

Wireless Access Point Lab

Aaron Yu

Lab Partners: Jacob Chen and Aiden Yin

Cisco CCNP

# Purpose

The purpose of this lab was to configure a wireless access point (AP) to broadcast 3 wireless networks that end devices could connect to, with certain security measures to ensure only authenticated users could access these networks.

# Background Information

APs are devices that allow end devices to connect to wirelessly connect to a wired network to, for example, access the internet. These networks follow the IEEE 802.11 set of standards for wireless local area networks (WLAN), whose purpose is to ensure compatibility among different manufacturers. These networks can be configured with SSIDs (service set identifier; name of network), VLANs (to separate traffic from different networks), authentication methods, and others that we did not use in this lab. We used two types of authentication in this lab – a pre-shared key (PSK) and a RADIUS (remote authentication dial-in user service) server.

The PSK is sort of like the generic standard for Wi-Fi security – a password is created and shared with desired users, and those users can access the network while others without the password are not. The RADIUS server is different – it allows for multiple credentials to be able to access the same network and allows for fine tuning of the authorization and accounting of a user. RADIUS also does not require authentication at the access point but is instead authentication is done by sending a request to a centralized server, and letting that machine do the work, meaning for networks with multiple devices with need of AAA, the devices only need to connect with that server. RADIUS is also very scalable, allowing many different accounts with different credentials and access policies to be configured.

We configured the RADIUS server using a barebones Ubuntu setup with the FreeRADIUS software. Ubuntu is a popular Linux distribution used for many different types of devices. FreeRADIUS is a free solution for hosting a RADIUS server on Linux machines.

There are two types of APs, as well – lightweight and autonomous. Lightweight APs rely on external WLAN controllers (WLC) to be able to function, but the autonomous APs can function by themselves. In this lab, we used an autonomous AP.

# Lab Summary

We started up by replacing the OS of the AP. The current OS configured was made for lightweight APs, so we had to replace it with an autonomous AP OS. We used a TFTP server to import the image to our AP, then used a given script to switch the image. After setting this up, we researched on how to create and configure our networks with names and security measures. We first configured the PSK networks Guest and Personal, setting up SSIDs, PSKs, and VLANs. We then got stuck on how to set up the RADIUS server, but eventually we figured out we could just set up a Linux device to broadcast FreeRADIUS, and have the AP use that. We finally configured the RADIUS server on our AP, along with DHCP and NAT, debugged any issues we had, then tested for proper connectivity and behavior of the AP.

# Lab Commands

dot11 ssid <SSID name>: Configure an SSID with the configured name.

authentication open: Enables open authentication for SSID

authentication open eap eap\_methods: Enables open authentication with EAP for SSID

authentication key-management wpa version <version #>: Configures WPA version for SSID

wpa-psk ascii <password>: Configures PSK and encrypts the password in config for SSID

mbssid guest-mode: Enable MBSSID (multiple basic SSIDs) with guest mode on the SSID

vlan <vlan #>: Set the VLAN the SSID will use

interface <interface>

encryption vlan <vlan #> mode ciphers aes-ccm: Enable AES with CCM mode to encrypt traffic going through the specified VLAN

ssid <SSID #>: Configure SSIDs on the interface

encapsulation dot1Q <VLAN #> <native ?>: Apply VLAN to a sub interface and determine whether it is the native VLAN or not

ip nat <outside|inside>: Configure interface to be outside or inside port for NAT translations

switchport trunk encapsulation dot1q: Use 802.1Q encapsulation for trunking on the interface

switchport trunk allowed vlan <VLAN list>: Configure VLANs that are allowed to use the trunk interface

switchport mode trunk: Enable trunking for interface

radius server <name>: Configure server name of remote RADIUS server and enter configuration for specified server

address ipv4 <IP> auth-port <port #> acct-port <port #>: Configure RADIUS server IP, UDP port for authentication, and UDP port for accounting.

timeout <seconds>: Configure wait time for RADIUS server to respond before retransmitting request

retransmit <retries>: Configure retransmit attempts if RADIUS server does not send back a response

key <key>: Configure a key to encrypt communication between RADIUS

eap profile <name>: Create and enter configuration for an EAP profile

method mschapv2: Configure the EAP-MSCHAPV2 method on EAP profile

aaa group server radius <name>: Create and enter configuration for an AAA server group

server name <name>: Configure the name of the server used by the AAA server group

aaa authentication login eap\_methods group <name>: Configures the AP to use EAP with the specified group

wlccp ap eap profile <name>: Configure the EAP profile the WLC will use

ip dhcp excluded-address <IP>: Exclude IP from DHCP IP distribution

ip dhcp pool <name>: Create and enter configuration for a DHCP pool

network <IP> <mask>: Confiugre network that DHCP pool can use to distribute IPs

dns-server <IP>: Configure DNS server that DHCP server gives out to clients

default-router <IP>: Configure default gateway that DHCP server gives out to clients

ip nat <inside|outside> source list <access-list #> interface <interface> overload: Configure traffic on the interface on the inside of the network and within the access list to use NAT, and overload NAT addressing with PAT

access-list <#> permit <IP> <mask>: Configure specified access list to allow traffic from specified network

# Network Diagram

A diagram of a network

Description automatically generated

# Configurations

## show run

### AP

Current configuration : 4029 bytes

! Last configuration change at 04:58:51 UTC Fri Mar 1 2002

version 15.3

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

service password-encryption

hostname AP

logging rate-limit console 9

enable secret 5 $1$ySJC$WKGeN6ausDAtFkMop0BIE1

aaa new-model

aaa group server radius AP\_RAD\_GROUP

server name Jacob\_PC

aaa authentication login eap\_methods group AP\_RAD\_GROUP

aaa session-id common

no ip source-route

no ip cef

no ip domain lookup

dot11 pause-time 100

dot11 syslog

dot11 ssid JacobAaronAiden\_67\_CCNP\_Ent

vlan 13

authentication open eap eap\_methods

authentication network-eap eap\_methods

authentication key-management wpa version 2

mbssid guest-mode

dot11 ssid JacobAaronAiden\_67\_CCNP\_Guest

vlan 11

authentication open

mbssid guest-mode

dot11 ssid JacobAaronAiden\_67\_CCNP\_Personal

vlan 12

authentication open

authentication key-management wpa version 2

mbssid guest-mode

wpa-psk ascii 7 106D000A061843595F

eap profile AUTH\_METHODS

method mschapv2

no ipv6 cef

username Cisco password 7 01300F175804

bridge irb

interface Dot11Radio0

no ip address

shutdown

antenna gain 0

station-role root

bridge-group 1

bridge-group 1 subscriber-loop-control

bridge-group 1 spanning-disabled

bridge-group 1 block-unknown-source

no bridge-group 1 source-learning

no bridge-group 1 unicast-flooding

interface Dot11Radio1

no ip address

encryption vlan 12 mode ciphers aes-ccm

encryption vlan 13 mode ciphers aes-ccm

ssid JacobAaronAiden\_67\_CCNP\_Ent

ssid JacobAaronAiden\_67\_CCNP\_Guest

ssid JacobAaronAiden\_67\_CCNP\_Personal

antenna gain 0

peakdetect

dfs band 3 block

mbssid

channel dfs

station-role root

interface Dot11Radio1.1

encapsulation dot1Q 1 native

bridge-group 1

bridge-group 1 subscriber-loop-control

bridge-group 1 spanning-disabled

bridge-group 1 block-unknown-source

no bridge-group 1 source-learning

no bridge-group 1 unicast-flooding

interface Dot11Radio1.11

encapsulation dot1Q 11

bridge-group 11

bridge-group 11 subscriber-loop-control

bridge-group 11 spanning-disabled

bridge-group 11 block-unknown-source

no bridge-group 11 source-learning

no bridge-group 11 unicast-flooding

interface Dot11Radio1.12

encapsulation dot1Q 12

bridge-group 12

bridge-group 12 subscriber-loop-control

bridge-group 12 spanning-disabled

bridge-group 12 block-unknown-source

no bridge-group 12 source-learning

no bridge-group 12 unicast-flooding

interface Dot11Radio1.13

encapsulation dot1Q 13

bridge-group 13

bridge-group 13 subscriber-loop-control

bridge-group 13 spanning-disabled

bridge-group 13 block-unknown-source

no bridge-group 13 source-learning

no bridge-group 13 unicast-flooding

interface GigabitEthernet0

no ip address

duplex auto

speed auto

dot1x pae authenticator

interface GigabitEthernet0.1

encapsulation dot1Q 1 native

bridge-group 1

bridge-group 1 spanning-disabled

no bridge-group 1 source-learning

interface GigabitEthernet0.11

encapsulation dot1Q 11

bridge-group 11

bridge-group 11 spanning-disabled

no bridge-group 11 source-learning

interface GigabitEthernet0.12

encapsulation dot1Q 12

bridge-group 12

bridge-group 12 spanning-disabled

no bridge-group 12 source-learning

interface GigabitEthernet0.13

encapsulation dot1Q 13

bridge-group 13

bridge-group 13 spanning-disabled

no bridge-group 13 source-learning

interface BVI1

mac-address 44d3.ca03.7d9b

ip address 10.0.11.200 255.255.255.0

ipv6 address dhcp

ipv6 address autoconfig

ipv6 enable

ip forward-protocol nd

ip http server

no ip http secure-server

ip http help-path http://www.cisco.com/warp/public/779/smbiz/prodconfig/help/eag

radius server Jacob\_PC

address ipv4 10.0.11.100 auth-port 1812 acct-port 1813

timeout 10

retransmit 3

key 7 111D1C16031B050B557878

bridge 1 route ip

wlccp ap eap profile AUTH\_METHODS

line con 0

logging synchronous

line vty 0 4

transport input all

end

### Router

Current configuration : 2406 bytes

! Last configuration change at 21:10:12 UTC Thu Jun 6 2024

version 15.5

service timestamps debug datetime msec

service timestamps log datetime msec

no platform punt-keepalive disable-kernel-core

hostname R1

boot-start-marker

boot-end-marker

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

no aaa new-model

ip dhcp excluded-address 192.168.51.1

ip dhcp excluded-address 192.168.52.1

ip dhcp excluded-address 192.168.53.1

ip dhcp pool Guest

network 192.168.51.0 255.255.255.0

dns-server 1.1.1.1

default-router 192.168.51.1

ip dhcp pool Psk

network 192.168.52.0 255.255.255.0

dns-server 1.1.1.1

default-router 192.168.52.1

ip dhcp pool Enterprise

network 192.168.53.0 255.255.255.0

dns-server 1.1.1.1

default-router 192.168.53.1

subscriber templating

multilink bundle-name authenticated

license udi pid ISR4321/K9 sn FDO211216BL

license accept end user agreement

spanning-tree extend system-id

redundancy

mode none

vlan internal allocation policy ascending

interface GigabitEthernet0/0/0

no ip address

ip nat inside

negotiation auto

interface GigabitEthernet0/0/0.1

encapsulation dot1Q 1 native

ip address 10.0.11.1 255.255.255.0

interface GigabitEthernet0/0/0.11

encapsulation dot1Q 11

ip address 192.168.51.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/0/0.12

encapsulation dot1Q 12

ip address 192.168.52.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/0/0.13

encapsulation dot1Q 13

ip address 192.168.53.1 255.255.255.0

ip nat inside

interface GigabitEthernet0/0/1

ip address dhcp

ip nat outside

negotiation auto

interface Serial0/1/0

no ip address

shutdown

interface Serial0/1/1

no ip address

shutdown

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

interface Vlan1

no ip address

ip nat inside source list 1 interface GigabitEthernet0/0/1 overload

ip forward-protocol nd

no ip http server

no ip http secure-server

ip tftp source-interface GigabitEthernet0

access-list 1 permit 192.168.51.0 0.0.0.255

access-list 1 permit 192.168.52.0 0.0.0.255

access-list 1 permit 192.168.53.0 0.0.0.255

control-plane

line con 0

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

ntp server pool.ntp.org

end

### Switch

Current configuration : 2381 bytes

version 12.2

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

hostname S1

boot-start-marker

boot-end-marker

no aaa new-model

system mtu routing 1500

authentication mac-move permit

ip subnet-zero

spanning-tree mode pvst

spanning-tree etherchannel guard misconfig

spanning-tree extend system-id

vlan internal allocation policy ascending

interface FastEthernet0/1

shutdown

interface FastEthernet0/2

switchport trunk encapsulation dot1q

switchport trunk allowed vlan 1,11-13

switchport mode trunk

interface FastEthernet0/3

interface FastEthernet0/4

interface FastEthernet0/5

interface FastEthernet0/6

interface FastEthernet0/7

interface FastEthernet0/8

interface FastEthernet0/9

interface FastEthernet0/10

switchport mode access

interface FastEthernet0/11

interface FastEthernet0/12

interface FastEthernet0/13

interface FastEthernet0/14

interface FastEthernet0/15

interface FastEthernet0/16

interface FastEthernet0/17

interface FastEthernet0/18

interface FastEthernet0/19

interface FastEthernet0/20

interface FastEthernet0/21

interface FastEthernet0/22

interface FastEthernet0/23

interface FastEthernet0/24

interface FastEthernet0/25

interface FastEthernet0/26

interface FastEthernet0/27

interface FastEthernet0/28

interface FastEthernet0/29

interface FastEthernet0/30

switchport trunk encapsulation dot1q

switchport trunk allowed vlan 1,11-13

switchport mode trunk

interface FastEthernet0/31

interface FastEthernet0/32

interface FastEthernet0/33

interface FastEthernet0/34

interface FastEthernet0/35

interface FastEthernet0/36

interface FastEthernet0/37

interface FastEthernet0/38

interface FastEthernet0/39

interface FastEthernet0/40

interface FastEthernet0/41

interface FastEthernet0/42

interface FastEthernet0/43

interface FastEthernet0/44

interface FastEthernet0/45

interface FastEthernet0/46

interface FastEthernet0/47

interface FastEthernet0/48

interface GigabitEthernet0/1

interface GigabitEthernet0/2

interface GigabitEthernet0/3

interface GigabitEthernet0/4

interface Vlan1

no ip address

shutdown

ip classless

ip http server

ip http secure-server

ip sla enable reaction-alerts

line con 0

line vty 0 4

login

line vty 5 15

login

end

## show vlan

### AP

Virtual LAN ID: 1 (IEEE 802.1Q Encapsulation)

vLAN Trunk Interfaces: Dot11Radio1.1

GigabitEthernet0.1

This is configured as native Vlan for the following interface(s) :

Dot11Radio1

GigabitEthernet0

Protocols Configured: Address: Received: Transmitted:

Dot11Radio1.1 (1)

Bridging Bridge Group 1 0 58

Other 0 12

0 packets, 0 bytes input

40 packets, 9081 bytes output

GigabitEthernet0.1 (1)

Bridging Bridge Group 1 0 58

Other 0 12

0 packets, 0 bytes input

30 packets, 6116 bytes output

Virtual LAN ID: 11 (IEEE 802.1Q Encapsulation)

vLAN Trunk Interfaces: Dot11Radio1.11

GigabitEthernet0.11

Protocols Configured: Address: Received: Transmitted:

Dot11Radio1.11 (11)

Bridging Bridge Group 11 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

GigabitEthernet0.11 (11)

Bridging Bridge Group 11 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

Virtual LAN ID: 12 (IEEE 802.1Q Encapsulation)

vLAN Trunk Interfaces: Dot11Radio1.12

GigabitEthernet0.12

Protocols Configured: Address: Received: Transmitted:

Dot11Radio1.12 (12)

Bridging Bridge Group 12 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

GigabitEthernet0.12 (12)

Bridging Bridge Group 12 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

Virtual LAN ID: 13 (IEEE 802.1Q Encapsulation)

vLAN Trunk Interfaces: Dot11Radio1.13

GigabitEthernet0.13

Protocols Configured: Address: Received: Transmitted:

Dot11Radio1.13 (13)

Bridging Bridge Group 13 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

GigabitEthernet0.13 (13)

Bridging Bridge Group 13 0 0

0 packets, 0 bytes input

0 packets, 0 bytes output

## show ip route

### Router

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

a - application route

+ - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is 192.168.40.1 to network 0.0.0.0

S\* 0.0.0.0/0 [254/0] via 192.168.40.1

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.0.11.0/24 is directly connected, GigabitEthernet0/0/0.1

L 10.0.11.1/32 is directly connected, GigabitEthernet0/0/0.1

C 192.168.40.0/23 is directly connected, GigabitEthernet0/0/1

192.168.40.0/32 is subnetted, 1 subnets

L 192.168.40.236 is directly connected, GigabitEthernet0/0/1

192.168.51.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.51.0/24 is directly connected, GigabitEthernet0/0/0.11

L 192.168.51.1/32 is directly connected, GigabitEthernet0/0/0.11

192.168.52.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.52.0/24 is directly connected, GigabitEthernet0/0/0.12

L 192.168.52.1/32 is directly connected, GigabitEthernet0/0/0.12

192.168.53.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.53.0/24 is directly connected, GigabitEthernet0/0/0.13

L 192.168.53.1/32 is directly connected, GigabitEthernet0/0/0.13

## show ip nat translations

### Router

Pro Inside global Inside local Outside local Outside global

tcp 192.168.40.236:4603 192.168.53.2:59045 104.16.80.230:443 104.16.80.230:443

tcp 192.168.40.236:4691 192.168.52.2:64828 44.198.23.94:443 44.198.23.94:443

tcp 192.168.40.236:4614 192.168.51.2:52875 104.16.80.230:443 104.16.80.230:443

tcp 192.168.40.236:4613 192.168.51.2:52868 104.16.80.230:443 104.16.80.230:443

tcp 192.168.40.236:4619 192.168.52.2:64709 20.25.227.174:443 20.25.227.174:443

tcp 192.168.40.236:4675 192.168.52.2:64776 52.73.235.185:443 52.73.235.185:443

tcp 192.168.40.236:4604 192.168.51.2:56532 184.28.50.162:443 184.28.50.162:443

tcp 192.168.40.236:4696 192.168.52.2:64842 34.199.53.216:443 34.199.53.216:443

tcp 192.168.40.236:4605 192.168.51.2:56534 35.186.224.39:443 35.186.224.39:443

tcp 192.168.40.236:4608 192.168.53.2:53519 104.17.143.163:443 104.17.143.163:443

tcp 192.168.40.236:4580 192.168.51.2:52838 104.16.80.230:443 104.16.80.230:443

## show ip dhcp bindings

### Router

Bindings from all pools not associated with VRF:

IP address Client-ID/ Lease expiration Type State Interface

Hardware address/

User name

192.168.51.3 0114.abc5.36cc.2a Jun 07 2024 09:39 PM Automatic Active GigabitEthernet0/0/0.11

192.168.52.2 01ac.ed5c.3e56.89 Jun 07 2024 09:08 PM Automatic Active GigabitEthernet0/0/0.12

192.168.52.3 0114.abc5.36cc.2a Jun 07 2024 09:40 PM Automatic Active GigabitEthernet0/0/0.12

192.168.53.2 01ac.ed5c.3e56.89 Jun 07 2024 05:52 PM Automatic Active GigabitEthernet0/0/0.13

## show dot11 bssid

### AP

Interface BSSID Guest SSID

Dot11Radio1 64d9.89bd.f050 Yes JacobAaronAiden\_67\_CCNP\_Ent

Dot11Radio1 64d9.89bd.f051 Yes JacobAaronAiden\_67\_CCNP\_Guest

Dot11Radio1 64d9.89bd.f052 Yes JacobAaronAiden\_67\_CCNP\_Personal

# Problems

We faced a few problems during this lab, mainly when configuring RADIUS. Our first issue was not being able to configure the AP to broadcast multiple WLANs. This issue was resolved quickly, as a bit of research found that we needed to configure MBSSID. Our next issue was figuring out how to host the RADIUS server. For RADIUS, we knew that using FreeRADIUS would be best for us, but the methods we used for a Linux machine took a while to figure out. We first tried the Windows Subsystem for Linux (WSL), a VM to easily host a Linux instance on a Windows machine, but this did not work because port forwarding is not supported on WSL. Our next attempt was to use Docker, which is containerizing software to host instances of OSs for different purposes. In our case, we wanted to run Linux with FreeRADIUS using Docker, but it did not work even after debugging, and we still do not know why. We finally settled for a physical Linux installation with FreeRADIUS, and that worked, but was something we were trying to avoid as we didn’t want to set up a new Linux device just for RADIUS.

# Conclusion

In this lab we learned how to configure an AP to broadcast three WLANs for connectivity, with different security measures put in place to only allow desired users onto a network. Most of the time spent was on figuring out how to set up FreeRADIUS, since many different methods came up short for many reasons which were beyond our control.

