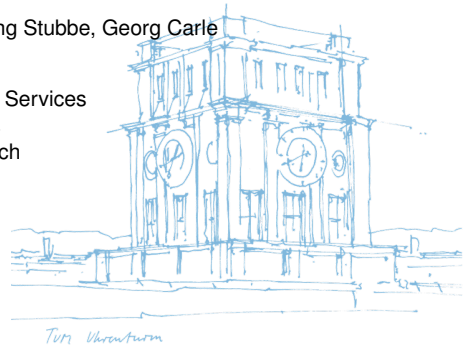
Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

SYN Flood Defense in Programmable Data Planes

Dominik Scholz, Sebastian Gallenmüller, Henning Stubbe, Georg Carle

Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich



Motivation

SYN Floods are (Still) a Serious Problem

Flaw in TCP handshake protocol

- "top-placed SYN flooding, whose share [...] reached a record high of 92.6"[1]
- problem will always exist

→ networks/end nodes always need protection

- volume of attacks increases

→ solutions need to scale

→ **Move mitigation to the data plane**
Anywhere in your network, efficiently, flexible

Available solutions

- network stacks

→ do not scale

- using your favorite packet processing framework

→ not portable

- commercial solutions (e.g. traffic scrubbing centers)

→ closed-source

[1] Kaspersky: "DDoS attacks in Q1 2020", [Online] `https://securelist.com/ddos-attacks-in-q1-2020/96837/`

Outline

SYN Flood-specific Mitigation

SYN Flood Mitigation with P4

Teaser & Conclusion

SYN Flood-specific Mitigation

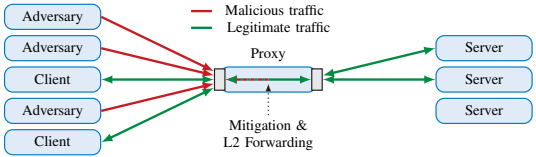
Deployment Scenarios

On the endhost

- does not scale
- takes away resources from server

SYN proxy

- in network or in cloud
- protect multiple servers or whole network(s)
- intercepts TCP flows



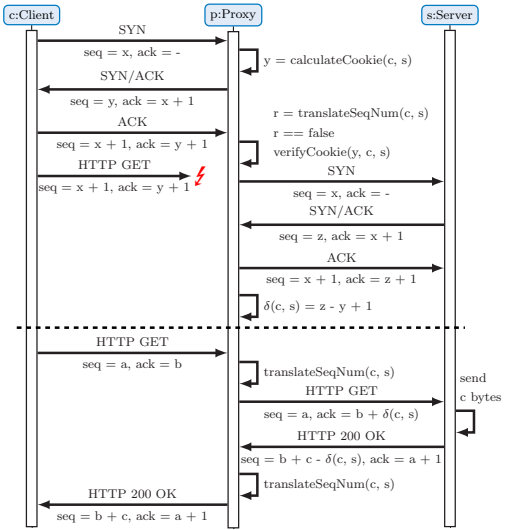
Several challenges when implementing mitigation strategies as SYN proxy in data plane
We focus on SYN cookies and SYN authentication

SYN Flood-specific Mitigation

Protecting Against SYN Floods

SYN cookies

- cryptographic hash (cookie) bound to flow
- no state maintained by proxy until handshake finished
- two connections
  - client – proxy
  - proxy – server
- proxy needs to translate between both connections
- initial data segments might be lost

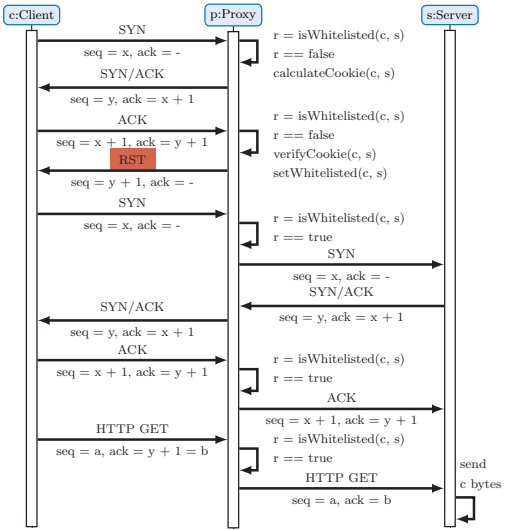


SYN Flood-specific Mitigation

Protecting Against Syn Floods

SYN authentication

- interrupt initial connection attempt
- whitelist client/subnet once challenge completed
  - reset
  - full handshake
  - higher-layer connection
- combined with cookie
- one connection, no translation



SYN Flood-specific Mitigation

Comparison

	SYN Cookies	SYN Authentication
Packet modification	every segment	handshake
Transparent	yes	no
Option support	limited (encoded)	full
State		
State per	flow	flow/subnet
Lookup for	not SYN	every segment

Program core

- parse up to and including TCP header
- essentially L2 forwarder
- received packet is modified according to strategy used
- state (e.g. whitelist) maintained in match-action table

https://bit.ly/36IDtQP

Target-specific changes

- architecture model
- cryptographic hash for cookie

SYN Flood Mitigation with P4

Challenges

Cryptographic hash

- possible for several targets [2]
- add extern (DPDK, NPU)
- modify architecture (FPGA)
- offload to another node (ASIC)

→ portability issue

Whitelisting

- maintaining state requires control plane
- e.g. evicting outdated entries
- alternative: bloom-filter using register extern

→ architecture specific (resources, performance)

[2] Scholz, Dominik, et al. "Cryptographic Hashing in P4 Data Planes." 2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS). IEEE, 2019.

Teaser & Conclusion

What else is in the paper?

- performance figures
  - t4p4s/DPDK
  - Agilio SmartNIC
  - NetFPGA SUME
- case study: SYN flood mitigation in Linux
- comparison with software implementation based on libmoon/DPDK
  - time: 6 months vs. 2 weeks
  - LoC: 1.000 vs 100
  - targets: DPDK vs DPDK, NPU, FPGA

Conclusion

- easy to implement
- scales
- portable
- but requires cryptographic hashing
- targets still require domain-specific knowledge