Mid-Semester Presentation - Toward a Resilient U.S. Power Grid

Valentina Alzate, Ben Cillie, Caroline Reynolds, Megan Rosen, Daniel Weber

Project Goal

The DSN Lab made a system

We’re trying to break the system

Project Goal

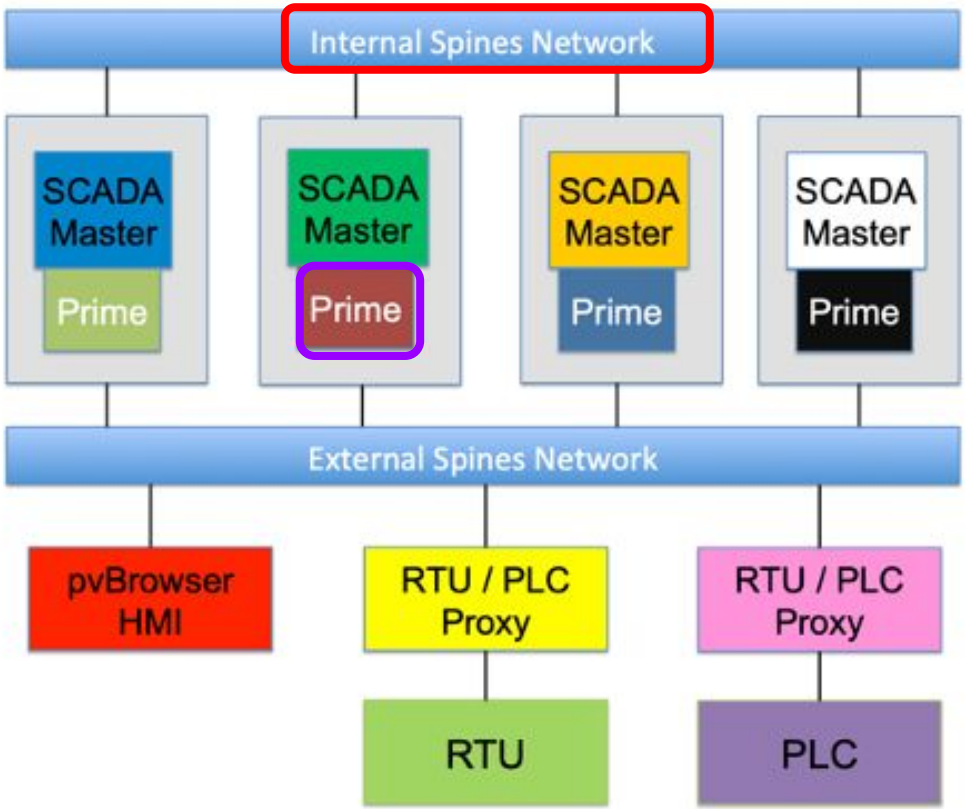
A little more formally:

The Spire System was created to reliably control the power grid. Our goal as a team is to attack the system and find a way to break it or slow it below specified speeds.

20 ms → 33 ms

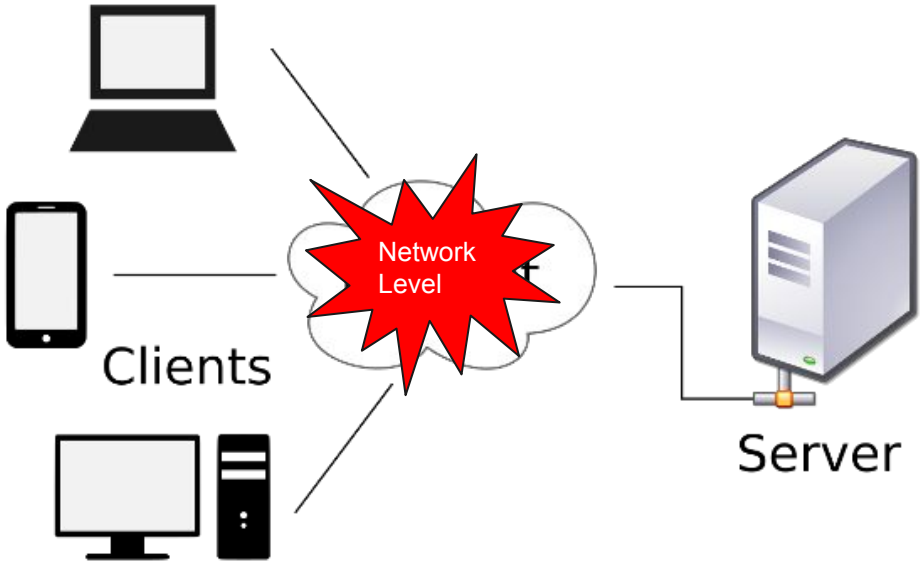
The Spire System

The goal of Spire is to create an intrusion-tolerant reliable system to operate the power grid that is exposed to the open internet.



Spines

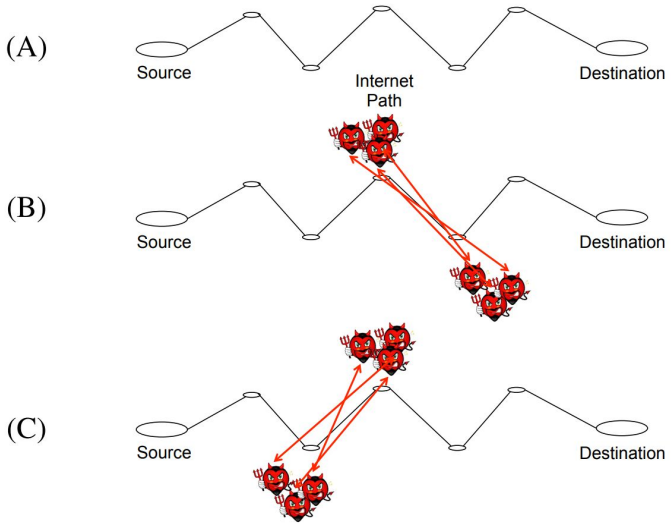
Network-Level Attacks



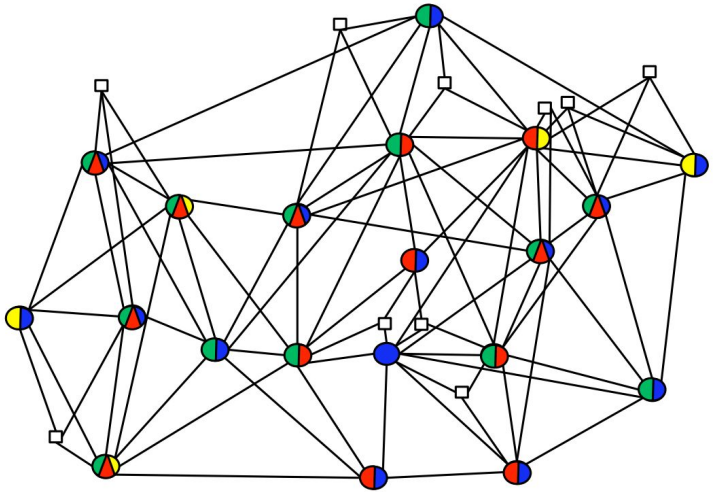
Spines

An Intrusion Tolerant Network

Conventional Infrastructure



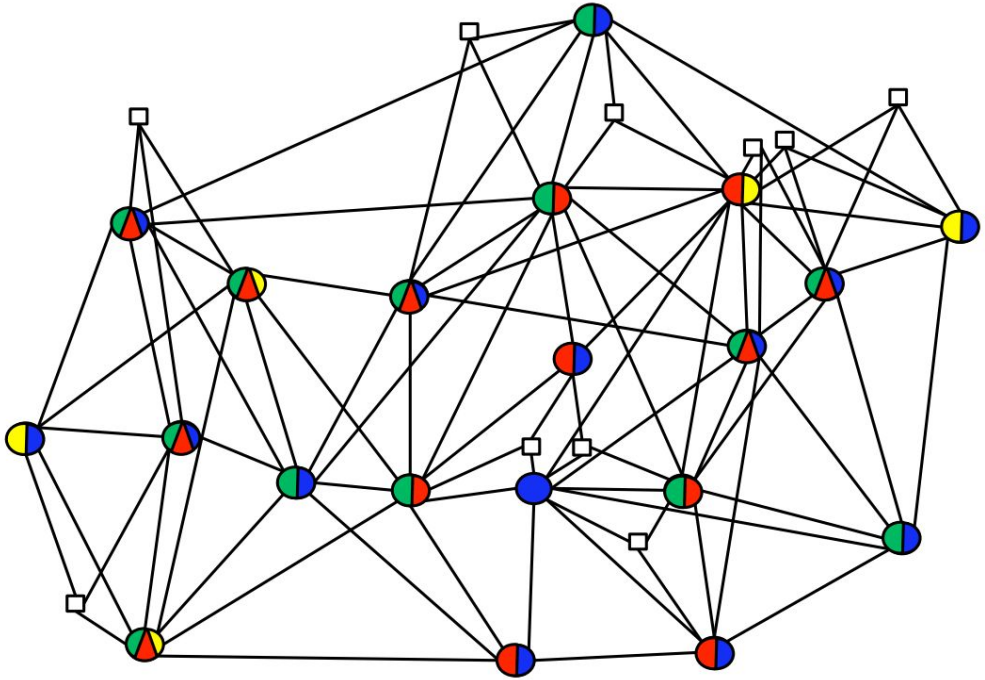
- Overlay network built on top of existing IP infrastructure
  - Multi-homing



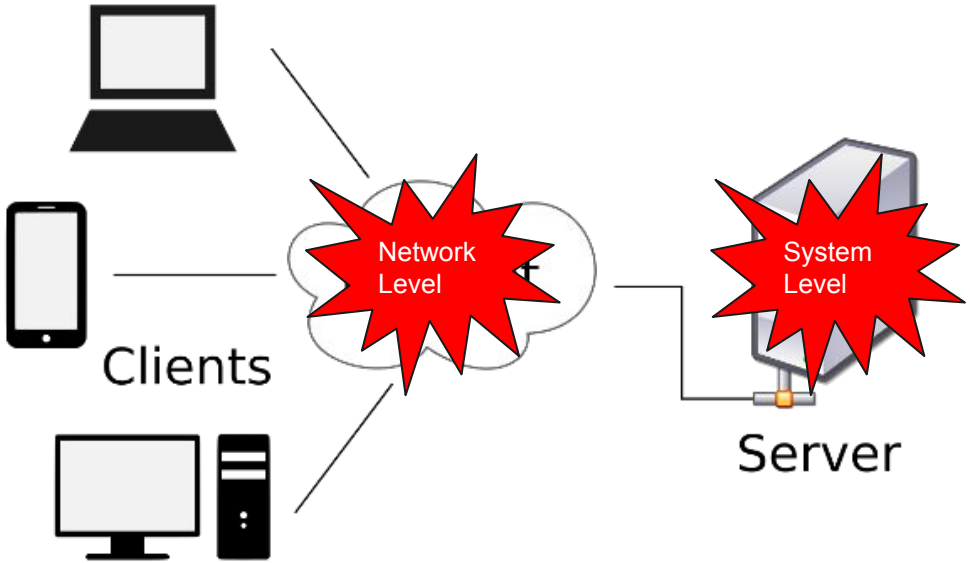
Spines

An Intrusion Tolerant Network

- Intrusion Tolerance
  - Fairness Principle
  - Flooding



System-Level Attacks



Prime

How to Create a Reliable System?

- Problems to Solve:
  - What happens if our server goes down?
  - What happens if our server is compromised by an attacker?

The Answer: REDUNDANCY

Prime

Consensus Algorithms

- We seek 3 things:
  - 1) Termination
  - 2) Integrity
  - 3) Agreement
- Prime guarantees that we achieve these properties in a timely manner.
  - Older protocols did not enforce a timeliness condition

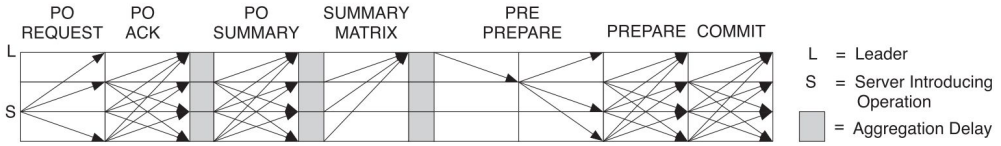
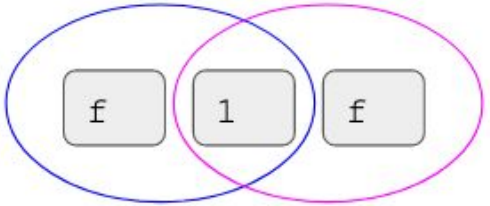


Fig. 3. Operation of Prime with a malicious leader that performs well enough to avoid being replaced ($f = 1$).

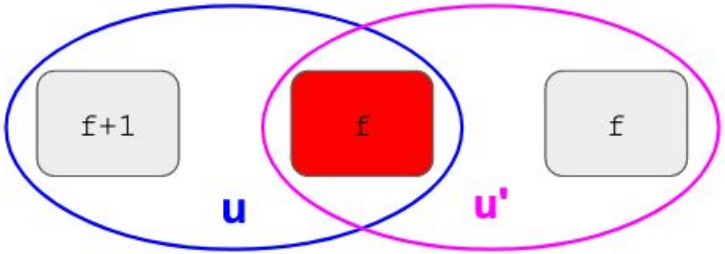
Prime

How many replicas do we need?

- Fail Stop Failure
  - A replica becomes completely unresponsive
- Handling Fail Stop Failure: $N \geq 2f + 1$



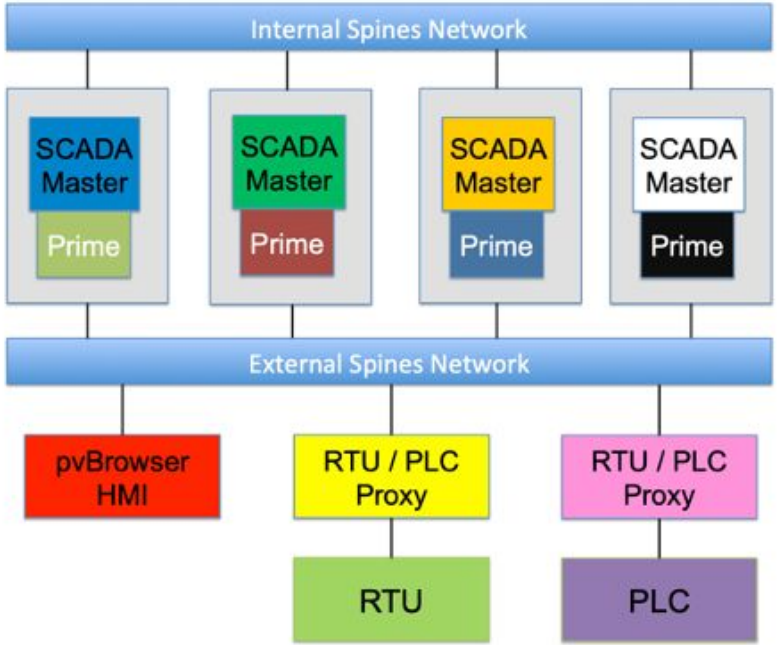
- Byzantine Failure
  - A replica responds in any unexpected way
  - Harder to account for in a system
- Handling Fail Stop Failure: $N \geq 3f + 1$



TL;DR - The Spire System

- Spines creates an intrusion-tolerant reliable network that isn't vulnerable to conventional network attacks (DOS, MITM, BGP Hijacking)

- Prime ensures that our distributed system maintains correctness while executing commands in a timely manner.



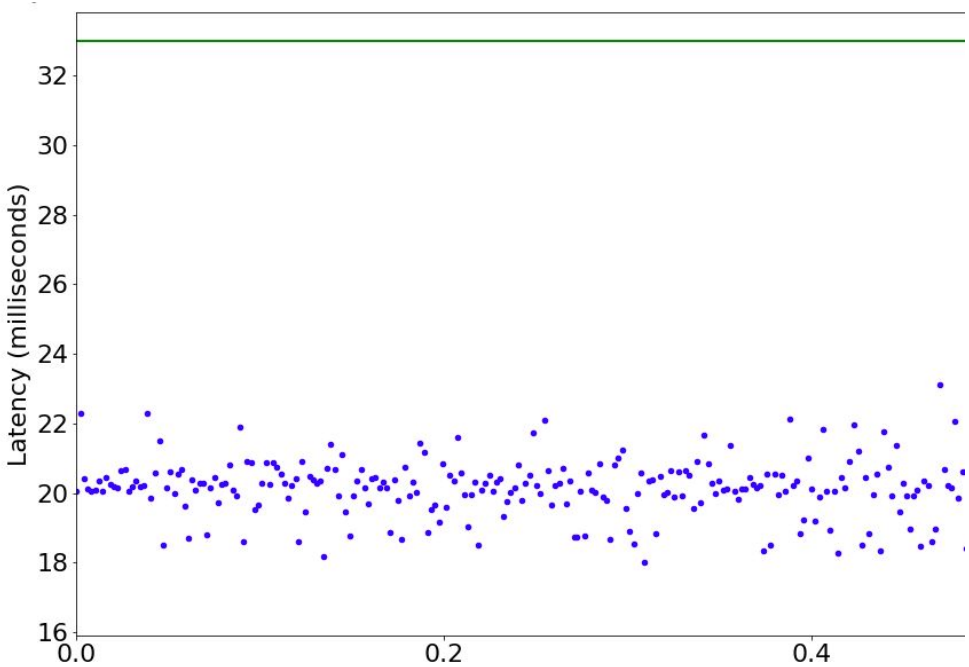
Testing and Benchmarking

- Testing from the Prime perspective
- We measure **latency:** *the time it takes for some data to get to its destination across the network*
- Modified Prime Client Program
  - Records timestamp, latency values
  - Exports data into CSV
- Cases we benchmarked:
  - Pure: SPIRE, Prime
  - Failstop: SPIRE, Prime
  - Byzantine: Prime
  - Byzantine + Failstop: Prime

Live Demo

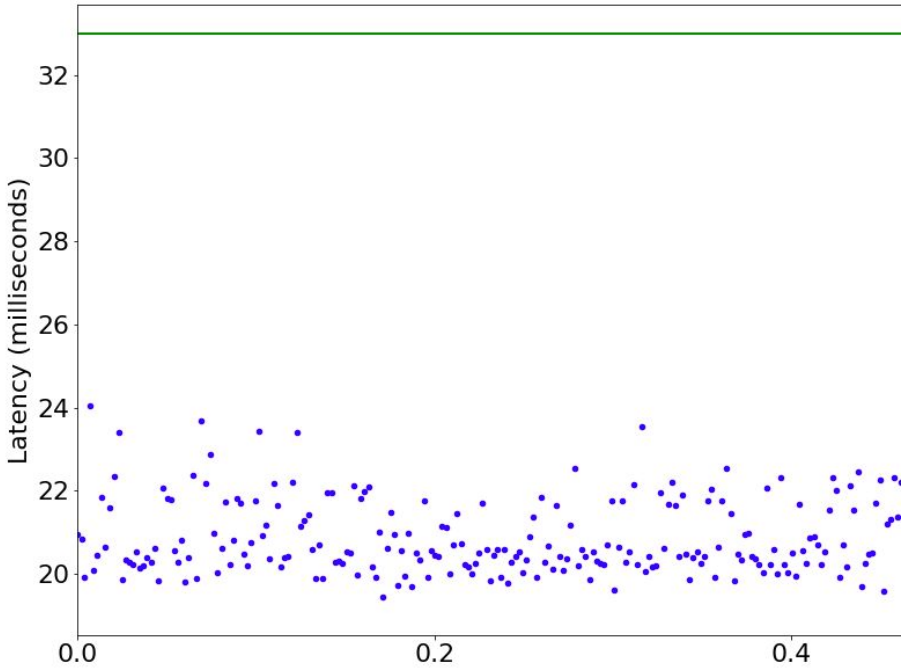
Baseline (SPIRE)

```
Min Latency: 15.909150000000002
Max Latency: 33.81385
Average Latency: 20.05959839357429
```



Failstop (SPIRE)

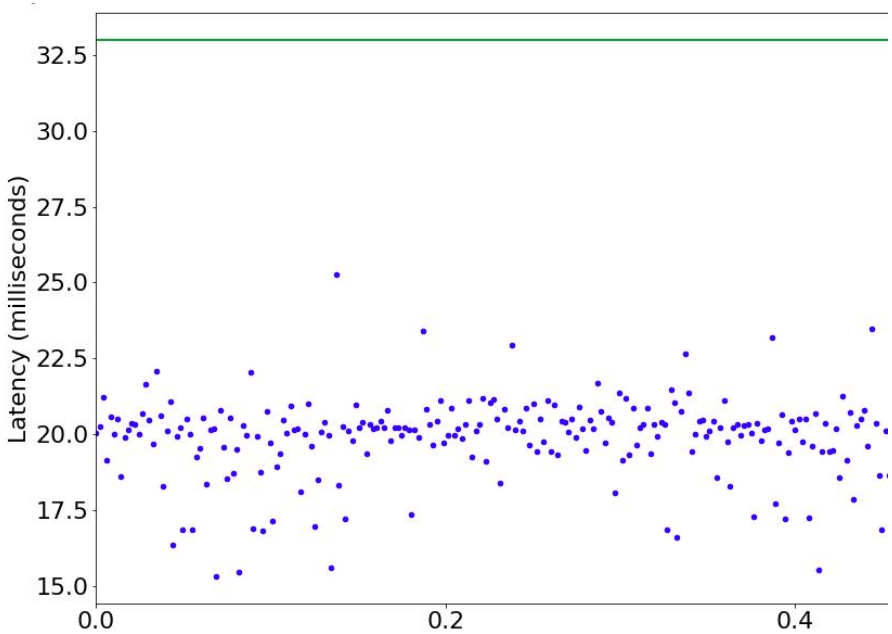
```
Min Latency: 18.5282
Max Latency: 33.68913333333333
Average Latency: 20.89085140562251
```



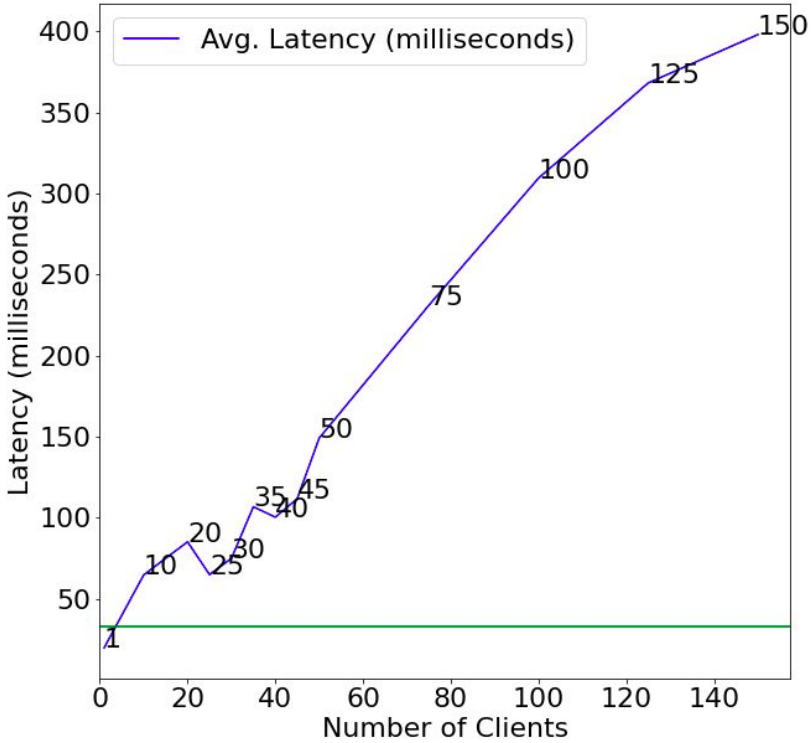
Baseline (Prime)

```
Min Latency: 14.43215
Max Latency: 33.88418333333333
Average Latency: 19.94734605087014
```

1 Client



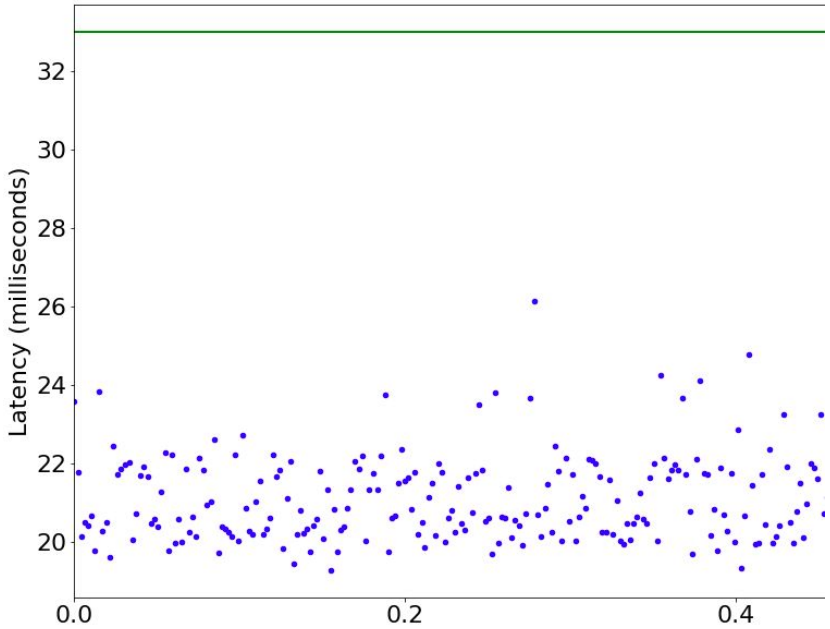
Avg Latency vs. # of Clients



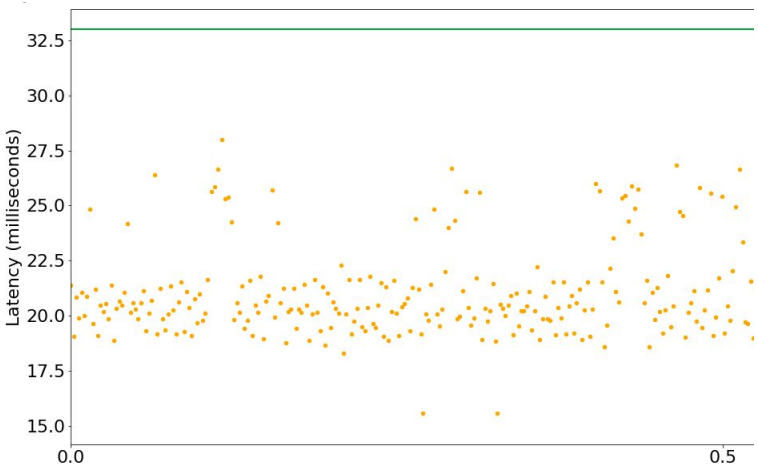
Failstop (Prime)

Min Latency: 18.581400000000002
Max Latency: 33.6866
Average Latency: 21.105781124498005

1 Client

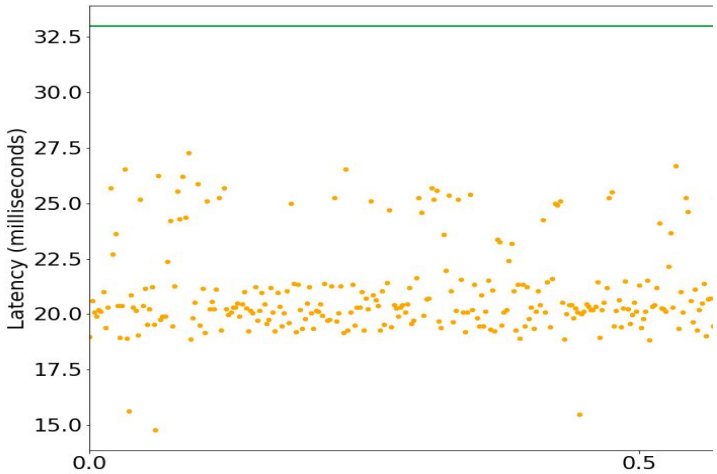


No Sequence Update Attack



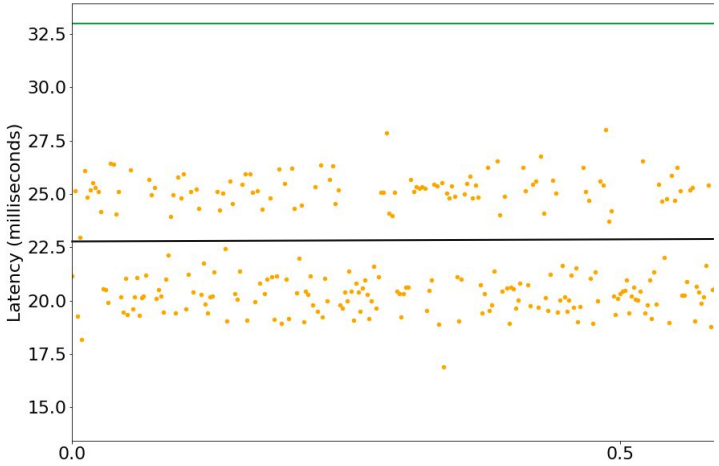
Average Latency: 20.857466453674135

Improper Sequence Update Attack



Average Latency: 21.1036558908046

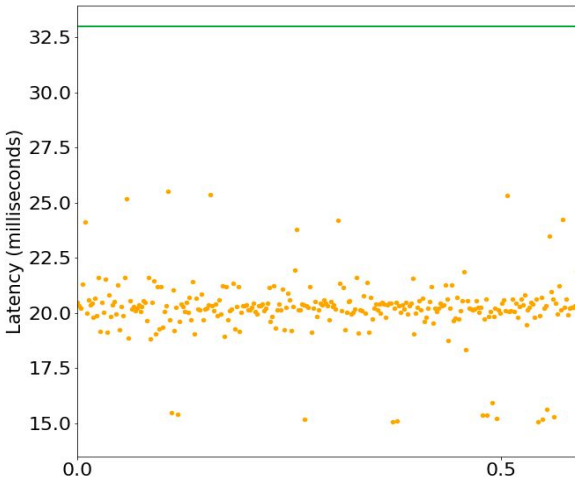
Infinite Pre-order Messages



Average Latency: 22.062072710103894

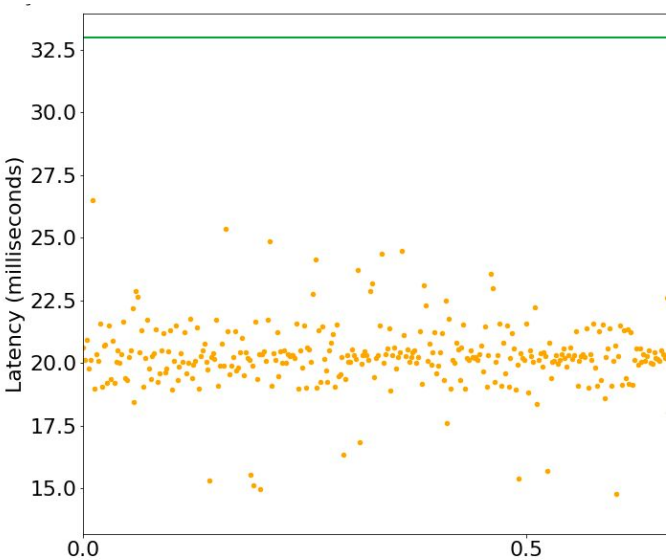
Spam Pre-Order Messages

100 / message



Average Latency: 22.062072710103894

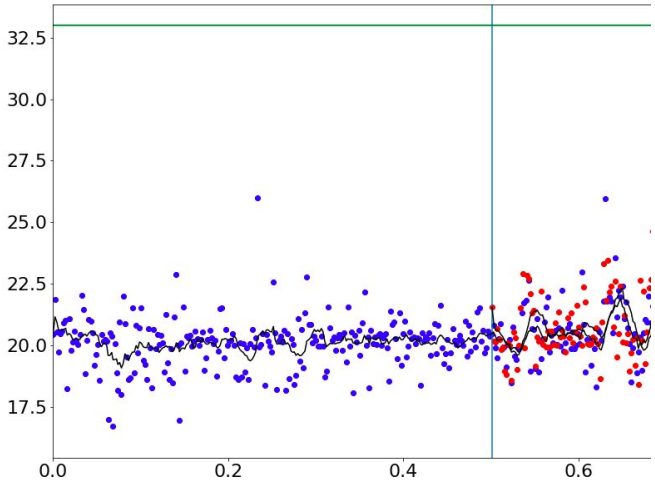
10,000 / message



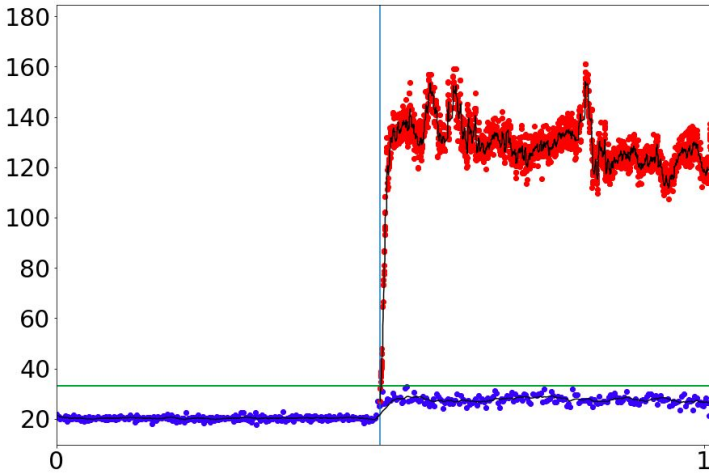
Average Latency: 20.24648035190615

Prime Client Resource Consumption

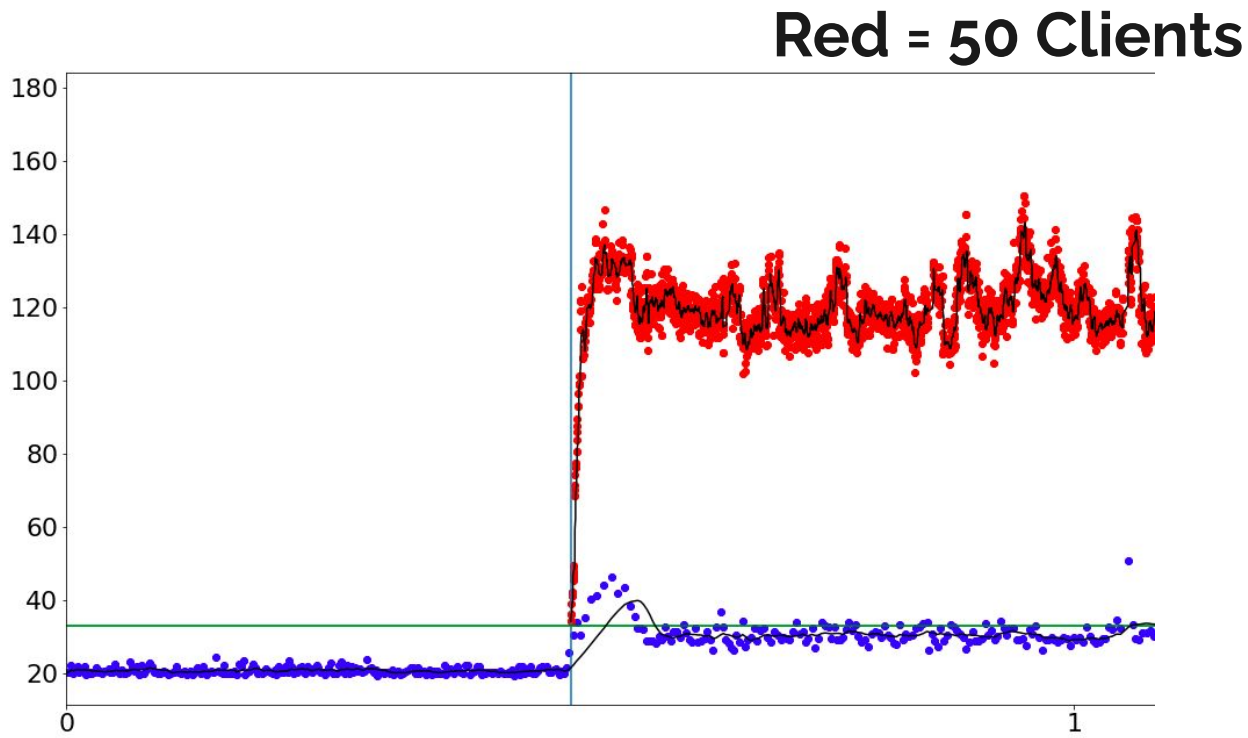
Baseline



Red = 50 Clients



Prime Client Resource Consumption + Failstop



Future Plans

- External: DOS, replay,
- Uses internal information about ports with external attack
- Combined external and internal attacks
- Consensus attack -> this requires > f
    - Attacks validity rather than liveness
- Mitigation Techniques:
    - Admission Control

Questions?

Extra Slides: ...

Spines

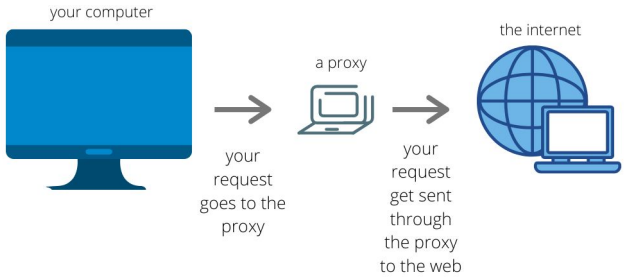
IP Network Problems

- IP Networks are vulnerable to DOS attacks
- IP Networks are **efficient**, but based on **trust**
  - **Thus vulnerable to impersonation**
- IP Networks are **scalable**
- IP Networks do not guarantee **reliability**
- IP Networks are not intrusion-tolerant
  - How do you handle a compromised node?

Proxies

How do we interface with insecure hardware?

- Most electrical control hardware is antiquated
  - Traffic is not encrypted
  - Relies on air gaps for security
- We interface with the control hardware through a general computer known as a proxy. The proxy air gaps the insecure hardware from the open internet and encrypts all traffic.
- A general purpose computer also allows us to standardize communication protocols.



Spire

Overall System Considerations