



**Antonio Cianfrani**

# **LAN Security**



# Security attacks in LAN (1/2)

- The classical attack in a LAN is the **MAC Address Flooding**.
- It exploits the security weakness of MAC forwarding table learning mechanism:
  - If an incoming frame with a new MAC source address is received, the switch add a row in the forwarding table
  - If an incoming frame has a destination MAC address not present in the forwarding table, the switch acts as an hub
  - The forwarding tables have a limited size
- **MAC Address Flooding:**
  - Frames with artificial source MAC address → the forwarding table is saturated → frames with new MAC destination address are forwarded in broadcast



## Security attacks in LAN (2/2)

- **DHCP Spoofing:** a malicious DHCP server is inserted in the LAN, so that fake info (default gateway) are notified to LAN hosts. This is a man-in-the-middle attack
- **DHCP starvation:** attack to the DHCP servers, sending a huge amount of DHCP requests so that to use all the available IP addresses.



# Port Security (1/4)

- **Port Security:** option to be configured on switch interface/s to increase the security level of the network
- The idea of Port Security is to limit the end devices that can be connected to a specific switch interface
- The security policy is based on the source MAC address of incoming packets and on the number of different source MAC addresses allowed on the interface.



## Port Security (2/4)

- If a frame having a MAC source address not allowed is received, the interface switch to ***Violation Mode***:
  - ✓ Shutdown by default (*error disabled state*).
  - ✓ To recover from the *error disabled state* it is necessary to manually shutdown the interface
- It is possible to allow the access to a single MAC address or to a range of MAC addresses.
- The association among the interface and the allowed MAC address/es can be dynamic or static
- First step of the configuration:  
**Switch(config-if)# switchport port-security**



# Port Security (3/4)

- Three different Port Security configuration modes:
  - Static: the allowed MAC address/es are statically configured by the LAN administrator with the command  
**Switch(config-if)# switchport port-security mac-address mac-address**
- To look at the MAC address of a PC, from the Command Prompt: **ipconfig /all**

```
Command Prompt
C:\>ipconfig \all
Invalid Command.

C:\>ipconfig /all

FastEthernet0 Connection:(default port)

Connection-specific DNS Suffix...:
Physical Address.....: 0060.2F52.1EAA
Link-local IPv6 Address.....: FE80::260:2FFF:FE52:1EAA
IP Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: 0.0.0.0
DNS Servers.....: 0.0.0.0
DHCP Servers.....: 0.0.0.0
DHCPv6 Client DUID.....: 00-01-00-01-C9-28-3C-D9-00-60-2F-52-1E-AA
```



# Port Security (4/4)

- Dynamic: the allowed MAC addresses are learned dynamically up to a fixed number (1 by default) and saved only in the secure MAC address table
- Sticky Dynamic: the allowed MAC addresses are learned dynamically up to a fixed number and saved in the secure MAC address table and in the running configuration file.

```
S1#configure terminal
S1(config)#interface fastEthernet 0/18
S1(config-if)#switchport mode access
S1(config-if)#switchport port-security
S1(config-if)#switchport port-security
maximum 50
S1(config-if)#switchport port-security mac-
address sticky
S1(config-if)#end
```



# Port Security: violation

- Once a violation occurs on a port, the port will transition to an error disabled state.
- To recover from an error disabled state, a **shutdown** command and then a **no shutdown** on the interface (manual intervention by an administrator) must be executed.
- Error disabled ports can be configured to automatically recover from port security violations:

**S1(config-if)# errdisable recovery interval *Time\_Interval***

- The *Time\_Interval* is an integer value in the range [30-86400] seconds.





# Port Security checking (1/2)

```
switch#show port-security interface fastEthernet 0/18
```

```
Port Security           : Enabled
Port Status             : Secure-down
Violation Mode          : Shutdown
Aging Time              : 0 mins
Aging Type              : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 1
Total MAC Addresses     : 1
Configured MAC Addresses : 0
Sticky MAC Addresses    : 0
Last Source Address:Vlan : 0000.0000.0000:0
Security Violation Count : 0
```



# Port Security checking (1/2)

```
switch#show port-security address
```

```
Secure Mac Address Table
```

```
-----  
Vlan  Mac Address      Type                Ports    Remaining Age (mins)  
99     0050.BAA6.06CE        SecureConfigured    Fa0/18    -  
-----
```

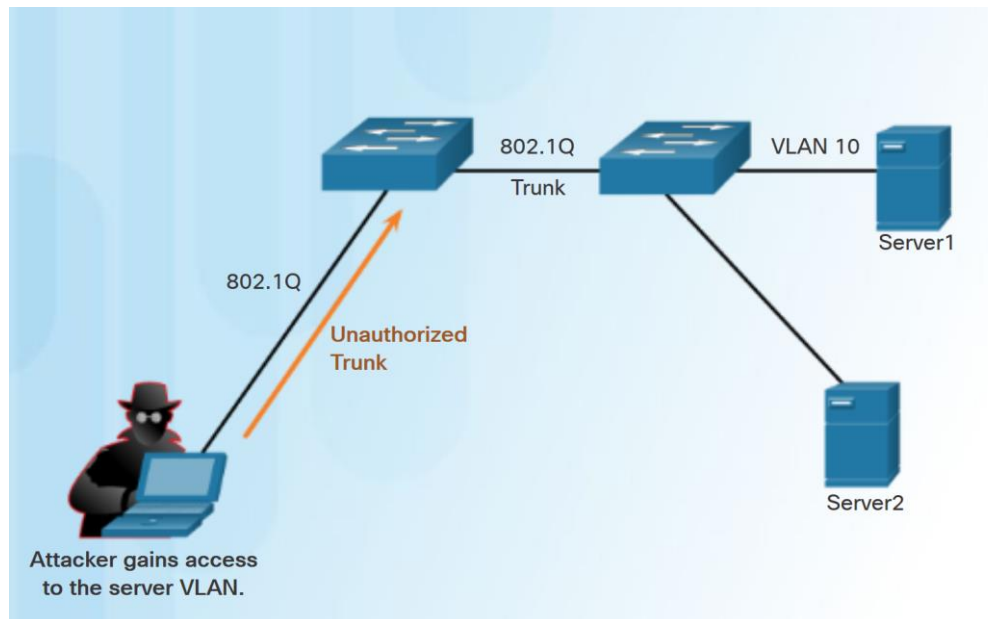
```
Total Addresses in System (excluding one mac per port)    : 0
```

```
Max Addresses limit in System (excluding one mac per port) : 8320
```



# VLAN Attacks

- Attacker can gain VLAN access by configuring a host to spoof a switch:
  - Creating a trunk with an "unsecured" switch
  - Exploiting the "auto trunking" (DTP) configured by default on switch ports





# VLAN Attacks

- Methods to mitigate VLAN attacks:
  - Explicitly configure access ports.
  - Disable auto negotiate on trunks.
  - Manually enable trunk links.
  - Disable unused ports, make them access ports, and assign to a black hole VLAN.
  - Change the default native VLAN.
  - Implement port security.



# VLAN Attacks

- To prevent basic VLAN attacks:
  - Disable DTP (auto trunking) negotiations on trunking and non-trunking ports using **switchport nonegotiate**.
  - Manually enable trunk links using **switchport mode trunk**.
  - Change the native VLAN from VLAN 1.
  - Disable unused ports and assign them to an unused VLAN.



# DHCP Attacks

- DHCP spoofing attack - An attacker configures a fake DHCP server on the network to issue IP addresses to clients.
- DHCP starvation attack - An attacker floods the DHCP server with bogus DHCP requests and leases all of the available IP addresses. This results in a denial-of-service (DoS) attack as new clients cannot obtain an IP address.
- Methods to mitigate DHCP attacks:
  - Configure **DHCP snooping**
  - Configure **port security**



# DHCP Snooping

- With DHCP snooping enabled on an interface, the switch will deny packets containing:
  - Unauthorized DHCP server messages coming from an untrusted port.
- DHCP snooping recognizes two types of ports:
  - **Trusted DHCP ports** - Only ports connecting to upstream DHCP servers should be trusted.
  - **Untrusted ports** - These ports connect to hosts that should not be providing DHCP server messages.



# DHCP Snooping configuration

- Step 1. Enable DHCP snooping globally  
`S1(config)# ip dhcp snooping`
- Step 2. Enable DHCP snooping on selected VLANs (at least one!)  
`S1(config)# ip dhcp snooping vlan vlan-id`
- Step 3. Configure trusted interfaces (untrusted is default)  
`S1(config-if)# ip dhcp snooping trust`
- Step 4. Disable option 82 insertion by switch  
`S1(config)# no ip dhcp snooping information option`
- Step 5. Configure rate limit (optional)  
`S1(config-if)# ip dhcp snooping trust`





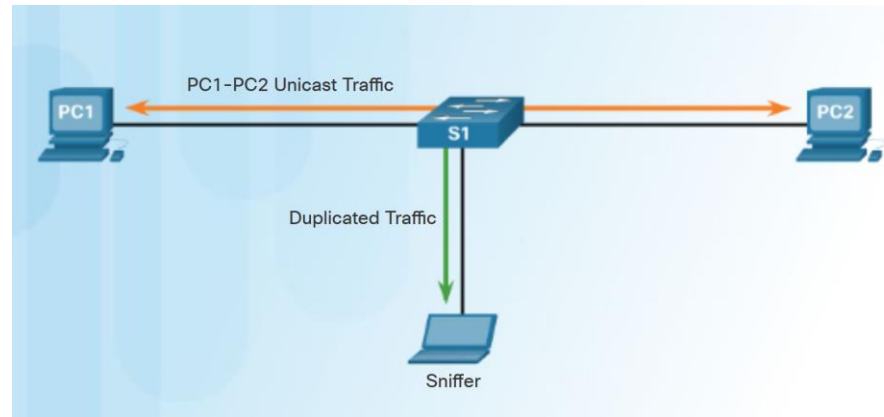
# DHCP Snooping on PT

- Packet Tracer has a bug: DHCP Snooping is not properly supported in case of multiple VLANs



# Port Mirroring

- Port mirroring allows a switch to copy and send Ethernet frames from specific ports to a selected destination port (usually connected to a packet analyzer or to an Intrusion Detection System).



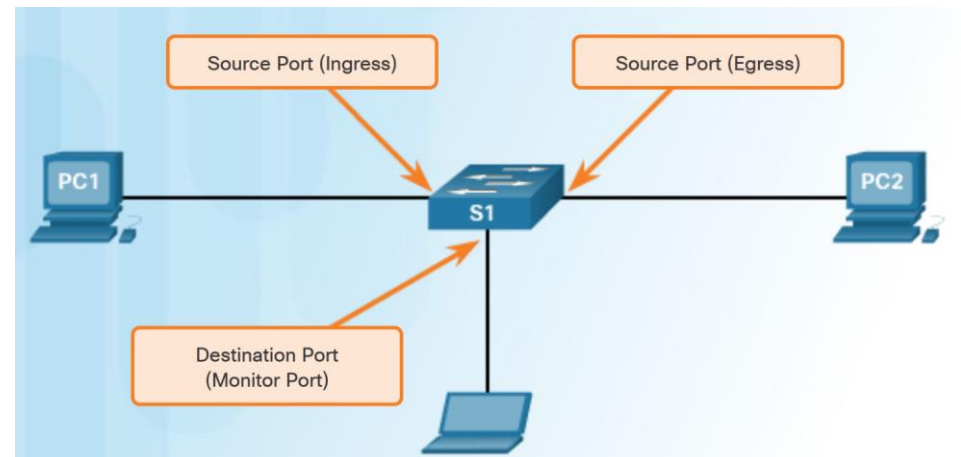
- SPAN is a type of port mirroring
- SPAN can be implemented in two different ways: Local SPAN or Remote SPAN.



# Local SPAN

- Local SPAN: mirroring is performed on the same switch.
- A SPAN session is the association between source ports (or VLANs) and a destination port.
- The destination port is no longer a normal switch port. Only monitored traffic passes through that port.

Term	Definition
Ingress traffic	This is traffic that enters the switch.
Egress traffic	This is traffic that leaves the switch.
Source (SPAN) port	This is a port that is monitored with use of the SPAN feature.
Destination (SPAN) port	This is a port that monitors source ports, usually where a packet analyzer, IDS or IPS is connected. This port is also called the monitor port.
SPAN session	This is an association of a destination port with one or more source ports.
Source VLAN	This is the VLAN monitored for traffic analysis.





# Configuring local SPAN

- A session number is used to identify a local SPAN session.
- The command ***monitor session*** configure SPAN with a source port and a destination port.
- A VLAN can be specified instead of a physical port.
- The ***show monitor*** command is used to check the SPAN sessions.



```
S1(config)# monitor session 1 source interface fastethernet 0/1
S1(config)# monitor session 1 destination interface fastethernet 0/2
```

```
S1# show monitor
Session 1
-----
Type                : Local Session
Source Ports        :
  Both              : Fa0/1
Destination Ports    : Fa0/2
Encapsulation        : Native
  Ingress           : Disabled

S1#
```



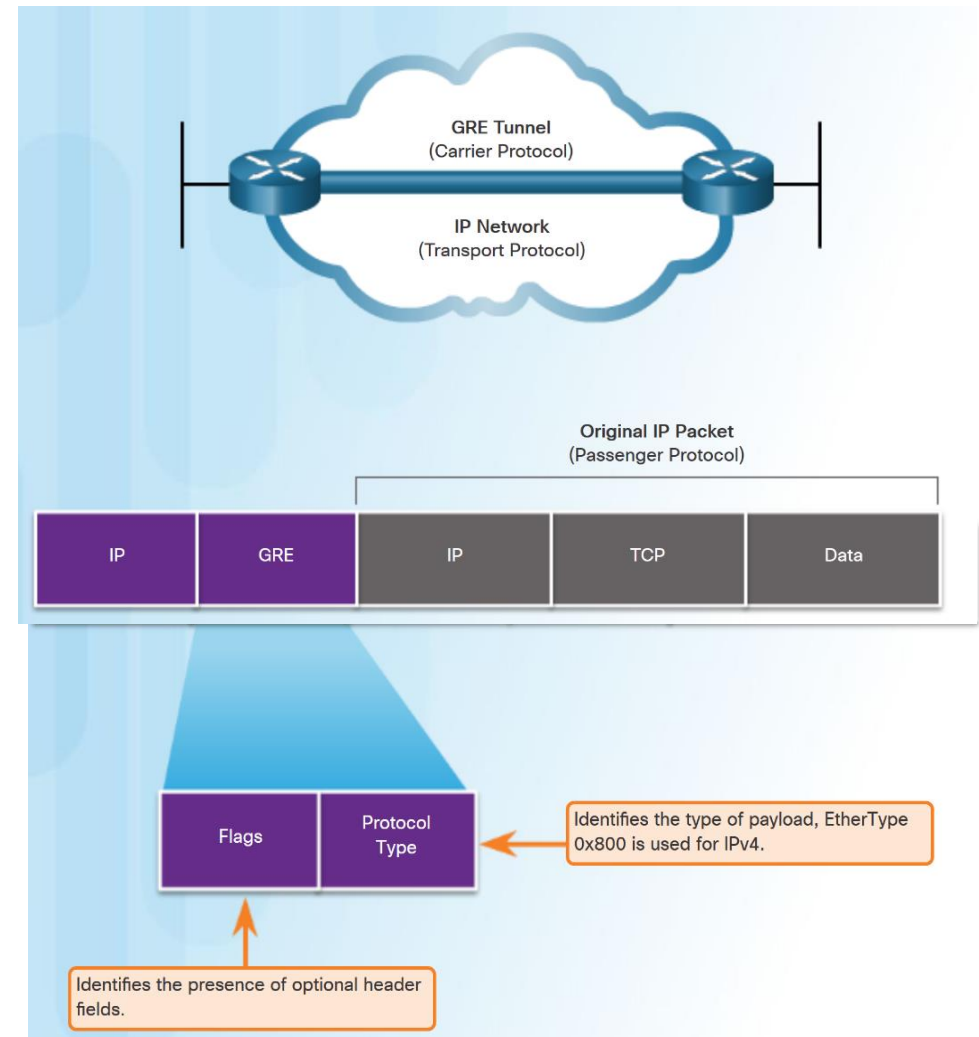
# Virtual Private Network (VPN)

- A VPN is a private network created over a public network (Internet) exploiting the tunneling mechanism.
- Tunneling: encapsulation of a packet (frame) with an outer header. Many different tunneling mechanisms have been defined depend on the type of
- A VPN can also provide security functions, such as encryption: IPsec.
- A VPN gateway is necessary: it could be a router or a dedicated device.
- The VPN gateway is responsible for encapsulating and encrypting outbound traffic (and decapsulating and decrypting inbound traffic).



# Generic Routing Encapsulation (GRE)

- Generic Routing Encapsulation (GRE) is a non-secure, site-to-site VPN tunneling protocol.
- GRE is defined as an IETF standard (RFC 2784).
- In the outer IP header, 47 is used in the protocol field.
- GRE encapsulation uses a protocol type field in the GRE header to support the encapsulation of any OSI Layer 3 protocol.





# Configuring GRE (1/2)

- Create a tunnel interface.

***R(config)# interface Tunnel X***

- Configure GRE as the tunnel interface mode.

***R(config-if)# tunnel mode gre ip***

- Assign an IP address to the tunnel interface (usually a private address)

- Specify the tunnel source physical interface.

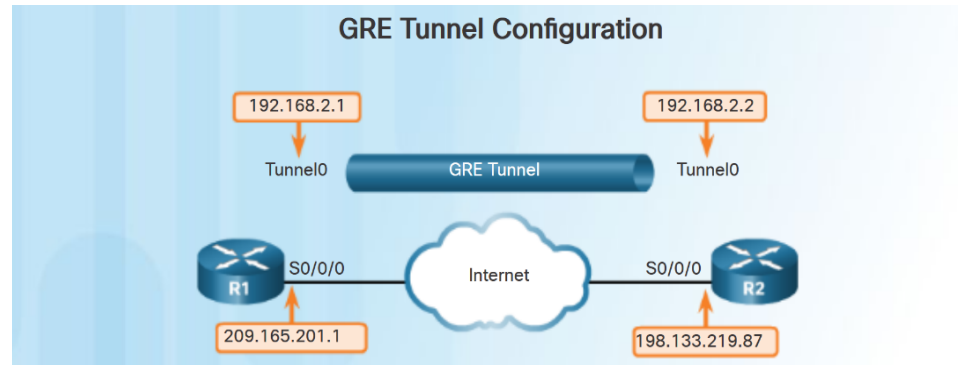
***R(config-if)# tunnel source InterfaceID***

- Specify the tunnel destination IP address (IP address of the physical interface of the tunnel end point).

***R(config-if)# tunnel destination IP\_Address***



# Configuring GRE (2/2)



***R1(config)# interface Tunnel 0***

***R1(config-if)# tunnel mode gre ip***

***R1(config-if)# ip address 192.168.2.1 255.255.255.252***

***R(config-if)# tunnel source S0/0/0***

***R(config-if)# tunnel destination 198.133.219.87***





# Verifying GRE

```
R1# show ip interface brief | include Tunnel
```

```
Tunnel0          192.168.2.1      YES manual up      up
```

```
R1# show interface Tunnel 0
```

```
Tunnel0 is up, line protocol is up
```

```
Hardware is Tunnel
```

```
Internet address is 192.168.2.1/24
```

```
MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,  
    reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation TUNNEL, loopback not set
```

```
Keepalive not set
```

```
Tunnel source 209.165.201.1, destination 209.165.201.2
```

```
Tunnel protocol/transport GRE/IP
```

```
<output omitted>
```