PHY and Link Layers: Attacks and Countermeasures

CYBR371: System and Network Security, (2024/T1)

Arman Khouzani, Mohammad Nekooei

Slides modified from "Masood Mansoori"

18 March, 2024

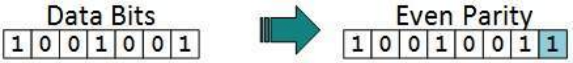
Victoria University of Wellington – School of Engineering and Computer Science

Data Link Layer: Recap

Network Interface (Data Link) Layer

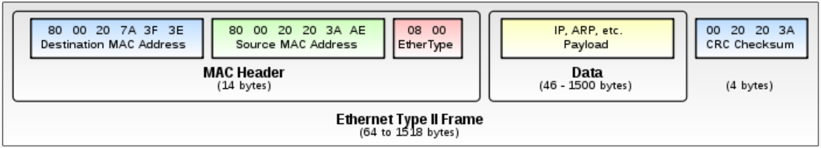
Data Link (Network Interface) Layer functions:

- Framing Physical Addressing
- Error Control (single bit, multiple bits, and burst error).
  - How does it detect errors?



- Flow Control
- Multiple Access

MAC Header (Ethernet Frame)



80 00 20 7A 3F 3E
Destination MAC Address

80 00 20 20 3A AE
Source MAC Address

08 00
EtherType

MAC Header
(14 bytes)

IP, ARP, etc.
Payload

Data
(46 - 1500 bytes)

00 20 20 3A
CRC Checksum

(4 bytes)

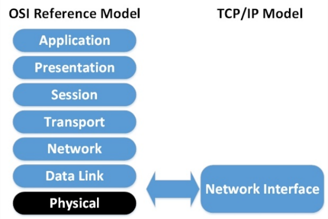
Ethernet Type II Frame
(64 to 1518 bytes)

Ethernet Broadcast Address: `FF:FF:FF:FF:FF:FF`

Ether Type: indicates the type of payload, e.g.,

`0x0800`	Internet Protocol version 4 (IPv4)
`0x0806`	Address Resolution Protocol (ARP)

Network Interface (Data Link) Layer Attacks

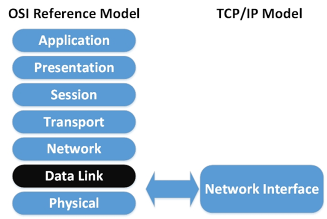


Physical Layer Attacks:

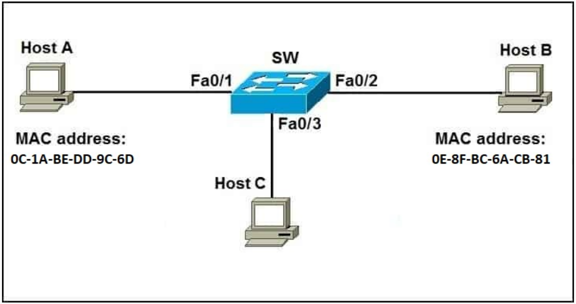
- Power Surge
- EMP
- Jamming
- Cutting wires

What can we do to protect systems against such attacks?

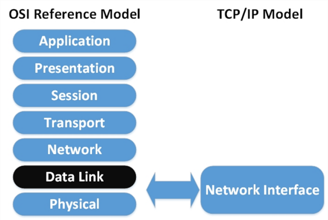
Network Interface Layer (Physical) Attacks



How does Data Link layer packet transmission work?



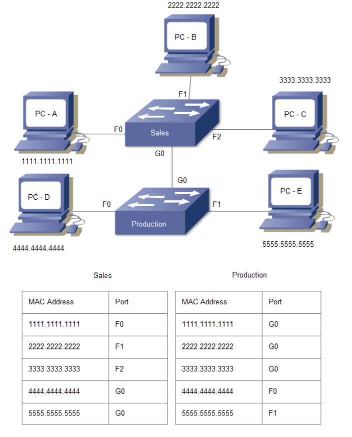
MAC layer attacks



MAC Layer attacks

- CAM table exhaustion
- MAC address spoofing
- Denial of service

CAM table (MAC address table, switch forwarding table)



Sales

MAC Address	Port
1111.1111.1111	F0
2222.2222.2222	F1
3333.3333.3333	F2
4444.4444.4444	G0
5555.5555.5555	G0

Production

MAC Address	Port
1111.1111.1111	G0
2222.2222.2222	G0
3333.3333.3333	G0
4444.4444.4444	F0
5555.5555.5555	F1

A MAC address table, sometimes called a Content Addressable Memory (CAM) table, is used on **Ethernet switches** to determine where to forward traffic on a LAN.

CAM Table Exhaustion/MAC Flooding

Essentially turns a switch into a hub:

- Floods the CAM table with new MAC-port mappings.
- Once table fills up, it broadcasts all messages (fail open).

A simple tool is “**`macof`**” (monkey.org/~dugsong/dsniff/)

Vlan	Mac Address	Type	Ports
1	0223.E754.641E	DYNAMIC	Fa0/1
1	01E0.4F19.2183	DYNAMIC	Fa0/2

Vlan	Mac Address	Type	Ports
1	0223.E754.641E	DYNAMIC	Fa0/1
1	01E0.4F19.2183	DYNAMIC	Fa0/2
1	0F29.E834.4215	DYNAMIC	Fa0/1
1	0405.F531.541E	DYNAMIC	Fa0/1
1	0884.A754.319C	DYNAMIC	Fa0/1
1	0067.C754.640F	DYNAMIC	Fa0/1
⍰	⍰	⍰	⍰...⍰

ARP (Address Resolution Protocol)

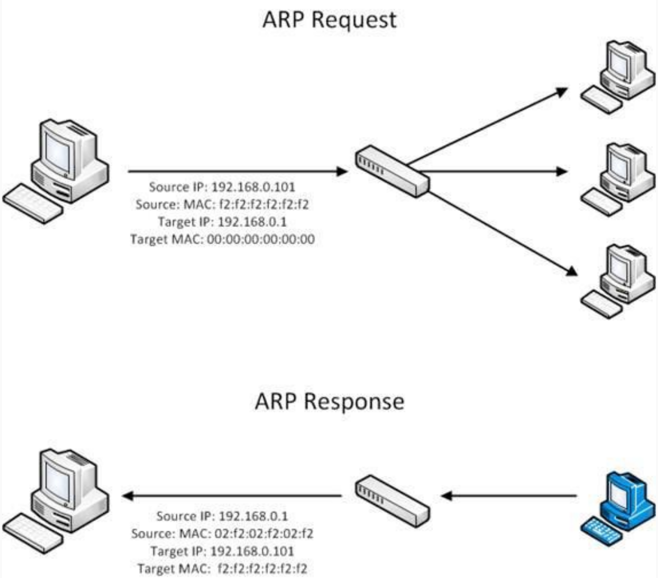
Primarily used to translate IP addresses to Ethernet MAC addresses on a local area network.

If IP address is not found in the **ARP table**:

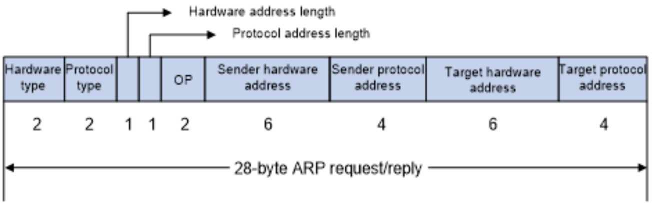
- A host sends a broadcast **ARP request**:
    - “**`Who has 10.0.3.4? Tell 10.0.3.2`**”
- System with that IP address sends a unicast **ARP reply**:
    - “**`I am 10.0.3.4`**”
- This includes the MAC address which can receive packets for that IP.

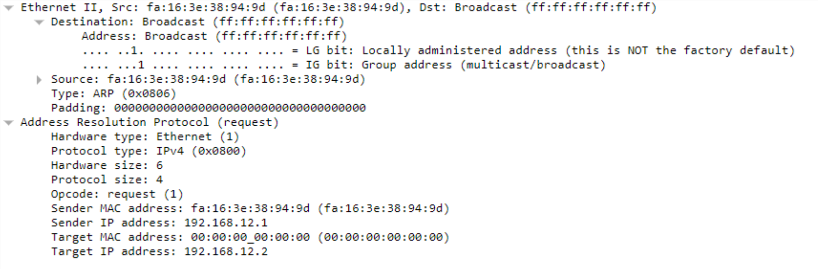
Message types: a) ARP request b) ARP reply c) ARP Gratuitous Message.

ARP Request and Response Process


ARP Request
Source IP: 192.168.0.101
Source: MAC: f2:f2:f2:f2:f2:f2
Target IP: 192.168.0.1
Target MAC: 00:00:00:00:00:00
ARP Response
Source IP: 192.168.0.1
Source: MAC: 02:f2:02:f2:02:f2
Target IP: 192.168.0.101
Target MAC: f2:f2:f2:f2:f2:f2

ARP Request and Response Process





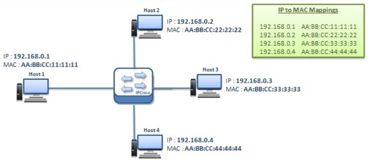
```
▼ Ethernet II, Src: fa:16:3e:38:94:9d (fa:16:3e:38:94:9d), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    ▼ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
          Address: Broadcast (ff:ff:ff:ff:ff:ff)
          .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
          .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
    ▶ Source: fa:16:3e:38:94:9d (fa:16:3e:38:94:9d)
      Type: ARP (0x0806)
      Padding: 000000000000000000000000000000000000
▼ Address Resolution Protocol (request)
      Hardware type: Ethernet (1)
      Protocol type: IPv4 (0x0800)
      Hardware size: 6
      Protocol size: 4
      Opcode: request (1)
      Sender MAC address: fa:16:3e:38:94:9d (fa:16:3e:38:94:9d)
      Sender IP address: 192.168.12.1
      Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
      Target IP address: 192.168.12.2
```

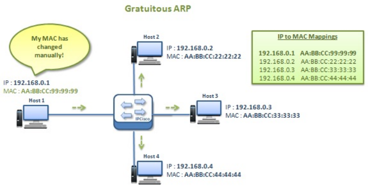
ARP Gratuitous Message

ARP Response that was not prompted by an ARP Request.

- The Gratuitous ARP is sent by a node as a broadcast (**`FF:FF:FF:FF:FF:FF`** MAC address) to announce its IP to MAC mapping to the other hosts on the network.

1. When a host newly joins a network



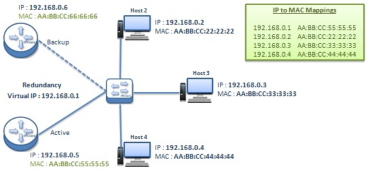


ARP Gratuitous Message

ARP Response that was not prompted by an ARP Request.

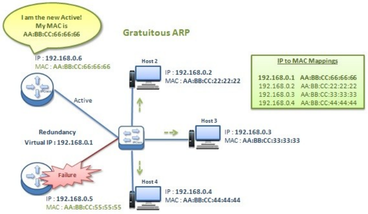
- The Gratuitous ARP is sent by a node as a broadcast (**FF:FF:FF:FF:FF:FF** MAC address) to announce its IP to MAC mapping to the other hosts on the network.

1. When a host newly joins a network
2. May be used in virtual environments, where a specific Virtual Machine ‘jumps’ to a new physical system



IP to MAC Mappings

192.168.0.1	AA:BB:CC:55:55:55
192.168.0.2	AA:BB:CC:22:22:22
192.168.0.3	AA:BB:CC:33:33:33
192.168.0.4	AA:BB:CC:44:44:44



IP to MAC Mappings

192.168.0.1	AA:BB:CC:66:66:66
192.168.0.2	AA:BB:CC:22:22:22
192.168.0.3	AA:BB:CC:33:33:33
192.168.0.4	AA:BB:CC:44:44:44

ARP Cache

Since sending an ARP request/reply for each IP datagram is inefficient, hosts maintain a cache (ARP Cache) of current entries (these entries are set to automatically expire after a period of time (typically 10 to 20 mins).

- To view the ARP cache, run command: **`arp -n`**

```
Address     HWtype  HWaddress          Flags  Iface
10.0.2.2    ether   52:54:0f:12:35:00         p0s3
10.0.2.10   ether   08:00:27:c3:2c:05         enp0s3
10.0.2.1    ether   10:14:05:43:fe:93         enp0s3
10.0.2.9    ether   24:e5:23:11:24:01         enp0s
```

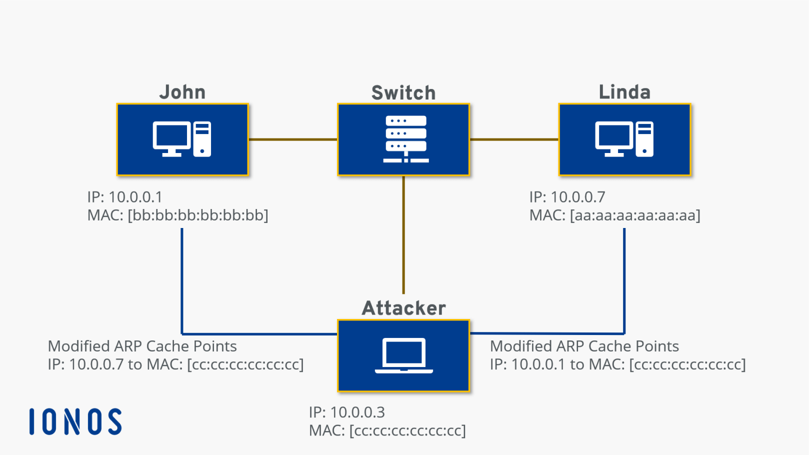
- Clear ARP cache :
  - Run command: **`sudo ip -s -s neigh flush all`**
    Run command: **`arp -n`** (should now show less rows)

ARP Spoofing

ARP spoofing, also called **ARP cache poisoning**:

- Involves causing a target to associate an IP address with an incorrect MAC address.
- Inject forged information into ARP cache;
- MAC Spoofing at:
  - host,
  - Switch,
  - Router
- used for:
  - MiTM
  - DoS

ARP Spoofing



John

Switch

Linda

IP: 10.0.0.1
MAC: [bb:bb:bb:bb:bb:bb]

IP: 10.0.0.7
MAC: [aa:aa:aa:aa:aa:aa]

Attacker

Modified ARP Cache Points
IP: 10.0.0.7 to MAC: [cc:cc:cc:cc:cc:cc]

Modified ARP Cache Points
IP: 10.0.0.1 to MAC: [cc:cc:cc:cc:cc:cc]

IP: 10.0.0.3
MAC: [cc:cc:cc:cc:cc:cc]

IONOS

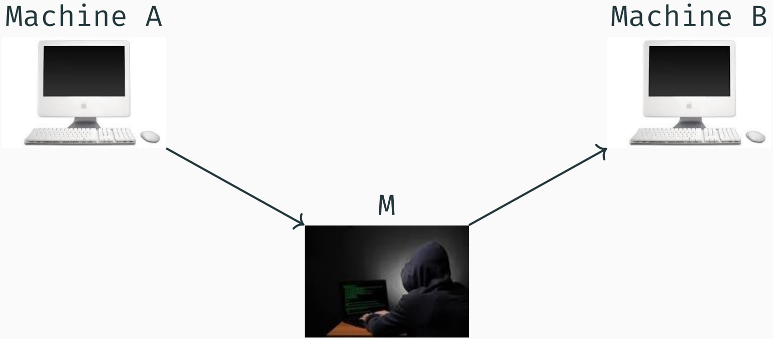
ARP spoofing: constructing an Ethernet frame

```
from scapy.all import *
E = Ether()
A = ARP()

# add the required attributes:
#   IP address of sender (victim Machine),
#   MAC address of sender (forged),
#   Op Type = 1/2,
#   IP address of receiver

frame = E/A
sendp(frame)
```

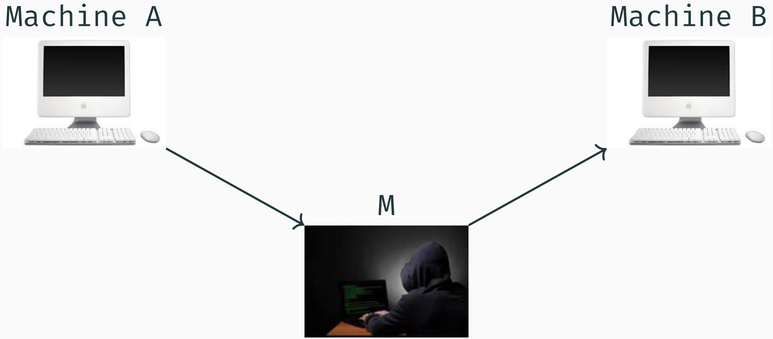
Man-In-the-Middle (MITM) attack through ARP Spoofing



M has to be on the same network as A and B.

NIC only looks at the MAC header, does not look at the IP header, which is a payload of the Ethernet header

Man-In-the-Middle (MITM) attack through ARP Spoofing



Machine A

Machine B

M

Easy if M is in the middle, e.g. a router sitting between A and B.

What if you are not in the middle? Redirect traffic...

What happens when packet goes to the IP layer? 2 scenarios:

- M is a router
- M is a host

Countermeasures

Countermeasures

Question: Why does such an attack succeed in the first place?

Defence:

- Hold down timers
- Static ARP table
- Dynamic ARP inspection (uses DHCP snooping at gateways)
- Port Security

Detection:

- Arpwatch: observes change in ARP packets (only suitable for networks with static IP addresses)
- XARP: Observes change in ARP packets and also sends ARP packets to validate ARP tables

Next: IP (Layer 3) Security