

VLAN Hijacking

Objective: In this lab I used the Yersinia tool to take advantage of default settings on a switch. By enabling DTP (Dynamic Trunking Protocol) I was able to make the switch form a trunk link with my Kali Linux. This allowed my Kali Linux to receive traffic from different VLANs, including one I wasn't originally part of. The goal was to show how attackers can use weak or default configurations to gain access to network traffic they shouldn't normally see.

Equipment: Kali Linux, 2811 Router, 2960 Switch, and Mac

Steps:

1. Create a DHCP pool for VLAN 2

```
ip routing
ip dhcp excluded-address 192.168.2.254
!
ip dhcp pool Home
  network 192.168.2.0 255.255.255.0
  default-router 192.168.2.254
```

2. Keep interface g1/0/1 on VLAN 1 and move g1/0/2-3 to VLAN 2

```
switcha#sh vl br
VLAN Name          Status    Ports
---- -----
1    default        active   Gi1/0/1, Gi1/0/4, Gi1/0/5
                           Gi1/0/6, Gi1/0/7, Gi1/0/8
                           Gi1/0/9, Gi1/0/10, Gi1/0/11
                           Gi1/0/12, Gi1/0/13, Gi1/0/14
                           Gi1/0/15, Gi1/0/16, Gi1/0/17
                           Gi1/0/18, Gi1/0/19, Gi1/0/20
                           Gi1/0/21, Gi1/0/22, Gi1/0/23
                           Gi1/0/24, Gi1/0/25, Gi1/0/26
                           Gi1/0/27, Gi1/0/28, Gi1/0/29
                           Gi1/0/30, Gi1/0/31, Gi1/0/32
                           Gi1/0/33, Gi1/0/34, Gi1/0/35
                           Gi1/0/36, Gi1/0/37, Gi1/0/38
                           Gi1/0/39, Gi1/0/40, Gi1/0/41
                           Gi1/0/42, Gi1/0/43, Gi1/0/44
                           Gi1/0/45, Gi1/0/46, Gi1/0/47
                           Gi1/0/48, Gi1/0/49, Gi1/0/50
                           Gi1/0/51, Gi1/0/52
2    VLAN0002        active   Gi1/0/2, Gi1/0/3
1002 fddi-default    act/unsup
1003 token-ring-default act/unsup
VLAN Name          Status    Ports
---- -----
1004 fddinet-default act/unsup
1005 trnet-default  act/unsup
switcha#
```

3. With default configurations or poor configurations, **the interface is on auto negotiation**

```
switcha>en
switcha#show int g1/0/1 switchport
Name: Gi1/0/1
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
switcha#
```

4. Go to my Kali Linux to set up the Yersinia attack



5. Select the DTP attack



6. Form the trunk link





7. Verify it works by looking for any changes on my CLI

```
switcha#
Jun 20 12:11:51.548: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
Jun 20 12:11:52.544: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed state to down
Jun 20 12:11:55.557: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed state to up
Jun 20 12:12:24.564: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
```

8. Check my interfaces trunk

```
switcha#sh int trunk

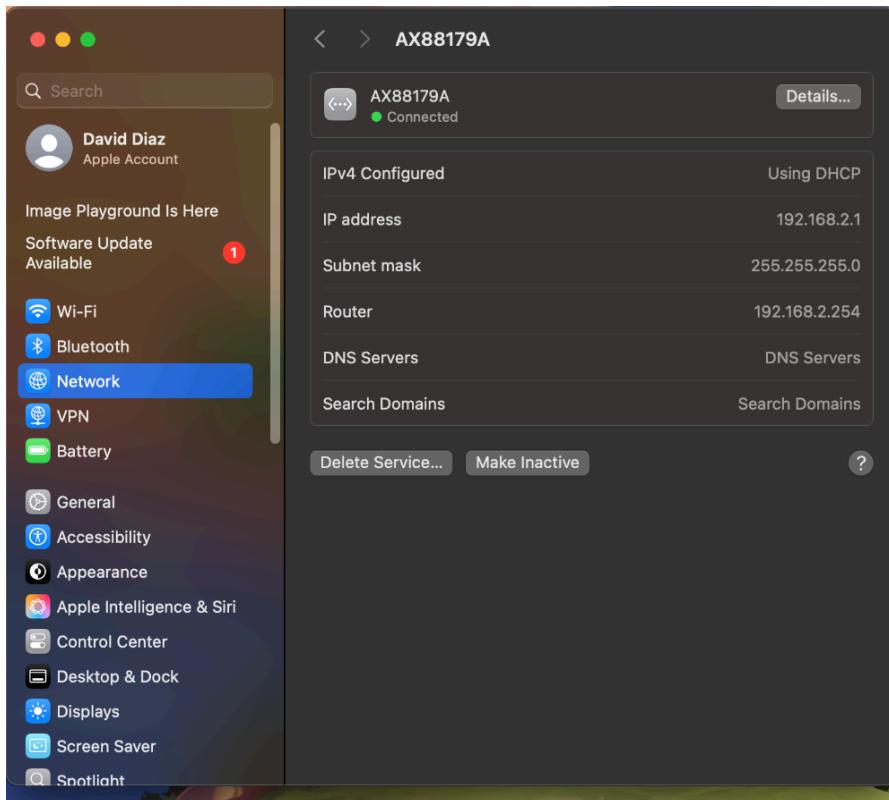
Port      Mode          Encapsulation  Status        Native vlan
Gi1/0/1   auto         802.1q        trunking    1

Port      Vlans allowed on trunk
Gi1/0/1   1-4094

Port      Vlans allowed and active in management domain
Gi1/0/1   1-2

Port      Vlans in spanning tree forwarding state and not pruned
Gi1/0/1   1-2
switcha#
```

9. On VLAN 2 I connect my Mac to the network and do a DHCP request



10. I use wireshark to snoop on traffic and capture the DHCP request, while being on VLAN 1

Wireshark screenshot showing captured traffic on interface eth1. The packet list shows a DHCP Discover request from Cisco f1:05:01 (MAC address 08:00:27:7c:3b:2e) to the broadcast address (FF:FF:FF:FF:FF:FF). The details and bytes panes show the DHCP options, including the Requested IP Address option (Option 50) set to 192.168.2.1.

No.	Time	Source	Destination	Protocol	Length	Info
472	237.152274673	Cisco_f1:05:01	PVST+	STP	68	Conf. Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
61	34.956837942	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
66	36.952880607	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
69	38.952275162	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
71	40.952373864	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
82	42.954399070	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
109	44.957243454	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
129	46.9606881824	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
134	48.959389545	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
143	50.962213025	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
153	52.961544913	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
158	54.964395569	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
163	56.963711077	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
168	58.977937946	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
169	60.988692112	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
173	62.982742555	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
176	64.985543146	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
186	66.984880256	Cisco_f1:05:01	PVST+	STP	68	Conf. TC + Root = 32768/2/ac:f5:e6:f1:05:00 Cost = 0 Port = 0x0001
70	39.030140254	0.0.0.0	255.255.255.255	DHCP	346	DHCP Discover - Transaction ID: 0x5d71cc3b
78	42.64952493	0.0.0.0	255.255.255.255	DHCP	346	DHCP Request - Transaction ID: 0x5d71cc3b
30	17.125119916	Cisco_f1:05:01	CDP/FTP/DTP/Pag/UD_CDP	CDP	468	Device ID: switch1 Port ID: GigabitEthernet1/0/1

Hops: 0
Transaction ID: 0xd71cc3b
Seconds elapsed: 38
Boot flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0
Your (client) IP address: 0.0.0.0
Next server IP address: 0.0.0.0
Relay agent IP address: 0.0.0.0
Client MAC address: CeLink_13:78:8e (6c:6e:07:13:78:8e)
Client hardware address padding: 000000000000000000000000
Server host name not given
Boot file name not given
Magic cookie: DHCP
Option: (53) DHCP Message Type (Request)
Option: (55) Parameter Request List
Option: (57) Maximum DHCP Message Size
Option: (61) Client identifier
Option: (50) Requested IP Address (192.168.2.1)
Option: (54) DHCP Server Identifier (192.168.2.254)
Option: (12) Host Name
Option: (255) End
Padding: 0000000000

Detailed view of the Requested IP Address option (Option 50):
 Option 50: Requested IP Address (192.168.2.1)
 Option 54: DHCP Server Identifier (192.168.2.254)
 Option 12: Host Name
 Option 255: End
 Padding: 0000000000

11. In order to stop this attack, **I have to configure my port with no negotiation** (in real world situations, it should be applied to every interface). This ensures that no DTP can be performed

```
switcha(config)#int g1/0/1
switcha(config-if)#switchport nonegotiate
switcha(config-if)#do sh int g1/0/1 switchport
Name: Gi1/0/1
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
switcha(config-if)#[
```

