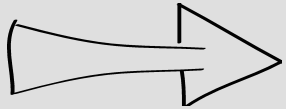


P4/C Stateful Firewall

Dataplane Acceleration Developer Day
2016

Agenda

- Introduction
- What is a Firewall
- Network Address Translation
- Test Setup
- Sequence Diagram
- P4 Firewall Demonstration
- Code Walkthrough

What is a Firewall

- Goal: apply configured policies (allow / drop) to network traffic
  - Server software - embedded in / running on OS
  - Networking hardware - between physical network interfaces
  - SmartNIC software - attached to PCIe VFs / physical network interfaces
- Different Approaches
  - Default allow
  - Default deny

NAT Firewall

- Stateful/NAT Firewall
  - Everything external is blocked
  - Internal hosts are allowed to make external requests
  - External replies are allowed after a request was made from an internal host

Network Address Translation (NAT)

- Types of NAT
  - Full Cone NAT (Static NAT)
  - Restricted Cone NAT (Dynamic NAT)
  - Port Restricted Cone NAT (Dynamic NAT)
  - Symmetric NAT (Dynamic NAT)

Network Address Translation (NAT)

- Port Restricted Cone NAT
  - Maps a public IP address and Port to a LAN IP and Port
  - Internal client must first have sent packets to IP address (X) before it can receive packets from X
  - Where a restricted cone NAT will accept connections from any source port a port restricted cone NAT restricts this further by only accepting connections from the IP address and port it sent the outbound request to

Tables

- To Controller
  - To add a custom header
- From Controller
  - To remove the custom header
- NAT
  - Do Network Address Translation
- Payload Scan
  - Searching for a keyword or character in the payload

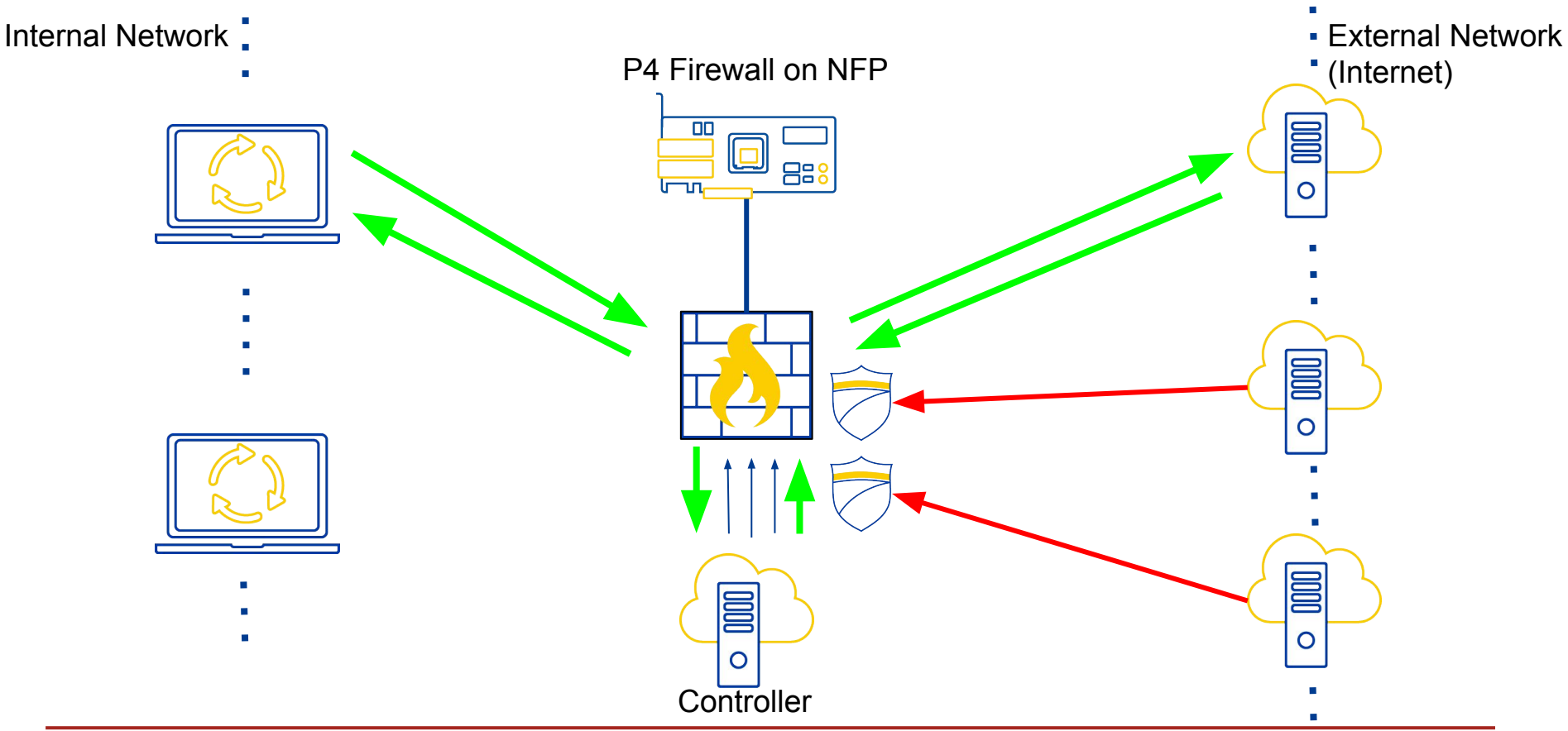
NAT Actions

- External -> Internal -> Miss
  - Default
  - Drop
- Internal -> External -> Miss
  - Send to controller
- Internal -> External -> Hit
  - Rules are added to allow forwarding
- External -> Internal -> Hit
  - Rules are added to allow forwarding

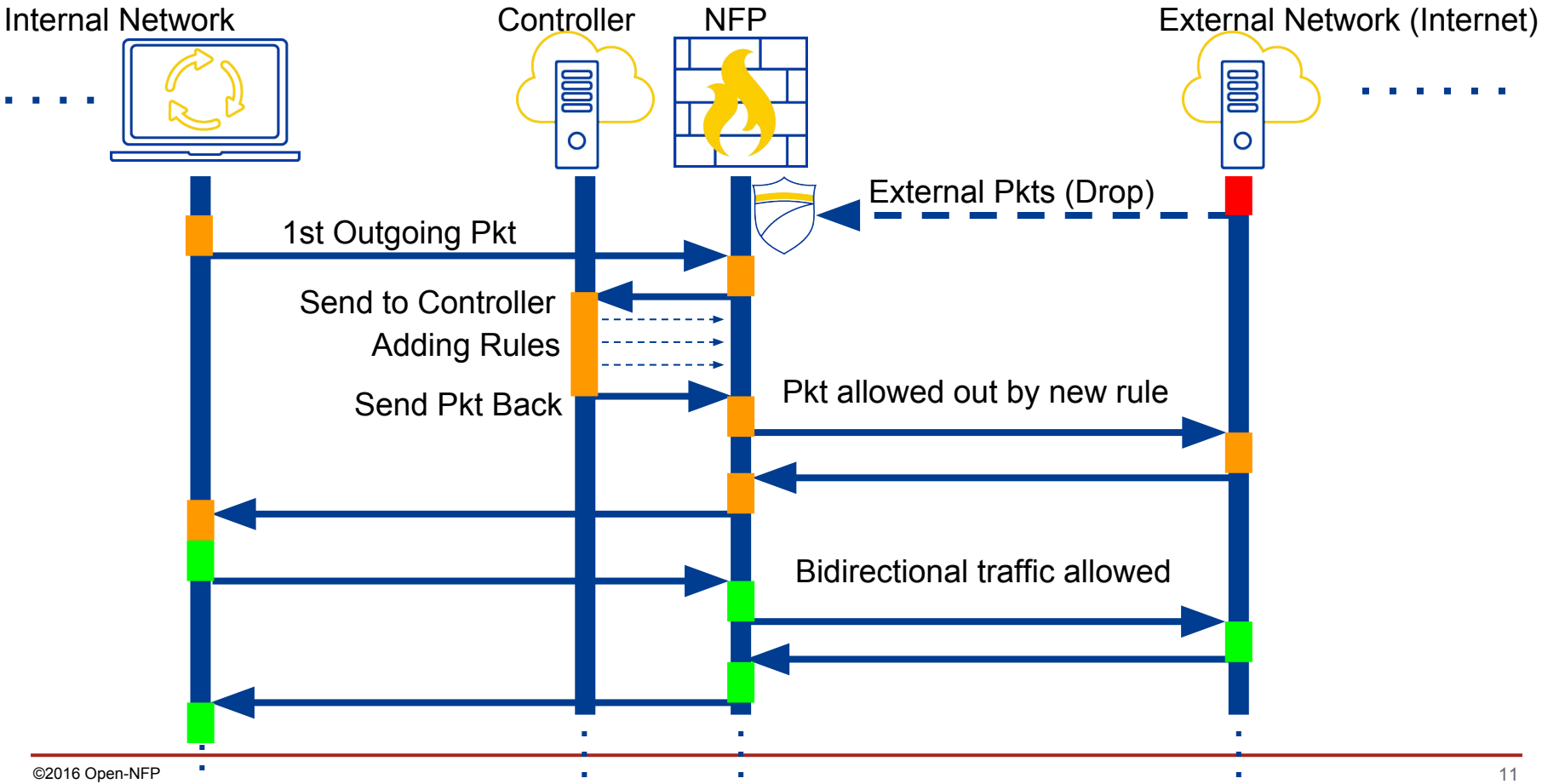
NAT Hit

- Internal -> External -> Hit
  - Source IP -> Router's Public IP
  - Source Port -> Port Selected by Route (starting at 1025)
- External -> Internal -> Hit
  - Destination IP -> Router's Public IP
  - Destination Port -> Port previously selected by router

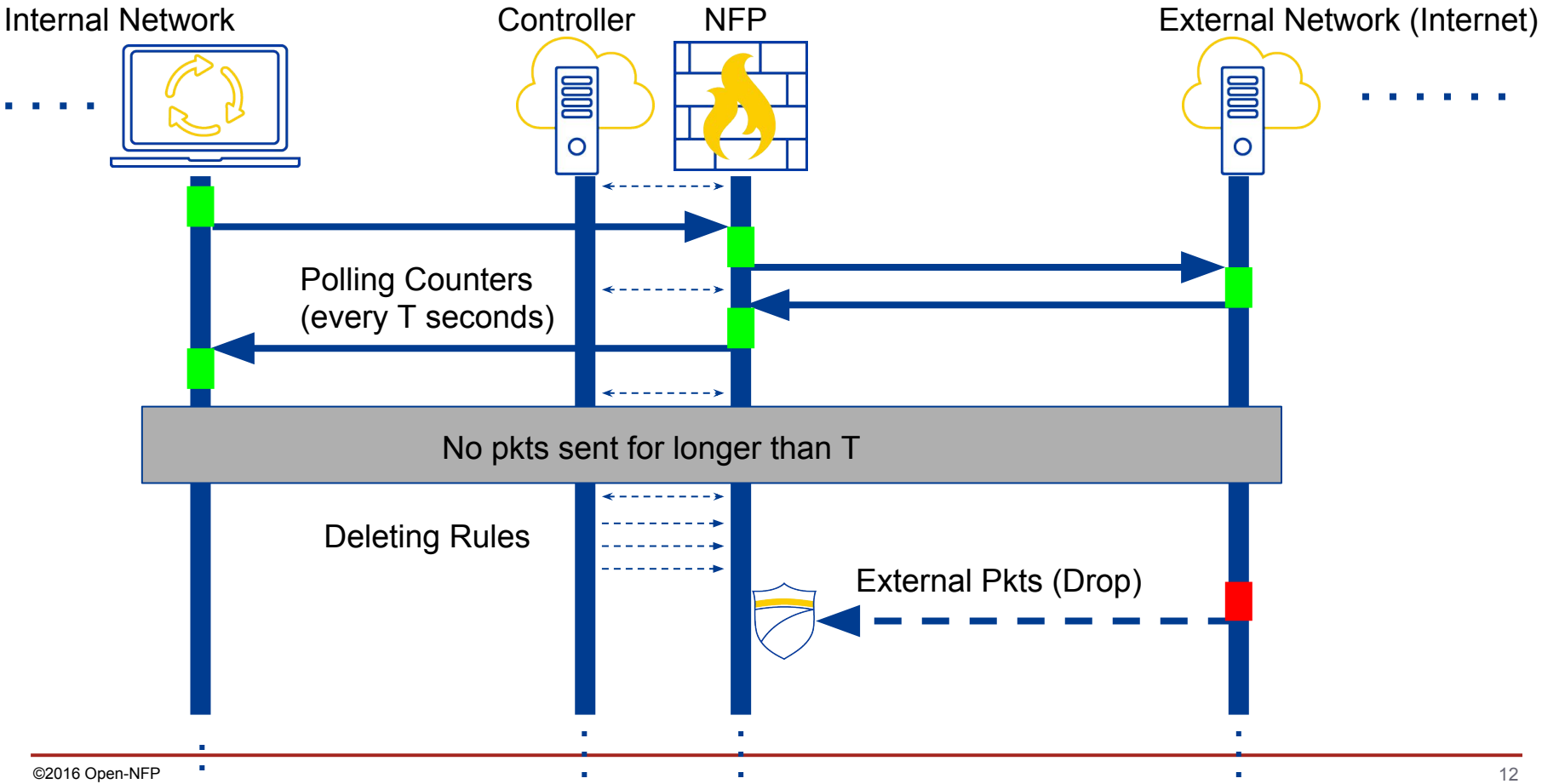
Test Setup



Firewall Sequence Diagram



Timeouts



P4 Firewall Demonstration

- Multiple Match Types per rule
  - Exact
  - Ternary
  - Valid
- 5 Tuple Matching
  - Source/Destination IP
  - Source/Destination Port
  - Valid Protocol

P4 Firewall Demonstration

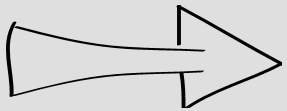
- Rule Priority
  - Hits initial (non-default) rule, unless a rule with higher priority is present
  - Priority represented as a number
- Add/Remove Custom Headers
  - Adding a custom header to identify the packet in a controller
  - Removing the header to send out a valid packet
- Custom C Function Call in P4
  - Primitive Action
  - Search Through Payload

P4 Firewall Demonstration

- Re-Calculating Checksums
- Counters
- Python Controller
- Dynamically Adding Rules
  - Using the RTAPI
- Timeouts - Dynamically Removing Rules
  - Using the RTAPI
- Breakpoints and Stepping

OpenNFP

DXDD 2016



Thank You