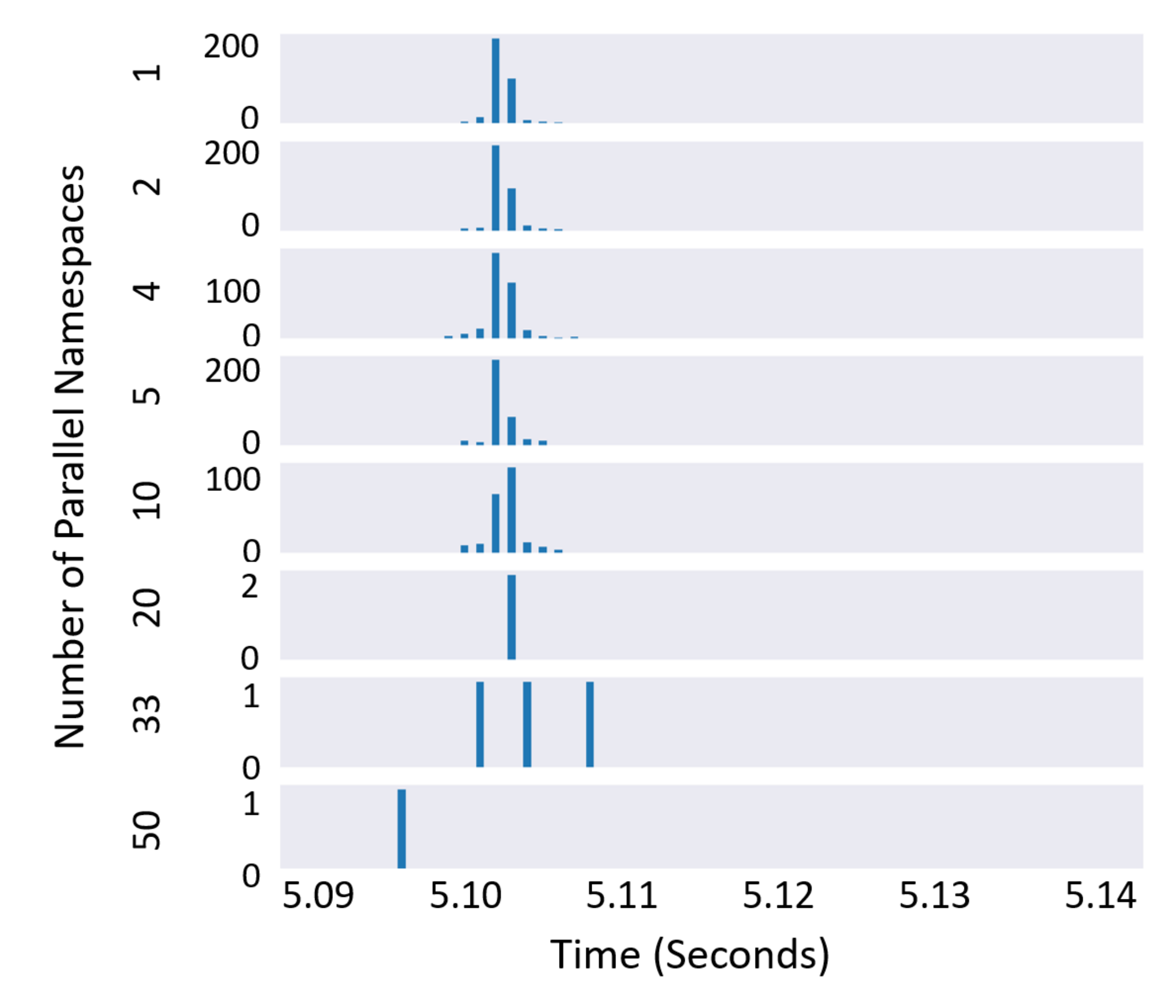


Scenario 1 – Load ≤ 64 Processes

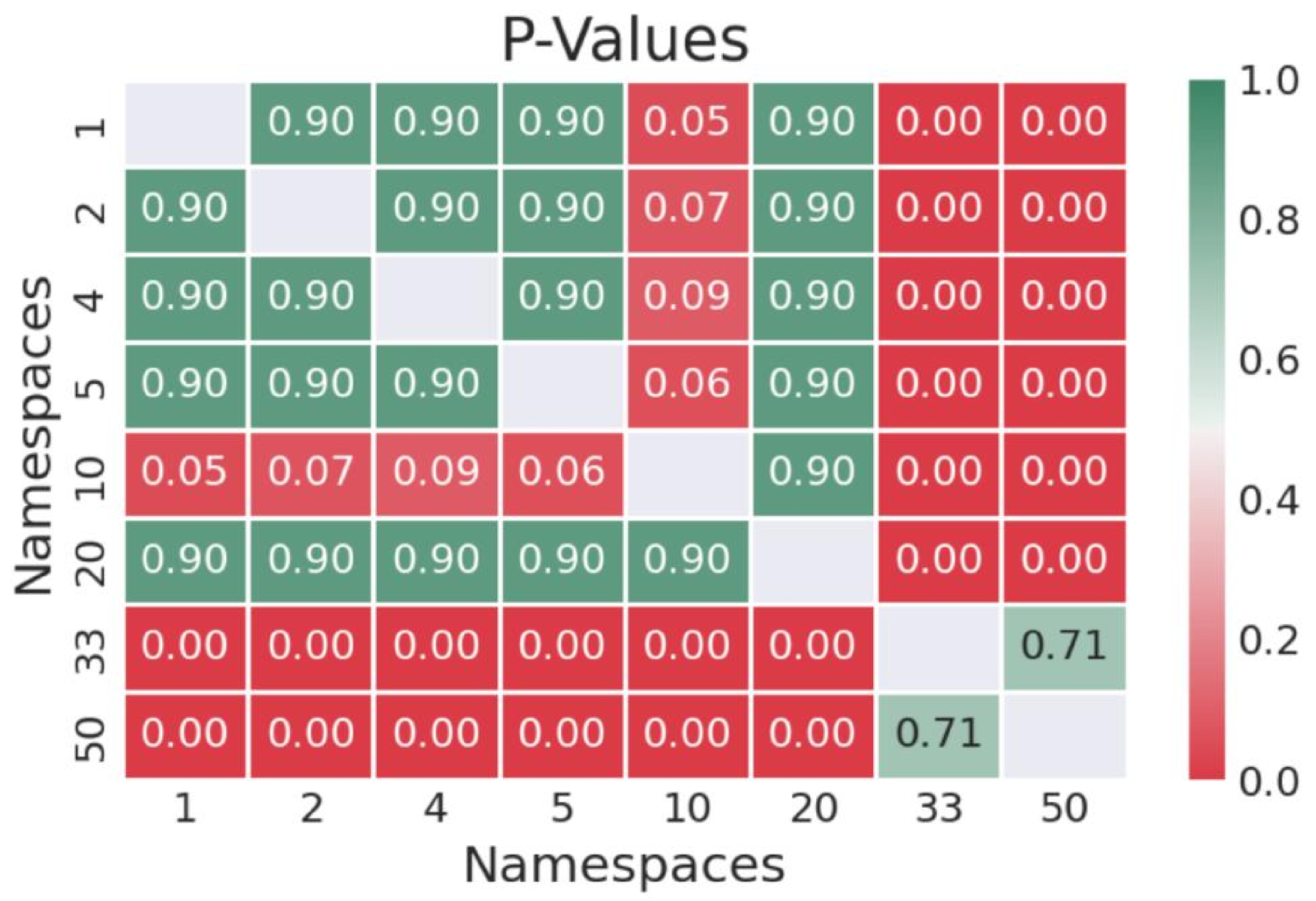




Scenario 1 – Throughput ≥ 250,000 bps

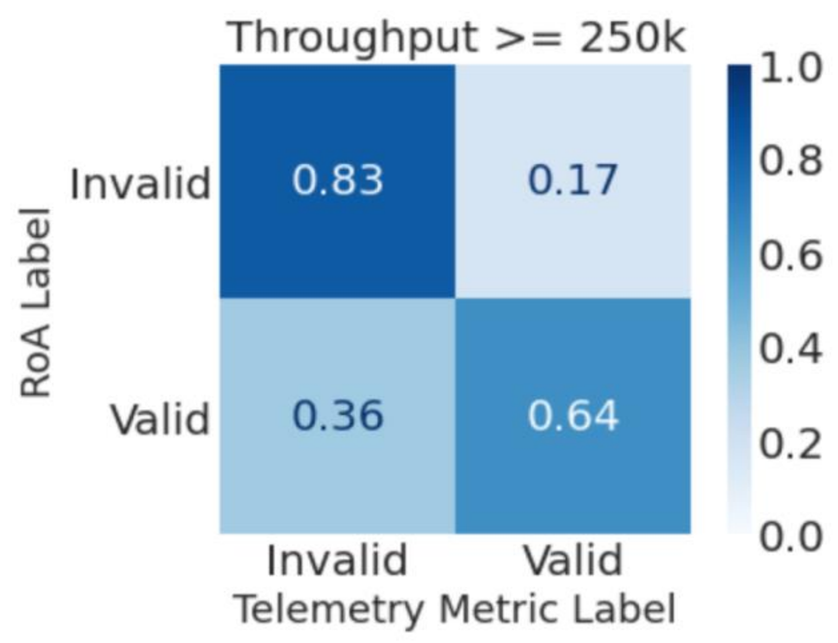


Scenario 1 – Throughput ≥ 250,000 bps



P-Values

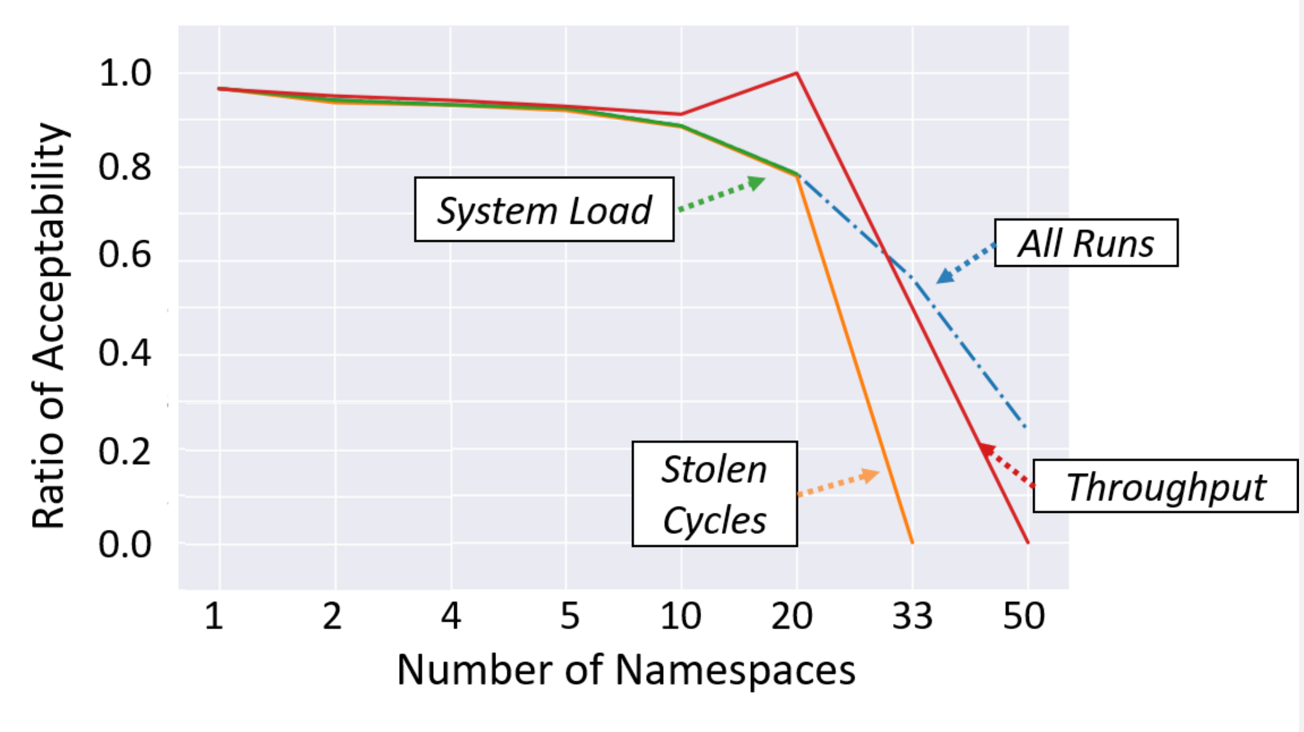
Namespaces	1	2	4	5	10	20	33	50
1		0.90	0.90	0.90	0.05	0.90	0.00	0.00
2	0.90		0.90	0.90	0.07	0.90	0.00	0.00
4	0.90	0.90		0.90	0.09	0.90	0.00	0.00
5	0.90	0.90	0.90		0.06	0.90	0.00	0.00
10	0.05	0.07	0.09	0.06		0.90	0.00	0.00
20	0.90	0.90	0.90	0.90	0.90		0.00	0.00
33	0.00	0.00	0.00	0.00	0.00	0.00		0.71
50	0.00	0.00	0.00	0.00	0.00	0.00	0.71	

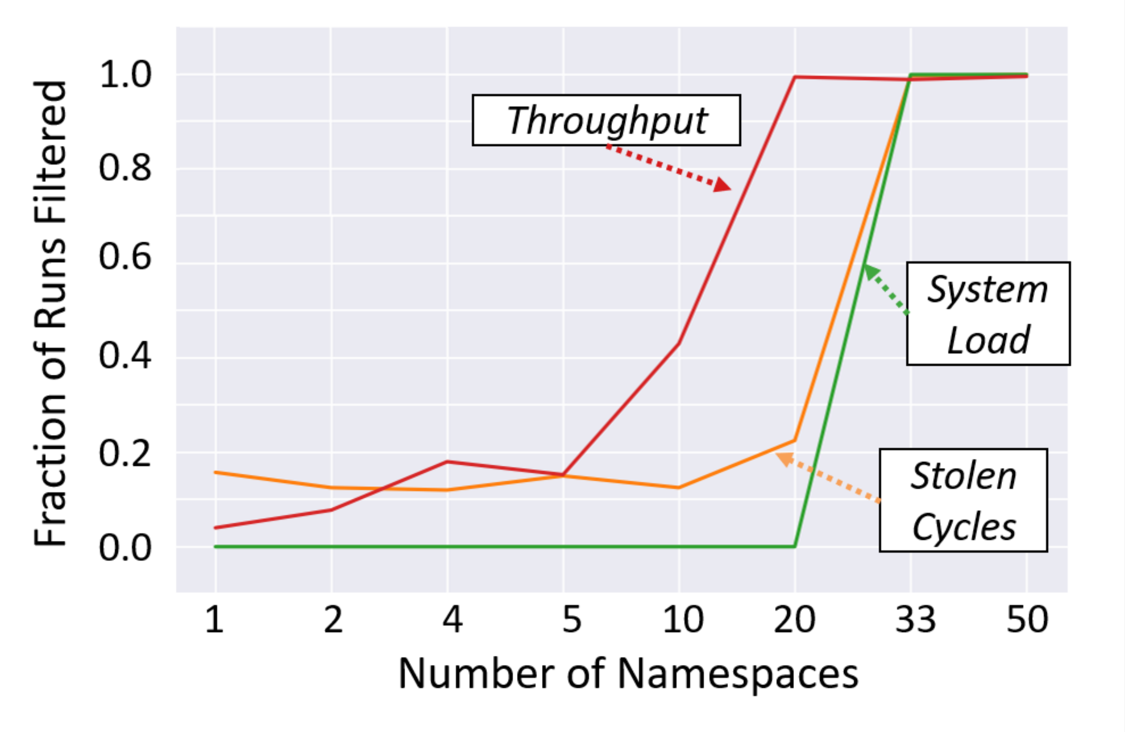


Throughput >= 250k

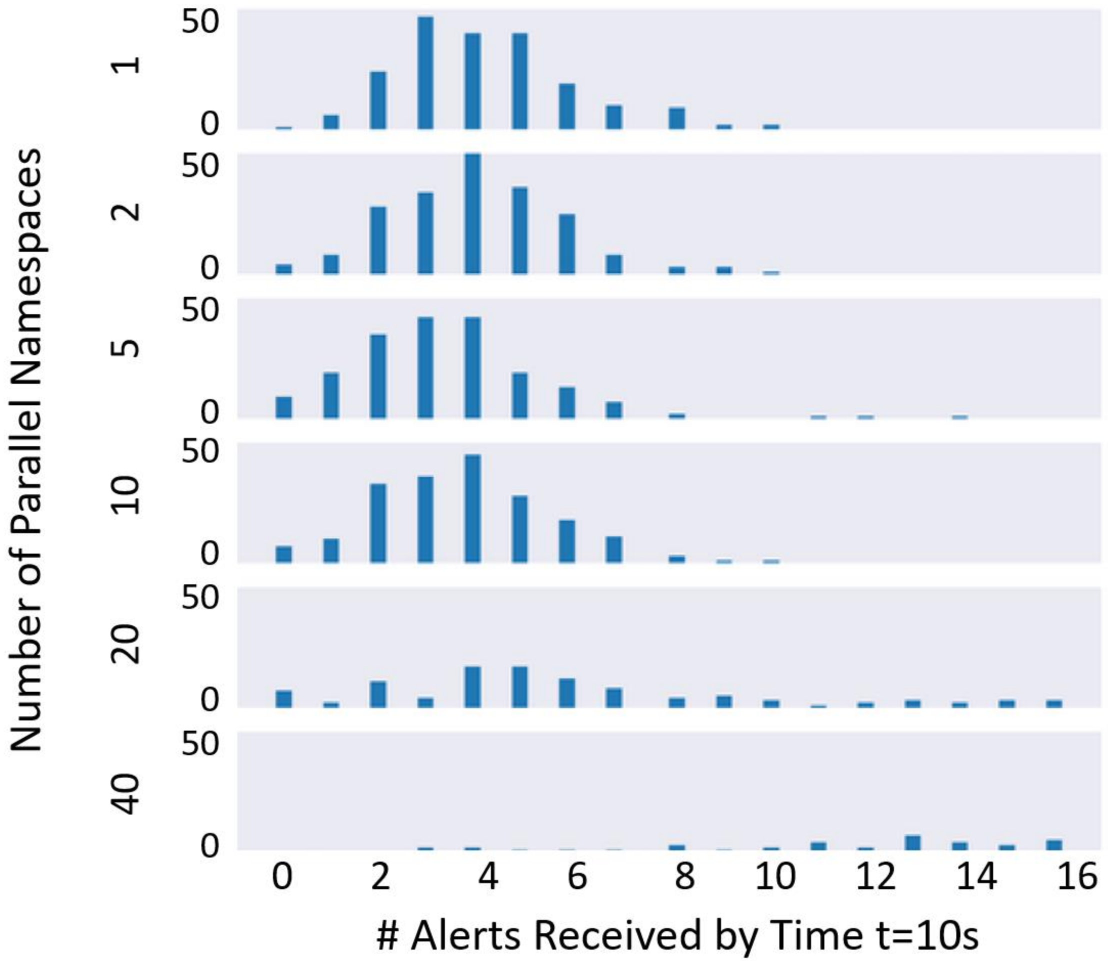
RoA Label \ Telemetry Metric Label	Invalid	Valid
Invalid	0.83	0.17
Valid	0.36	0.64

Scenario 1 – RoA and Runs Filtered

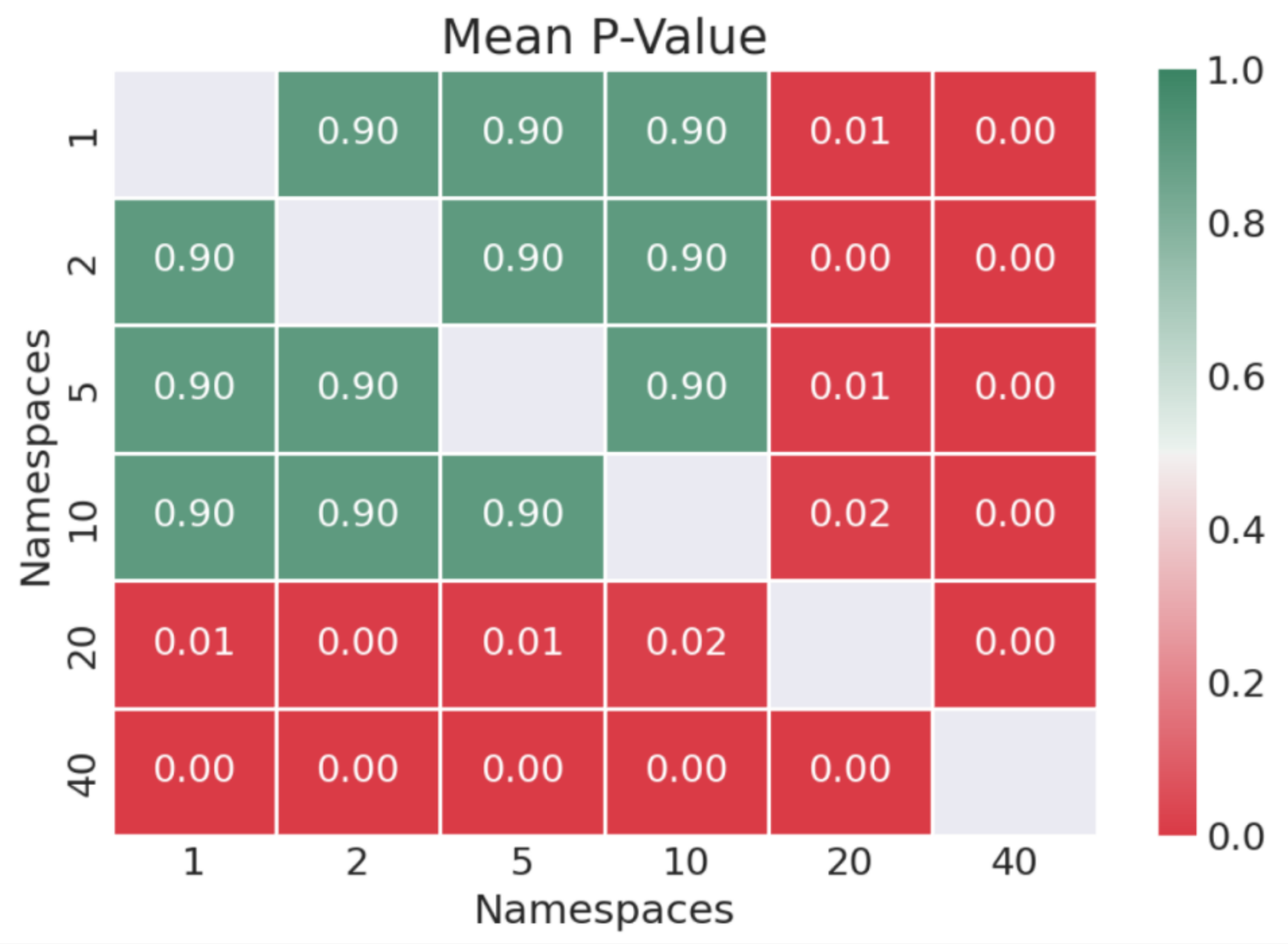




Scenario 2 – All Replicates



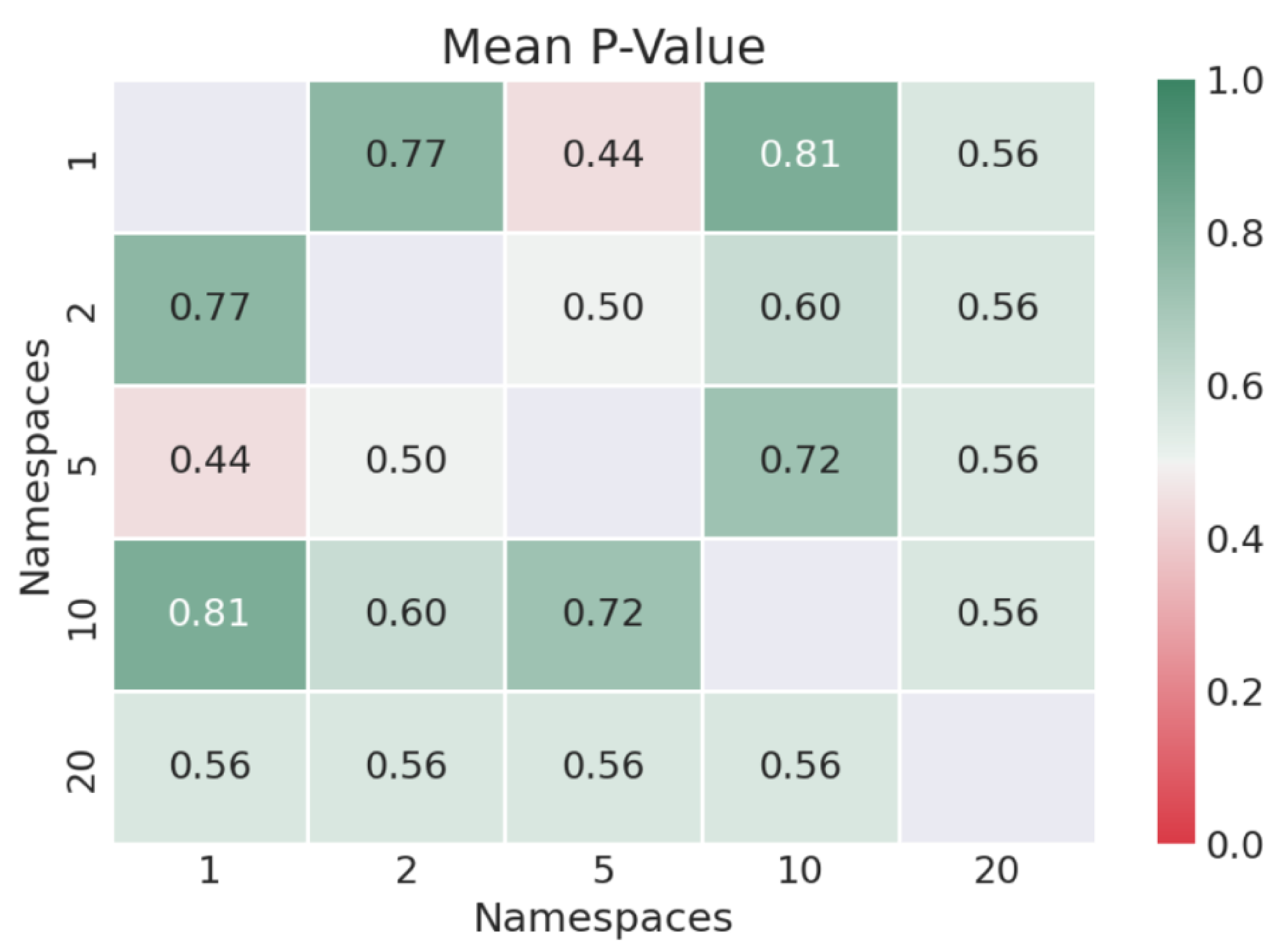
Scenario 2 – All Replicates

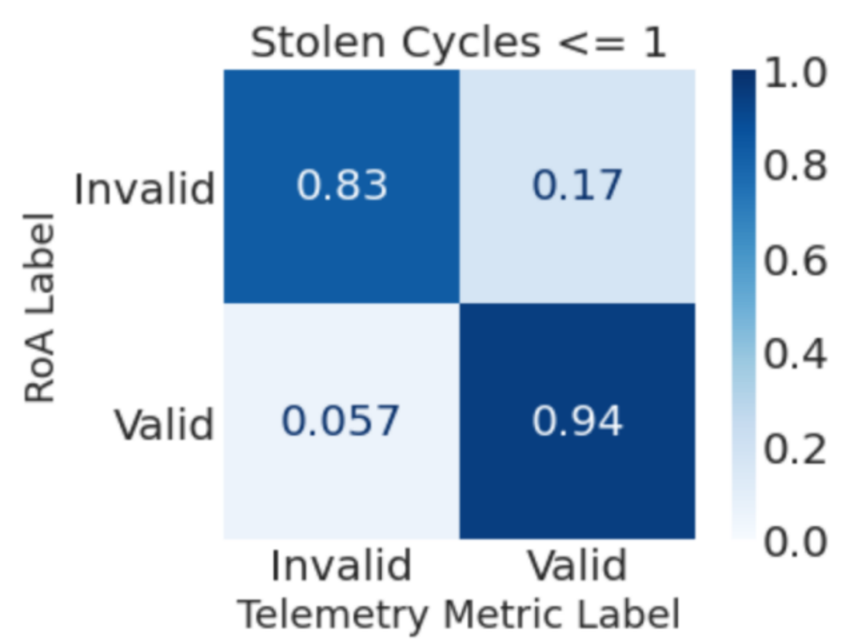


Mean P-Value

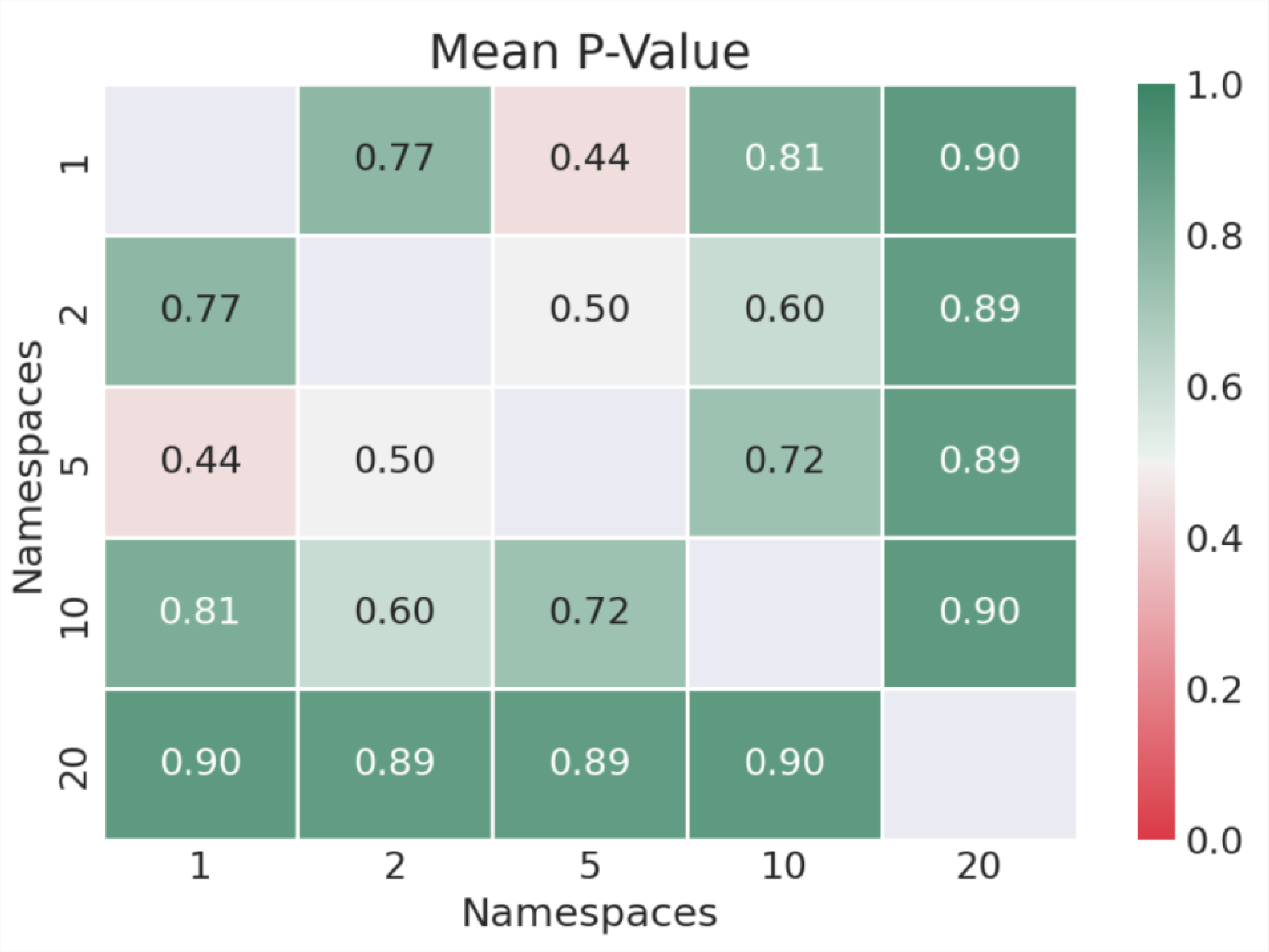
Namespaces	1	2	5	10	20	40
1		0.90	0.90	0.90	0.01	0.00
2	0.90		0.90	0.90	0.00	0.00
5	0.90	0.90		0.90	0.01	0.00
10	0.90	0.90	0.90		0.02	0.00
20	0.01	0.00	0.01	0.02		0.00
40	0.00	0.00	0.00	0.00	0.00	

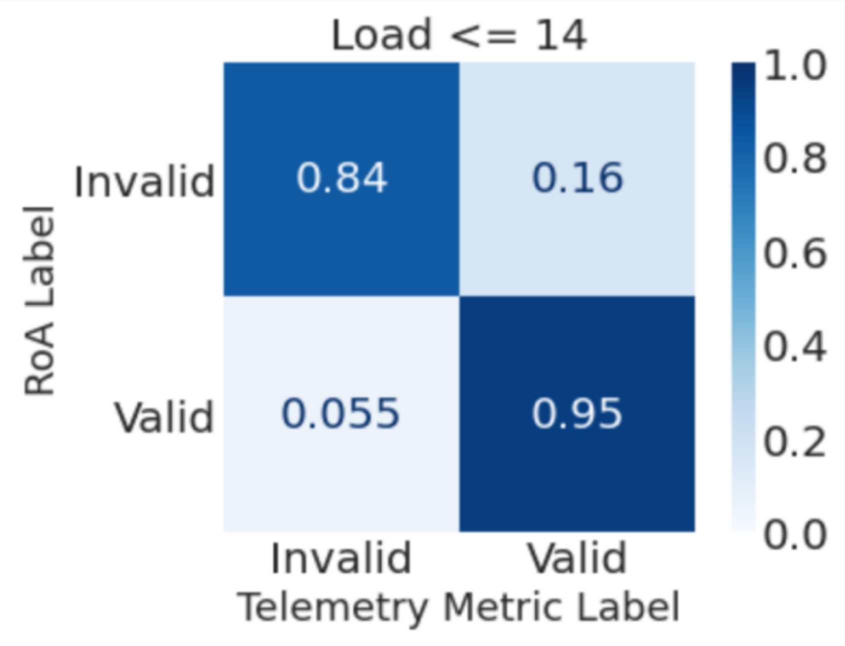
Scenario 2 – Stolen Cycles ≤ 1



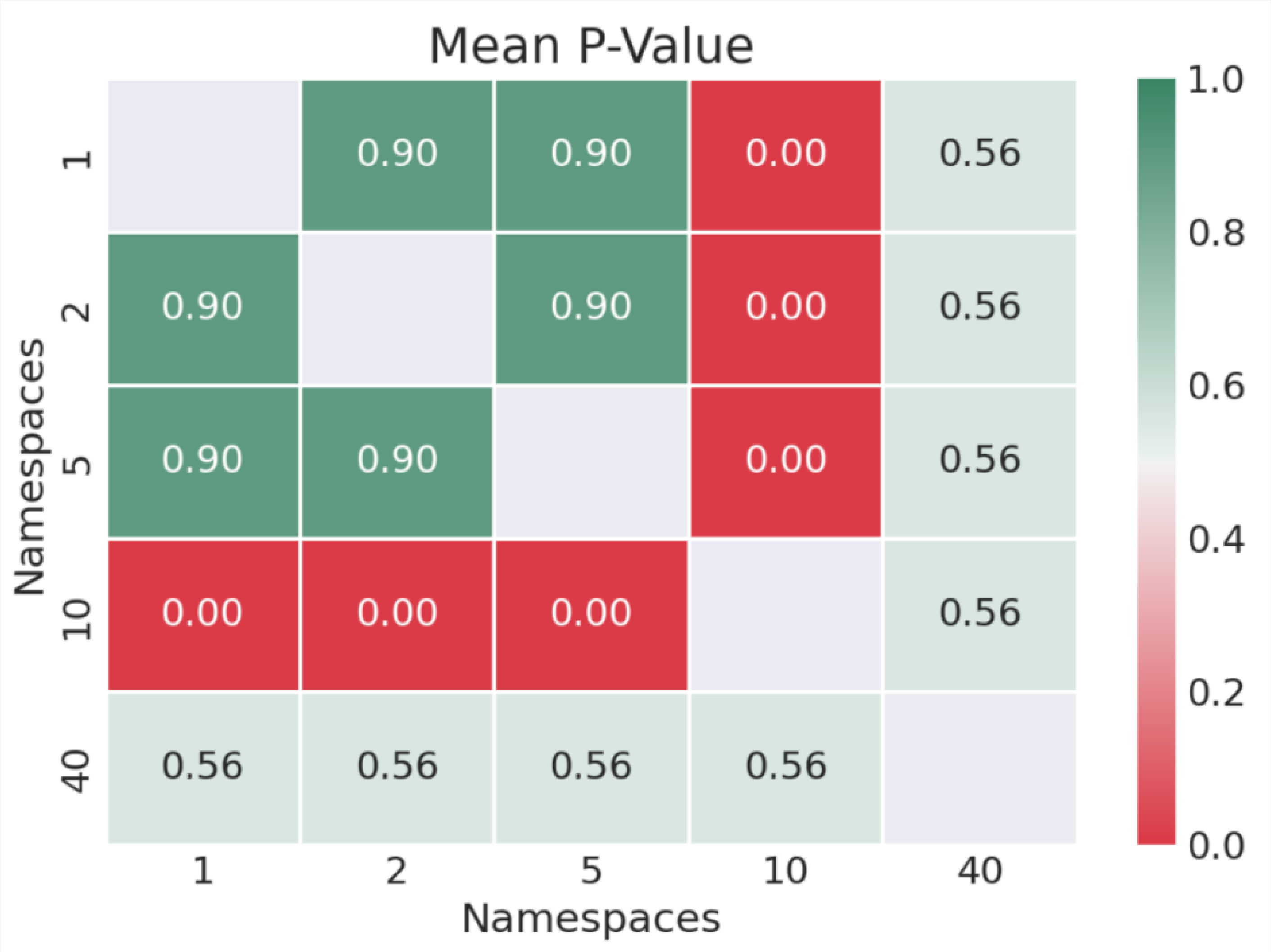


Scenario 2 – Load ≤ 14 Processes

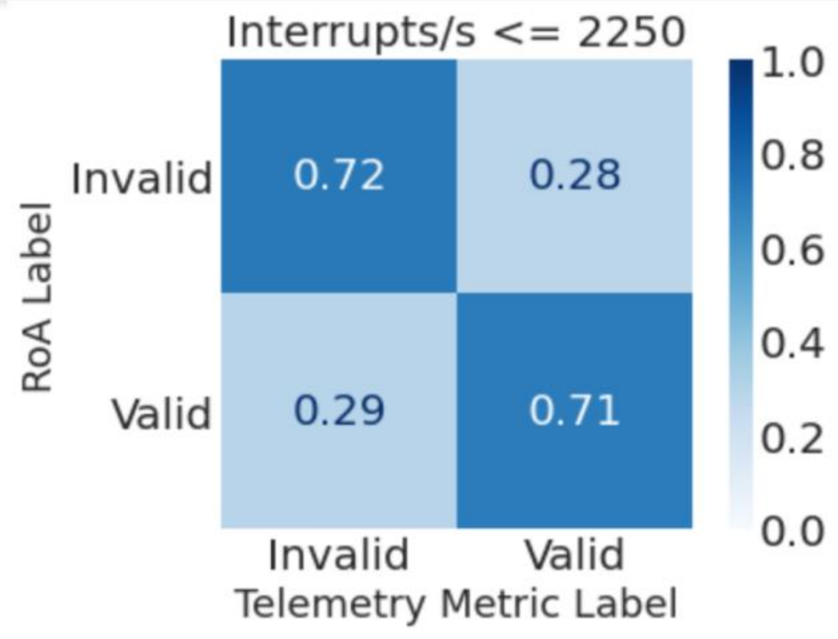




Scenario 2 – Interrupts per Second ≤ 2250



Namespaces	1	2	5	10	40
1		0.90	0.90	0.00	0.56
2	0.90		0.90	0.00	0.56
5	0.90	0.90		0.00	0.56
10	0.00	0.00	0.00		0.56
40	0.56	0.56	0.56	0.56	



RoA Label \ Telemetry Metric Label	Invalid	Valid
Invalid	0.72	0.28
Valid	0.29	0.71