Decoding the Poseidon Congestion Control Protocol: A Security Assessment

Steven Seiden, Weitao Wang, T. S. Eugene Ng

The Network Congestion Problem

- Today, data centers need to **rapidly serve immense amounts of data**
- To deal with the **high traffic flow**, ***congestion control protocols*** have been implemented within networks to regulate sender traffic
  - These protocols help **prevent packet loss** by **limiting the bandwidth** senders are allotted
  - Traditionally, achieving **network fairness**, thus allocating senders the proper throughput, can be **slow**

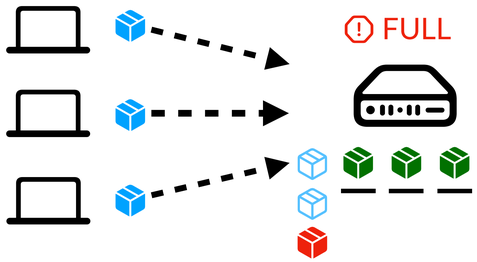


Figure 1: Packet From a Sender Being Dropped by an Overflowing Switch

Introducing Poseidon

- **Poseidon**, a distributed congestion control protocol, aims to **improve network fairness** by:
  - Embedding **network congestion information** into **packets' headers**
  - **Relying on senders** to adjust their sending rate based on received information

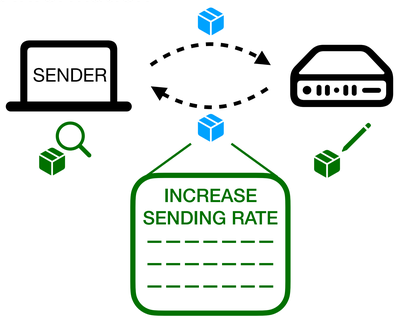


Figure 2: An Overview of the Poseidon Algorithm

- The distribution of congestion control creates new **security threats** explored in this project

Examining Threats

- Our threat model examines two attacks against the availability of the network; switch-based attacks and sender-based attacks

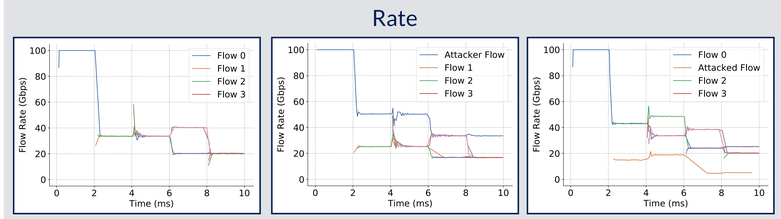


Figure 3: Measurements of all Flows' Sending Rates - Several Experiments

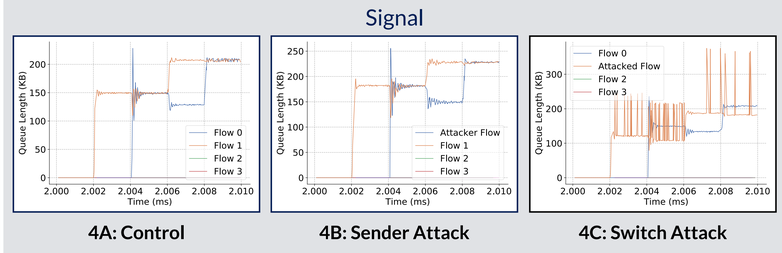


Figure 4: Measurements of Network Signal - Several Experiments

Exploring Sender-Based Attacks

- A sender attempts to take a network offline by **reporting falsified information**
- Experiments revealed **a sender reporting inaccurate bandwidth allocation information impedes fairness**
  - Figures 3B & 4B: *Attacker Flow* reports to have received 50% of its allocated bandwidth, allowing it to gain extra bandwidth, negatively affecting the other senders

Exploring Switch-Based Attacks

- A rogue switch manipulates the bandwidth given to senders by **reporting different queue lengths to different senders**
- Experiments revealed manipulating packet headers to report different queue lengths can **skew bandwidth allocation**
  - Figures 3C & 4C: Packets sent to the *Attacked Flow* state the queue length is half the actual size, reducing the *Attacked Flow's* rate and increasing all other flows' rates

Resolving Issues

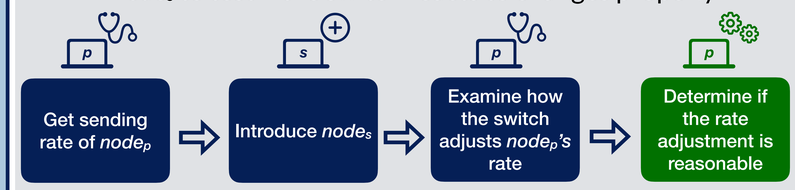
Mitigating Sender-Based Attacks

- Mitigating sender-based attacks is similar to that of traditional networks
- Methods include requiring the switch to scan for rogue senders

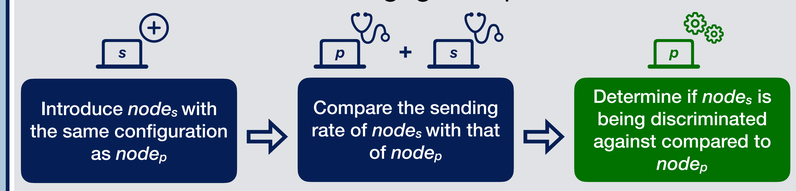
Mitigating Switch-Based Attacks

- Senders need a method to individually detect switch attacks
- Our solution: having a ***physical sender (node$_p$)*** introduce a ***virtual sender (node$_s$)*** to test the switch's reactions in two ways:

Comparing *node$_p$*'s sending rate before and after introducing *node$_s$* to test if the switch reacts to changes properly



Comparing *node$_p$*'s and *node$_s$*'s sending rates when sending simultaneously with similar configurations to determine if the switch is discriminating against specific senders



- Both methods have been **implemented in Python** and **allow any sender to detect** switch attacks under various conditions

What Was Learned

- Distribution of congestion control can **drastically improve network performance**, but can lead to **new security threats**
- Employing sender and switch security protocols can help **ensure network availability**

All work based on Poseidon: Wang, W., Moshref, M., Li, Y., Kumar, G., Ng, T. S. E., Cardwell, N., & Dukkipati, N. (2023). *Poseidon: Efficient, Robust, and Practical Datacenter {CC} via Deployable {INT}*. 255–274. https://www.usenix.org/conference/nsdi23/presentation/wang-weitao