P4Runtime Security and Man-in-the-Middle Attacks

TU Delft

1 Background

P4Runtime: data-plane API for communication between control and data planes [1]

- gRPC for remote procedure calls
- Protocol buffers for message format
- Multiple architectures possible but always one primary controller

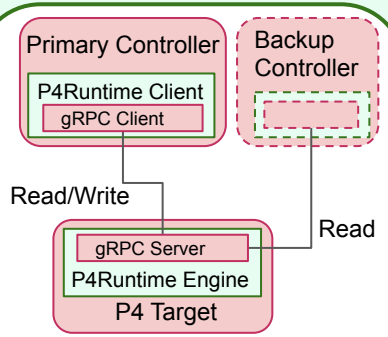

Figure 1: P4Runtime Example Architecture

5 Results

- Scenario 1:

With an insecure channel a controller can send a higher election_id to become the primary controller

This is not possible with a secure channel where the switch authenticates the controller

- Scenario 2:

Could not run controller in-band on a host in Mininet

6 Conclusions

- TLS should be used to secure channels, agreeing with the P4Runtime Specification [1]
- In an insecure channel any controller with a high election_id can become the primary controller

2 Research Question

Can the communication channel between the client and the P4Runtime Engine potentially be corrupted?

4 P4Runtime

- Controller with the highest election_id is the primary controller
- Main attacks are man-in-the-middle and channel flooding [3]
- P4Runtime trusts the messages from gRPC, so the gRPC connection needs to be secured, for example with mutual TLS

7 Future Work

- Use different network simulator or controller
- Effects of a successful man-in-the-middle attack
- Look at real topologies and how controller connects to switches

3 Methodology

- Used mininet and bmv2 to simulate a network, using p4lang/tutorials [2] as starting point
- Original host controller, attacking controller, and switches running on localhost
- Scenario 1: controllers competing to be primary, switch used for man-in-the-middle attacks between hosts
- Scenario 2: man-in-the-middle attack between controller and switch

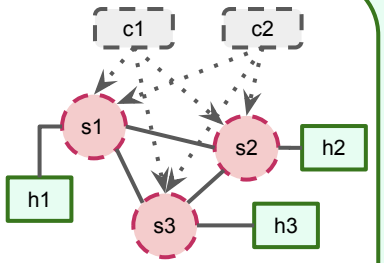

Figure 2: Scenario 1 with competing controllers

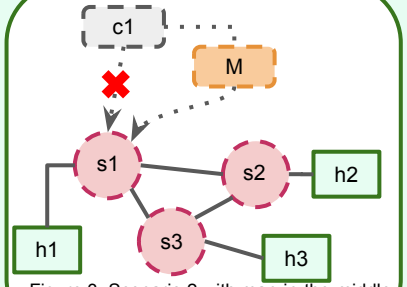

Figure 3: Scenario 2 with man-in-the-middle attack between controller and switch

References:

[1] The P4.org API Working Group, "P4runtime specification." [Online]. Available: https://p4.org/p4-spec/p4runtime/main/P4Runtime-Spec.html

[2] "P4 tutorial," 2022. [Online]. Available: https://github.com/p4lang/tutorials

[3] A. Agape, M. C. Danceanu, R. R. Hansen, and S. Schmid, "Charting the security landscape of programmable dataplanes," CoRR, vol. abs/1807.00128, 2018. [Online]. Available: http://arxiv.org/abs/1807.00128

By: Areti Katsikis,
a.katsikis@student.tudelft.nl
Responsible Professor: Fernando Kuipers
Supervisor: Chenxing Ji