Relying on RARE for DDoS Attack Protection - Demonstrating RARE Integration with GÉANT DDoS Attack Protection Services (FoD and NeMo Use Cases)

Nikos Kostopoulos, David Schmitz

08.12.2023

www.geant.org
DDoS-Attack (1)
• Victim host attacked by DDoS
• Victim's local network may also be impacted?
FoD and NeMo with freeRTR: DDoS detection and mitigation demo

- Firewall-On-Demand (FoD): own DDoS mitigation by the user via BGP FlowSpec
- NeMo: DDoS detection and mitigation
- freeRTR
  - emulation and demo-ing of CISCO-like router(s)
  - also used in real hardware
Firewall-On-Demand (FoD)
Firewall-On-Demand (FoD): Introduction

- not 'Firewall' in the usual sense!
- service for DDoS mitigation control by user himself
  - dynamically, on the routers
  - BGP FlowSpec-based
  - multi-tenant, eduGAIN-based
  - developed by GÉANT project
- z.B. GÉANT FoD service instance
  - mitigation within GÉANT core
  - for NREN NoC Admins
  - productive since > 8 years
- GÉANT WP8-T3-DDoS
  - Continued development and support
  - Collaboration with the GÉANT security team
Firewall-On-Demand (FoD): Benefit for Users

• user (NREN NoC) is able to perform DDoS mitigation
  • for own IP traffic: start/edit/stop
  • manually (WebUI) oder automated (REST API)
  • without contacting GÉANT NoC

• ⇒ flexible, independent, fast mitigation
  (most DDoS attacks: < 1 h)
Firewall-On-Demand (FoD): Input of a mitigation rule

- **Match**
  - source IP prefix (attacker)
  - destination IP prefix (multi-tenant)
  - IP protocol: ICMP, UDP, TCP
  - ggf. UDP/TCP port (lists)
  - IP fragment options
- **Mitigation**
  - drop all
  - rate limit
- **Expire time**
Firewall-On-Demand: Mitigation

- Victim host attacked by DDoS
- Victim's local network may also be impacted
Firewall-On-Demand: Mitigation

- DDoS traffic blocked as early as possible
- Based on BGP FlowSpec supported in routers
Firewall-On-Demand (FoD): Statistics of a mitigation rule

actually dropped bytes / packets via SNMP (JUNOS-specific filter stats via Firewall MIB) from routers, aggregated
Firewall-On-Demand (FoD): Overview of the mitigation rules

### My rules

#### Firewall Rules

<table>
<thead>
<tr>
<th>Name</th>
<th>Match</th>
<th>Then</th>
<th>Status</th>
<th>Applier</th>
<th>Updated</th>
<th>Expires</th>
<th>Response</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>testrule1_IFZALJ</td>
<td>Det Adr: 12.11.10.10/32, Src Adr: 0.0.0.0/0, Protocol: icmp</td>
<td>rate-limit</td>
<td>DEACTIVATED</td>
<td>admin</td>
<td>2021-07-23</td>
<td>2998-01-01</td>
<td>Rule expired</td>
<td>Reactivate</td>
</tr>
<tr>
<td>testrule2_5SOBCQ</td>
<td>Det Adr: 12.11.10.12/32, Src Adr: 0.0.0.0/0, Protocol: udp</td>
<td>rate-limit</td>
<td>DEACTIVATED</td>
<td>admin</td>
<td>2021-07-23</td>
<td>2021-08-21</td>
<td>Rule expired</td>
<td>Reactivate</td>
</tr>
<tr>
<td>testrule2_PHLDL</td>
<td>Det Adr: 12.11.10.12/32, Src Adr: 0.0.0.0/0, Protocol: udps</td>
<td>rate-limit</td>
<td>ACTIVE</td>
<td>admin</td>
<td>2021-07-23</td>
<td>2021-08-21</td>
<td>Successfully committed</td>
<td>Deactivate</td>
</tr>
</tbody>
</table>
"FoD in a box" using Docker Compose

- Docker based-container running FoD inside
  - as reference installation
  - for testing
- Docker Compose specification for FoD container, Freertr router, attacker and victim host containers
- instructions how to build and use Docker Compose specification manually
- automated FoD Mitigation Demo (based on Docker Compose)
"FoD in a box" using Docker Compose: Automated Mitigation Demo

- runs only in terminal, not via Web UI
- rules emitted into FoD via Python code

Demo
"FoD in a box" based on Containerlab

- Containerlab (https://containerlab.dev/)
  - similar as Docker Compose, but more network-centric
  - typically prebuild containers for testing specific network components (e.g., routers, freeRTR, FoD, etc.) are used

- Containerlab specification for FoD with freeRTR:

- Automated FoD Mitigation Demo (based on Containerlab)
  - demo script: https://github.com/rare-freertr/freeRtr-containerlab/blob/main/lab/005-rare-hello-fod/containerlab-fod-freertr.sh (requires containerlab to be installed)
"FoD in a box" based on Containerlab: Automated Mitigation Demo

- runs only in terminal, not via Web UI
- rules emitted into FoD via Python code

Demo
NeMo with freertr: installation and use