SDN-Based DDOS detection & mitigation

group13

b04902040王郁婷、b04902103蔡昀達、b04902108蘇彥齊、b03902092康譽騰

Functions

1. Set up experiement environment with mininet & Ryu SDN Framework
2. Conduct DDos **attack**
3. Implement several SDN-based DDoS **detection** mechanisms
4. Implement several SDN-based DDoS **Defense** Framework
5. Compare the effectiveness between different results

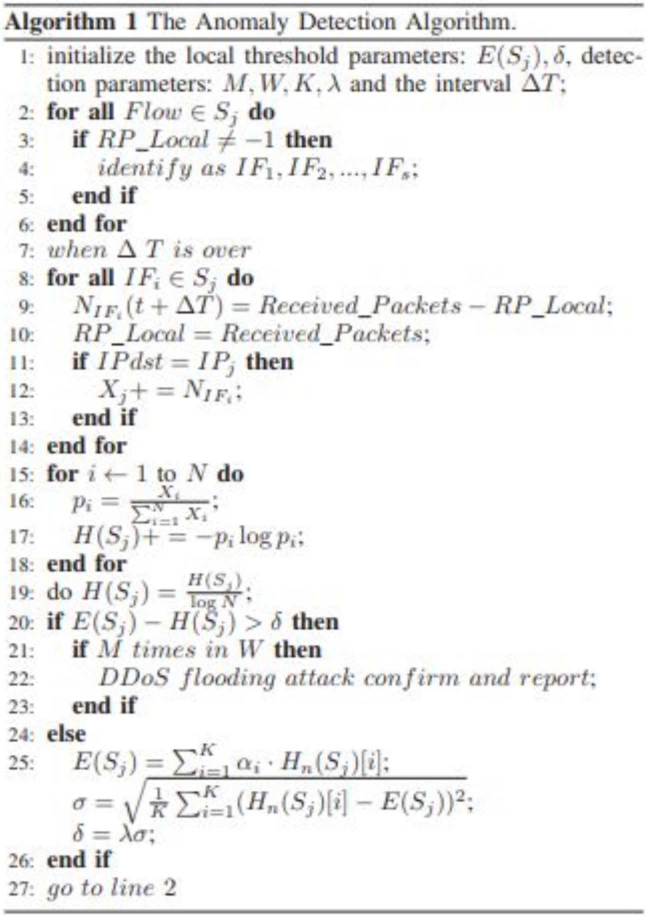
Attack

1. Replay recorded packets of daily usage(ex. watching youtube)
   - 網路流量錄製: Wireshark
   - 重播: TCP replay
2. hping
   - a command-line oriented TCP/IP packet assembler/analyzer
     - Firewall testing
     - Network testing, using different protocols
     - Advanced traceroute, under all the supported protocols

Detection

1. basic
2. entropy-based
3. destination-based
4. connection-based

Entropy-based detection



Algorithm 1 The Anomaly Detection Algorithm.

```
1:  initialize the local threshold parameters: $E(S_j), \delta$, detection parameters: $M, W, K, \lambda$ and the interval $\Delta T$;
2:  for all $Flow \in S_j$ do
3:      if $RP\_Local \neq -1$ then
4:          $identify\ as\ IF_1, IF_2, ..., IF_s$;
5:      end if
6:  end for
7:  $when\ \Delta\ T\ is\ over$
8:  for all $IF_i \in S_j$ do
9:      $N_{IF_i}(t + \Delta T) = Received\_Packets - RP\_Local$;
10:     $RP\_Local = Received\_Packets$;
11:     if $IPdst = IP_j$ then
12:         $X_j+ = N_{IF_i}$;
13:     end if
14: end for
15: for $i \leftarrow 1$ to $N$ do
16:     $p_i = \frac{X_i}{\sum_{i=1}^{N} X_i}$;
17:     $H(S_j)+ = -p_i \log p_i$;
18: end for
19: do $H(S_j) = \frac{H(S_j)}{\log N}$;
20: if $E(S_j) - H(S_j) > \delta$ then
21:     if $M\ times\ in\ W$ then
22:         $DDoS\ flooding\ attack\ confirm\ and\ report$;
23:     end if
24: else
25:     $E(S_j) = \sum_{i=1}^{K} \alpha_i \cdot H_n(S_j)[i]$;
        $\sigma = \sqrt{\frac{1}{K} \sum_{i=1}^{K} (H_n(S_j)[i] - E(S_j))^2}$;
        $\delta = \lambda\sigma$;
26: end if
27: $go\ to\ line\ 2$
```

Mitigation

1. basic
   - block specific IP when DDos detected
2. limit connection

Test1 Normal Condition

Use tcpreplay to simulate normal user’s traffic in the background.

From the two h1 terminals open some TCP connections towards h2 .

```
h2$ python echo_server.py 2000
```

```
h1$ nc -T af11 10.0.0.2 2000
```

```
h1$ tcpreplay -i s2-eth9 pkt.pcap
```

Test2 Attack Condition

```
h2$ hping3 10.0.0.1 --flood -S -a 10.0.0.3
```

--flood = Sending packets as fast as possible, without taking care to show incoming replies. Flood mode.

-S = syn packet

Test3 Attack Condition

```
h2$ hping3 10.0.0.2 --flood -S -V --rand-source
```

--flood = Sending packets as fast as possible, without taking care to show incoming replies. Flood mode.

-S = syn packet

--rand-source

Test4 Mitigation

1. Block the attacker IP immediately.

2. limit the new connection rate with OFPMeterBandDscpRemark.

DEMO

Reference

- An Entropy-Based Distributed DDoS Detection Mechanism in Software-Defined Networking Rui Wang, Zhiping Jia∗, Lei Ju 2015 IEEE Trustcom/BigDataSE/ISPA
- Bawany, Narmeen Zakaria, Jawwad A. Shamsi, and Khaled Salah. "DDoS attack detection and mitigation using SDN: methods, practices, and solutions." Arabian Journal for Science and Engineering 42.2 (2017): 425-441.