Software Defined Networks

ARP Spoofing Defence Mechanism for a SDN Controller

Outline

01 **ARP Spoofing**

02 **Demonstration of Defense Mechanism**

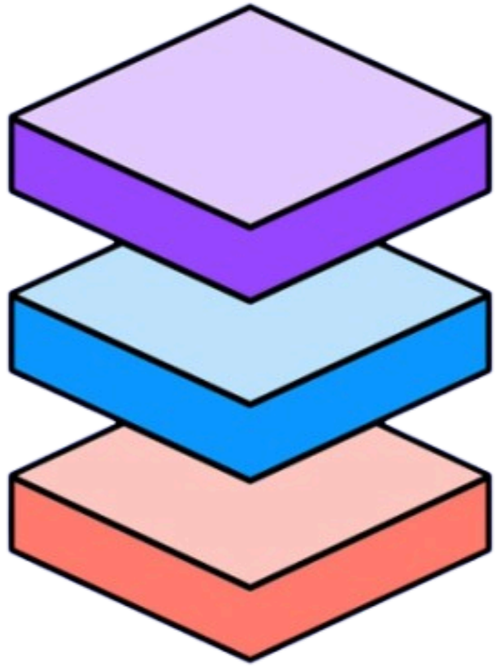
03 **References**

What's ARP Spoofing ?

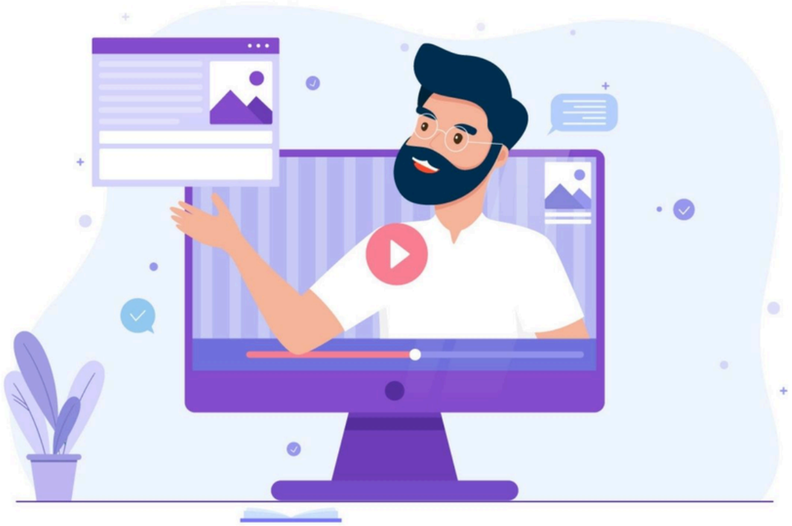
- **ARP spoofing is a technique used by attackers to perform cache poisoning by inserting false IP to MAC address mappings in victim's ARP cache.**
- **ARP spoofing attack comes in different forms namely request and response attacks.**
- **In this attack, an attacker spoofs ARP reply message claiming that this destination IP is mine and sends destination IP & his MAC in the ARP header.**
- **When victim receives this spoofed ARP message, it updates its ARP cache table with the attacker's forged IP-MAC pair.**

Technologies and Tools Utilized

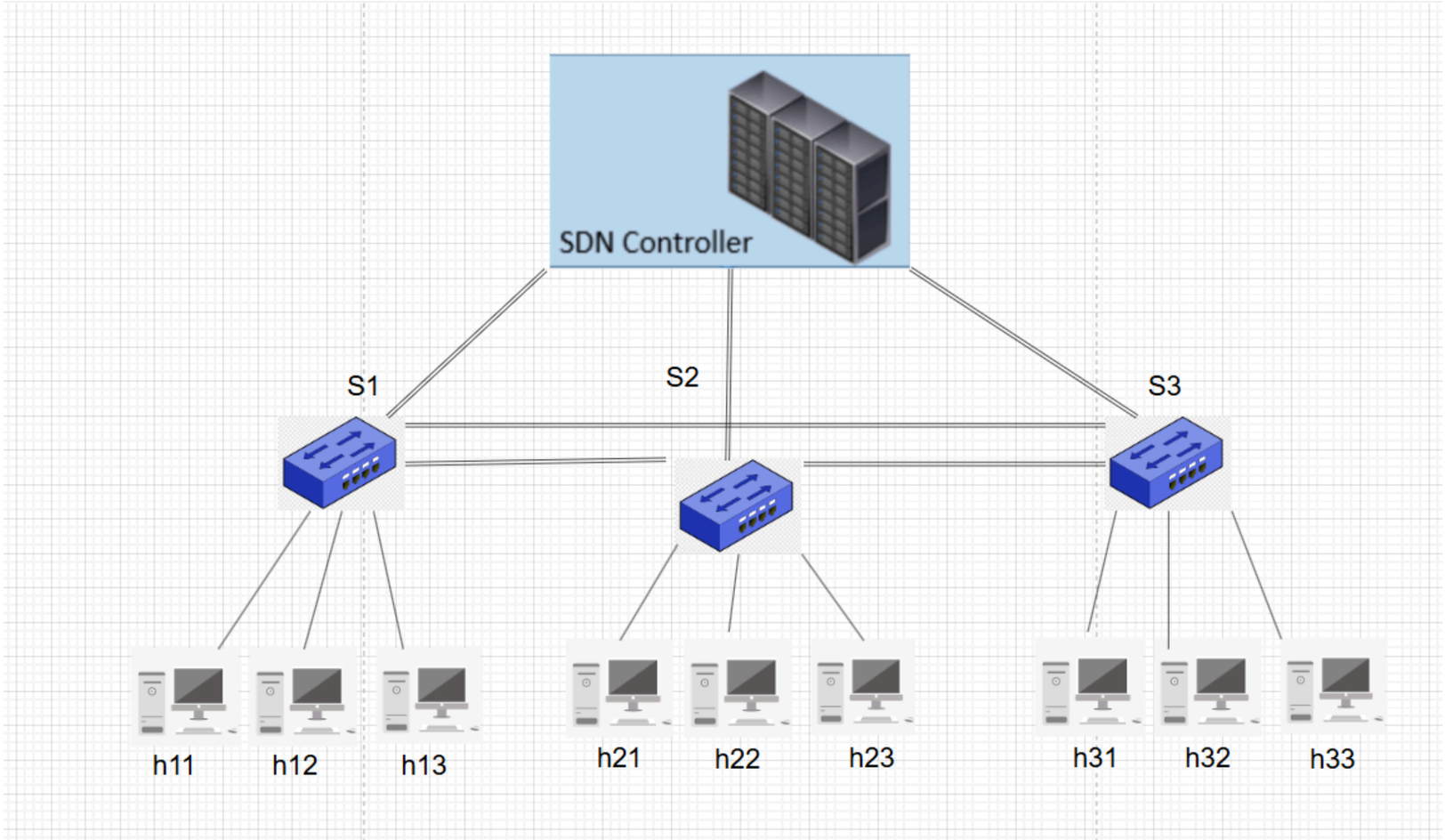
- **Mininet**
- **POX Controller**
- **Open flow protocol**
- **DHCP Server (Integrated in the Controller)**



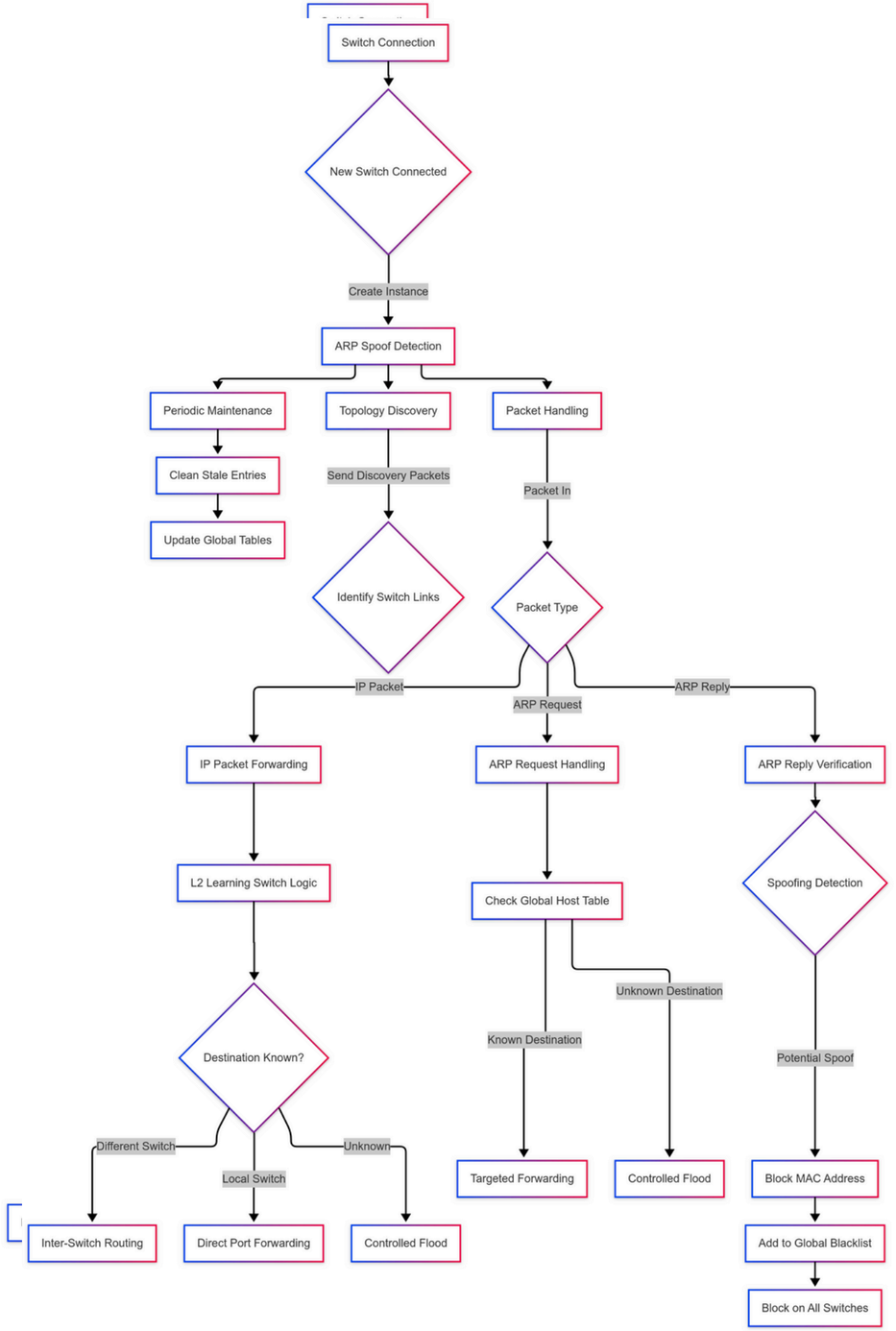
Demonstration of ARP Spoofing Defense mechanism in a SDN Controller



Topology Setup



Work flow of POX Controller



Demonstration of ARP Spoofing Mitigation in SDN

- **Initialization of Network Topology**

```
mininet@mininet-vm:~$ sudo python Topology.py
*** Creating network
*** Adding controller
*** Adding hosts:
h11 h12 h13 h21 h22 h23 h31 h32 h33
*** Adding switches:
s1 s2 s3
*** Adding links:
(h11, s1) (h12, s1) (h13, s1) (h21, s2) (h22, s2) (
*** Configuring hosts
h11 (cfs -1/100000us) h12 (cfs -1/100000us) h13 (cf
1/100000us) h31 (cfs -1/100000us) h32 (cfs -1/10000
*** Starting controller
c0
*** Starting 3 switches
s1 s2 s3 ...
Network is up. Launching CLI...
*** Starting CLI:
mininet>
```

- **Initialization of POX Controller**

```
mininet@mininet-vm:~/pox$ ./pox.py log.level --DEBUG proto.dhcpd --network=10.0.0.0/24 --ip=10.0.0.254 forwarding.l
rning_arp_mitigation
POX 0.7.0 (gar) / Copyright 2011-2020 James McCauley, et al.
DEBUG:proto.dhcpd:Removing my own IP (10.0.0.254) from address pool
DEBUG:proto.dhcpd:DHCP serving addresses from 10.0.0.1 to 10.0.0.254
INFO:forwarding.l2_learning_arp_mitigation:DHCPD detected. Registering DHCP Lease listener.
INFO:forwarding.l2_learning_arp_mitigation:Multi-Switch Controller Initialized
INFO:forwarding.l2_learning_arp_mitigation:Multi-Switch L2 Learning with ARP Spoof Detection Launched.
DEBUG:core:POX 0.7.0 (gar) going up...
DEBUG:core:Running on CPython (3.8.5/Jul 28 2020 12:59:40)
DEBUG:core:Platform is Linux-5.4.0-42-generic-x86_64-with-glibc2.29
WARNING:version:Support for Python 3 is experimental.
INFO:core:POX 0.7.0 (gar) is up.
DEBUG:openflow.of_01:Listening on 0.0.0.0:6633
INFO:openflow.of_01:[00-00-00-00-00-01 2] connected
INFO:forwarding.l2_learning_arp_mitigation:Switch 00-00-00-00-00-01 connected
INFO:forwarding.l2_learning_arp_mitigation:ARP Spoof Detection initialized on switch 00-00-00-00-00-01
INFO:openflow.of_01:[00-00-00-00-00-03 3] connected
INFO:forwarding.l2_learning_arp_mitigation:Switch 00-00-00-00-00-03 connected
INFO:forwarding.l2_learning_arp_mitigation:ARP Spoof Detection initialized on switch 00-00-00-00-00-03
INFO:openflow.of_01:[00-00-00-00-00-02 4] connected
INFO:forwarding.l2_learning_arp_mitigation:Switch 00-00-00-00-00-02 connected
INFO:forwarding.l2_learning_arp_mitigation:ARP Spoof Detection initialized on switch 00-00-00-00-00-02
DEBUG:forwarding.l2_learning_arp_mitigation:S1: Sent discovery packet on port 1
DEBUG:forwarding.l2_learning_arp_mitigation:S1: Sent discovery packet on port 2
DEBUG:forwarding.l2_learning_arp_mitigation:S1: Sent discovery packet on port 3
```

Contd..

- **Connectivity check-up**

```
*** Starting CLI:
mininet> h11 ping h12
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=100 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.340 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.032 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.032 ms
```

```
mininet> h12 ping h31
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data.
64 bytes from 10.0.0.7: icmp_seq=1 ttl=64 time=134 ms
64 bytes from 10.0.0.7: icmp_seq=2 ttl=64 time=0.551 ms
64 bytes from 10.0.0.7: icmp_seq=3 ttl=64 time=0.088 ms
64 bytes from 10.0.0.7: icmp_seq=4 ttl=64 time=0.055 ms
64 bytes from 10.0.0.7: icmp_seq=5 ttl=64 time=0.047 ms
```

- **Testing ARP Spoofing defence (h11 is the attacker)**

```
mininet> h12 ping h31 &
mininet> h11 arp -s 10.0.0.7 00:00:00:00:00:01
mininet> h11 python -c "from scapy.all import *; send(ARP(op=2, pdst='10.0.0.2', psrc='10.0.0.7', hwsrc='00:00:00:00:00:
01', hwdst='00:00:00:00:00:02'))"
.
Sent 1 packets.
```

Contd..

- **Results**

```
DEBUG:forwarding.l2_learning_arp_mitigation:S3: Flooding packet to 27 non-source ports
DEBUG:forwarding.l2_learning_arp_mitigation:S1: 00:00:00:00:00:01 -> 00:00:00:00:00:02 on port 3
DEBUG:forwarding.l2_learning_arp_mitigation:Updated global MAC table: 00:00:00:00:00:01 -> S1:3
DEBUG:forwarding.l2_learning_arp_mitigation:S1: ARP Reply: 00:00:00:00:00:01 claims 10.0.0.7
WARNING:forwarding.l2_learning_arp_mitigation:ARP Spoofing Detected: 00:00:00:00:00:01 is claiming 10.0.0.7 (expected 00
:00:00:00:00:07)
WARNING:forwarding.l2_learning_arp_mitigation:Spoofing Detected: MAC 00:00:00:00:00:01 is malicious on switch 00-00-00-0
0-00-01
INFO:forwarding.l2_learning_arp_mitigation:Blocked MAC 00:00:00:00:00:01 on all switches for 5 minutes
```

```
mininet> h11 ping h12
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
From 10.0.0.1 icmp_seq=9 Destination Host Unreachable
From 10.0.0.1 icmp_seq=10 Destination Host Unreachable
From 10.0.0.1 icmp_seq=11 Destination Host Unreachable
From 10.0.0.1 icmp_seq=12 Destination Host Unreachable
From 10.0.0.1 icmp_seq=13 Destination Host Unreachable
From 10.0.0.1 icmp_seq=14 Destination Host Unreachable
```

```
DEBUG:forwarding.l2_learning_arp_mitigation:Dropping packet from blacklisted MAC: 00:00:00:00:00:01
DEBUG:forwarding.l2_learning_arp_mitigation:S1: 00:00:00:00:00:01 -> ff:ff:ff:ff:ff:ff on port 3
DEBUG:forwarding.l2_learning_arp_mitigation:Dropping packet from blacklisted MAC: 00:00:00:00:00:01
DEBUG:forwarding.l2_learning_arp_mitigation:S1: 00:00:00:00:00:01 -> ff:ff:ff:ff:ff:ff on port 3
DEBUG:forwarding.l2_learning_arp_mitigation:Dropping packet from blacklisted MAC: 00:00:00:00:00:01
```

Thank You