



Getting Personal with WPA3 Wi-Fi Security

Keith Bogart

Cisco CCIE #4923

Key Concepts

- + Wi-Fi Key Management and Cryptographic Foundations
- + Evolution of Wi-Fi Security Protocols (WPA2 → WPA3)
- + Protecting Management and Open-Network Communications
- + WPA3 Enhancements

MAJOR TOPICS

- + The Role and Hierarchy of Keys in Wi-Fi Security
- + Pairwise Key Generation
- + Cryptographic Derivation and Handshake Mechanics
- + Transition from WPA2 to WPA3 and the Rise of SAE
- + Protecting Management Frames and Preventing Spoofing
- + Emerging Wi-Fi Security Enhancements: OWE & DPP



LEARNING OUTCOMES

- + Explain the Complete Wi-Fi Key Hierarchy and Its Functions
- + Differentiate Between WPA2 (PSK) and WPA3 (SAE) Key-Derivation Processes
- + Analyze the Importance and Operation of WPA3 Protected Management Frames (PMF) & Beacon Protection
- + Configure and Validate Modern Wi-Fi Security Enhancements

- + Familiarity with basic 802.11 WLAN terminology
- + Experience Connecting to WPA2 and WPA3 WLANs
- + General familiarity with Cryptographic Hashing functionality

PREREQUISITES



LET'S GO!





Recap of Wi-Fi Security Fundamentals

What You Should Know



- + Why Wi-Fi needs security
- + The differences between confidentiality, authentication, and data integrity
- + Main differences between WPA and WPA2
- + How to create a WLAN with WPA2 authentication

Evolution of Wi-Fi Security Protocols

WEP

- Introduced 1997
- RC4 cipher & 40/104-bit keys
- Broken due to IV reuse

WPA

- Introduced 2003
- TKIP with RC4
- Introduced “Personal” & “Enterprise” implementations
- Transition measure

WPA2

- Ratified 2004
- AES-CCMP
- Offline attacks possible

WPA3

- Released 2018
- SAE & PMF mandatory
- Forward secrecy

- + WEP: Old, weak, easily crackable. Not supported in most devices
- + WPA: Pre-standard version of 802.11i. Deprecated by Wi-Fi Alliance
- + WPA2: De facto standard in most Wi-Fi devices. Crackable when using weak passwords
- + WPA3: Latest standard, and extremely strong. Not supported on some devices.

Personal vs. Enterprise Authentication

Our Course Will Concentrate On This.



Personal (PSK/SAE)

- Pre-Shared Key for home networks
- WPA2: vulnerable to dictionary attacks if PSK is weak
- WPA3: SAE (PAKE) resists offline attacks and provides forward secrecy



Enterprise (802.1X/EAP)

- Uses 802.1X & RADIUS for dynamic per-session keys
- EAP methods (PEAP, EAP-TLS) provide mutual authentication
- Clients must validate server certificates to avoid rogue APs

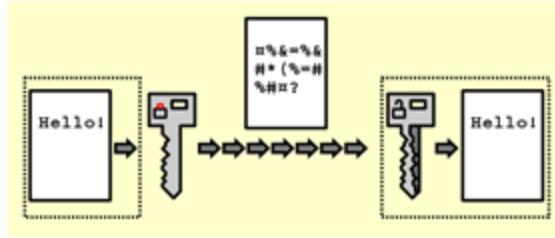


**Thank you for
watching!!**



Introduction to the RSN Framework

Why Keys Matter



- + All of Wi-Fi security revolves around keys
 - + Basis of wireless confidentiality
 - + Foundation for session integrity
 - + Stops replay and impersonation attacks
- + Several types of keys exist for different purposes
- + Derivation and distribution of keys defined by the RSN

Robust Security Network

- + The RSN framework was introduced in 802.11i (2004)
- + Elements of the framework include;
 - + How keys are derived and managed,
 - + Which cipher suites are allowed (AES-CCMP, TKIP, etc.),
 - + How the 4-way EAPOL handshake works,
 - + The concept of RSNIE (Robust Security Network Information Element) — a field in management frames that tells clients and APs which ciphers and authentication methods are supported.

RSN Information Element

```
> IEEE 802.11 Beacon frame, Flags: .....C
v IEEE 802.11 Wireless Management
  > Fixed parameters (12 bytes)
  v Tagged parameters (309 bytes)
    > Tag: SSID parameter set: "WPA-Transition"
    > Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
    > Tag: DS Parameter set: Current Channel: 100
    > Tag: Traffic Indication Map (TIM): DTIM 0 of 1 bitmap
    > Tag: Country Information: Country Code US, Environment All
    > Tag: Power Constraint: 0
    v Tag: RSN Information
      Tag Number: RSN Information (48)
      Tag length: 30
      RSN Version: 1
      > Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)
      Pairwise Cipher Suite Count: 1
      > Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)
      Auth Key Management (AKM) Suite Count: 2
      > Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) PSK 00:0f:ac (Ieee 802.11) SAE (SHA256)
      > RSN Capabilities: 0x00a8
      PMKID Count: 0
      PMKID List
      > Group Management Cipher Suite: 00:0f:ac (Ieee 802.11) BIP (128)
```

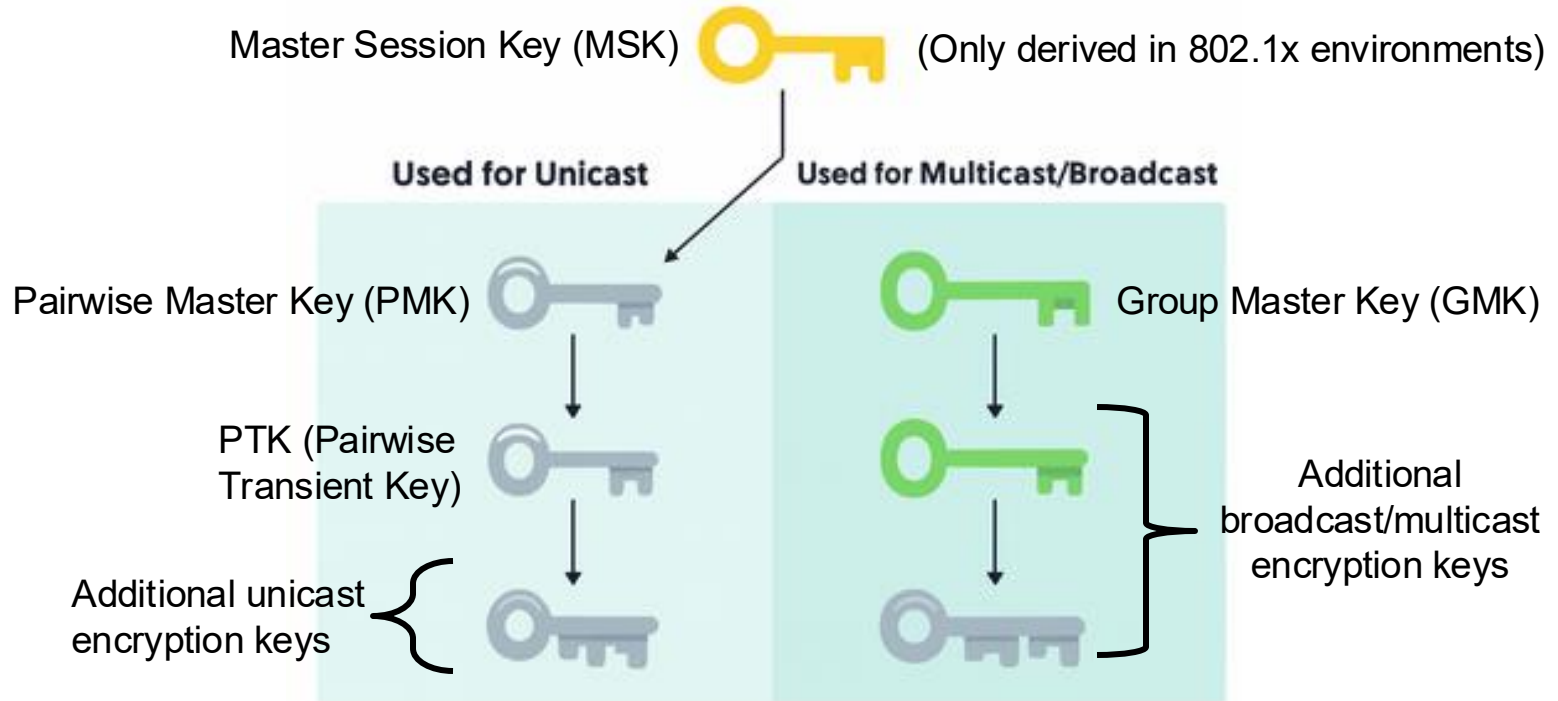
RSN IE

RSN Key Hierarchy

- + RSN defined a hierarchy of keys that are derived from each other
- + Different keys are used for different purposes
 - + Encryption of unicast frames
 - + Encryption of broadcast & multicast frames
 - + Integrity verification of some 802.11 Management frames
- + Fresh keys are derived after each association with an access point

Overview of RSN Keys

- + Once authentication has been completed, RSN dictates the generation and distribution of several types of keys



Pairwise

- + Many keys contain the word, “pairwise” in the name...but what does this mean?
- + Pairwise: a one-to-one security relationship between exactly two devices—the AP and one client (STA).
- + Pairwise keys provide *per-client isolation*.
 - + If someone learns your pairwise keys, they still can't decrypt other clients' unicast traffic on the same SSID
 - + Each client has its own distinct keys.



**Thank you for
watching!!**

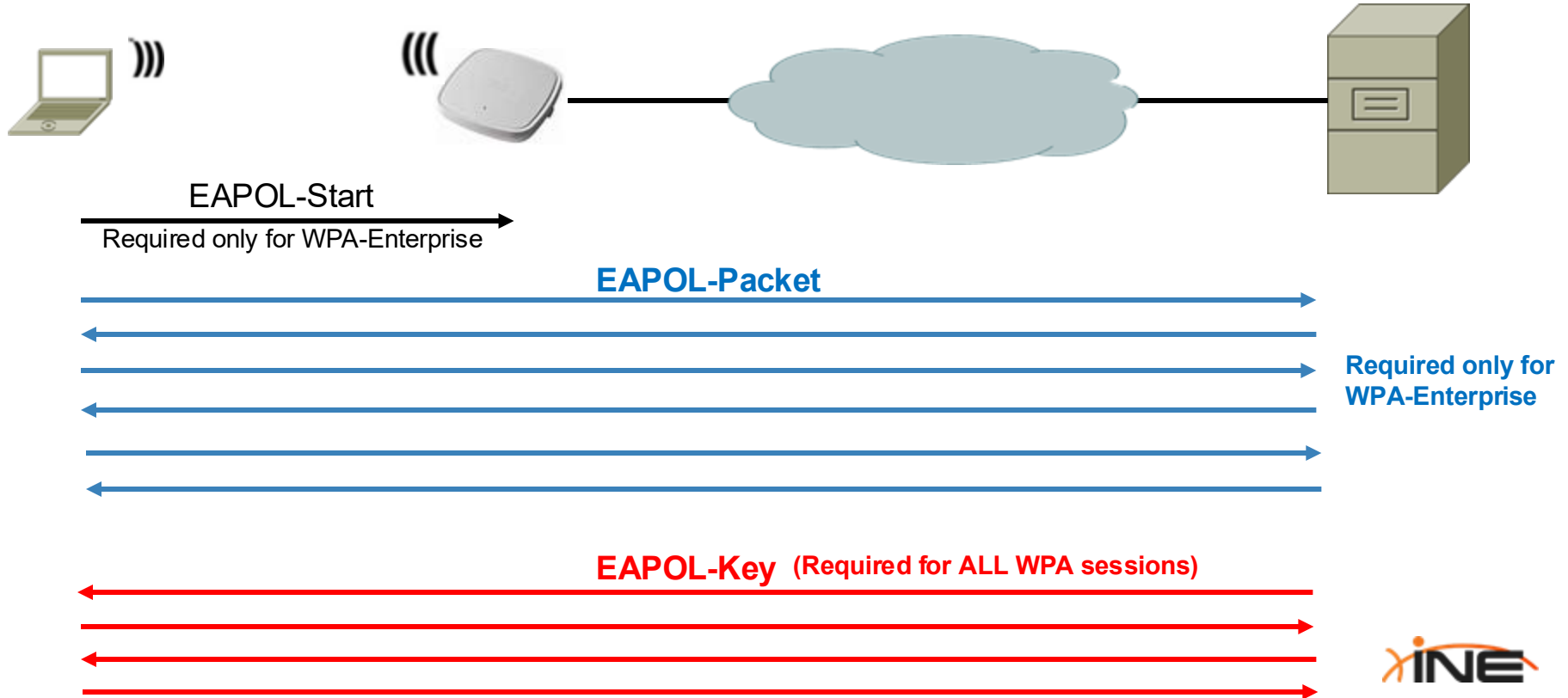


The EAP Framework

The EAP Framework

- + RFC 5247 defines the EAP (*Extensible Authentication Protocol*) Framework consisting of:
 - + Three defined roles of Supplicant, Authenticator, and Authentication Server
 - + A unique Ethertype value of 0x888e for “EAPOL”
 - + Several types of EAPOL payloads (EAPOL-Start, EAPOL-Packet, EAPOL-Key, etc)
- + In Wi-Fi, the full EAP framework is used only in Enterprise mode (802.1x)

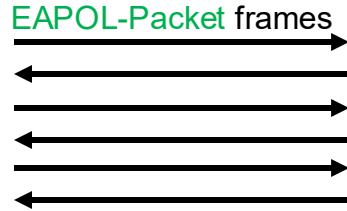
EAP Usage in WPA



The 4-Way Handshake

- + The 802.11i standard mandates the use of a 4-way EAPOL handshake between AP and Client *after the PMK has been established*
- + EAPOL Handshake is used for several purposes:
 - + To confirm mutual possession of the PMK
 - + To derive Transient Keys from the PMK
 - + To transmit Broadcast/Multicast encryption key from AP to Client
- + **EAPOL-Key** frames are used in this process

Review of RSN Keys & EAPOL Handshake



Derived from MSK
(Enterprise) or PSK (Personal)

EAPOL-Key Frames Required

Pairwise Master Key (PMK)

PTK (Pairwise
Transient Key)

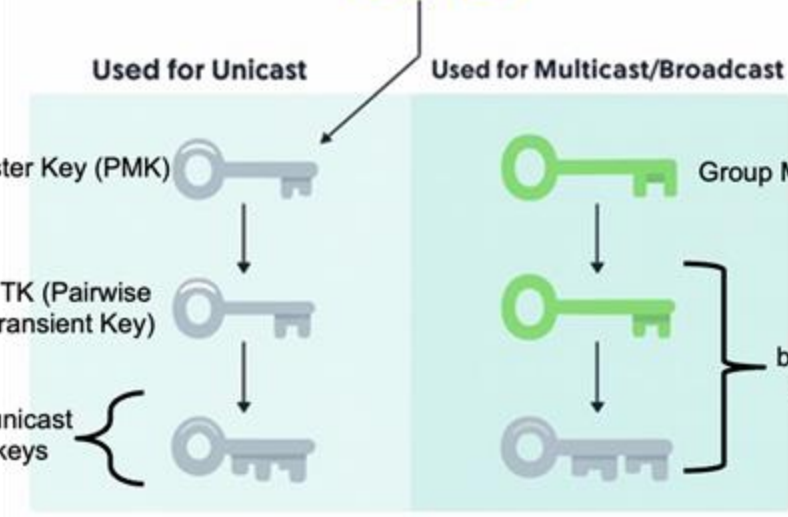
Additional unicast
encryption keys

Used for Unicast

Used for Multicast/Broadcast

Group Master Key (GMK)

Additional
broadcast/multicast
encryption keys



WPA EAPOL Handshake

304	2.091412	86:6c:b8:16:5e:89	TpLinkTechno_81:3...	802.11	157	Probe Request, SN=2075, FN=0, Flags=.....C, SSID="SkippyPip"
306	2.092083	TpLinkTechno_81:32:...	86:6c:b8:16:5e:89	802.11	468	Probe Response, SN=2990, FN=0, Flags=.....C, BI=100, SSID="SkippyPip"
308	2.092153	86:6c:b8:16:5e:89	TpLinkTechno_81:3...	802.11	70	Authentication, SN=2076, FN=0, Flags=.....C
310	2.093285	TpLinkTechno_81:32:...	86:6c:b8:16:5e:89	802.11	70	Authentication, SN=2991, FN=0, Flags=.....C
314	2.105668	86:6c:b8:16:5e:89	TpLinkTechno_81:3...	802.11	272	Association Request, SN=2077, FN=0, Flags=.....C, SSID="SkippyPip"
316	2.107910	TpLinkTechno_81:32:...	86:6c:b8:16:5e:89	802.11	248	Association Response, SN=2992, FN=0, Flags=.....C
320	2.119001	TpLinkTechno_81:32:...	86:6c:b8:16:5e:89	EAPOL	171	Key (Message 1 of 4)
330	2.136995	86:6c:b8:16:5e:89	TpLinkTechno_81:3...	EAPOL	193	Key (Message 2 of 4)
332	2.147976	TpLinkTechno_81:32:...	86:6c:b8:16:5e:89	EAPOL	275	Key (Message 3 of 4)
334	2.149832	86:6c:b8:16:5e:89	TpLinkTechno_81:3...	EAPOL	171	Key (Message 4 of 4)

> Tag: RM Enabled Capabilities (5 octets)	0010	12 0c 99 16 40 01 e7 a5	00 02 00 10 18 03 04 00
> Tag: HT Capabilities (802.11n D1.0)	0020	02 00 00 00 50 00 3c 00	86 6c b8 16 5e 89 50 d4
> Tag: HT Information (802.11n D1.0)	0030	f7 81 32 d8 50 d4 f7 81	32 d8 e0 ba 51 e4 10 64
> Tag: Overlapping BSS Scan Parameters	0040	2b 02 00 00 64 00 11 15	00 09 53 6b 69 70 70 79
> Tag: Extended Capabilities (8 octets)	0050	50 69 70 01 08 8c 12 98	24 b0 48 60 6c 03 01 9d
> Tag: VHT Capabilities	0060	46 05 73 d0 00 00 0c 2d	1a ef 09 03 ff ff ff 00
> Tag: VHT Operation	0070	00 00 00 00 00 00 00 00	01 00 00 00 00 00 00 00
> Tag: Tx Power Envelope	0080	00 00 00 3d 16 9d 05 04	00 00 00 00 00 00 00 00
> Tag: Vendor Specific: Microsoft Corp.: WMM/WME: Parameter Element	0090	00 00 00 00 00 00 00 00	00 00 00 4a 0e 14 00 0a
> Tag: Vendor Specific: Atheros Communications, Inc.: Advanced Capability	00a0	00 2c 01 c8 00 14 00 05	00 19 00 7f 08 05 00 0f
> Tag: Vendor Specific: Qualcomm Inc.	00b0	02 00 00 00 40 bf 0c f2	79 82 33 ea ff 00 00 ea
> Tag: Vendor Specific: Microsoft Corp.: WPA Information Element	00c0	ff 00 20 c0 05 01 9b 00	fc ff c3 05 03 3c 3c 3c
> Tag: RSN Information	00d0	3c dd 18 00 50 f2 02 01	01 80 00 03 a4 00 00 27
Tag Number: RSN Information (48)	00e0	a4 00 00 42 43 5e 00 62	32 2f 00 dd 09 00 03 7f
Tag length: 24	00f0	01 01 00 00 ff 7f dd 16	8c fd 04 00 00 49 4c
RSN Version: 1	0100	51 03 02 09 72 01 8c 16	00 00 46 00 00 00 dd 1a
> Group Cipher Suite: 00:0f:ac (Ieee 802.11) TKIP	0110	00 50 f2 01 01 00 00 50	f2 02 02 00 00 50 f2 04
Pairwise Cipher Suite Count: 2	0120	00 50 f2 02 01 00 00 50	f2 02 30 18 01 00 00 0f
> Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM) 00:0f:ac (Ieee 802.11) TKIP	0130	ac 02 02 00 00 0f ac 04	00 0f ac 02 01 00 00 0f
Auth Key Management (AKM) Suite Count: 1	0140	ac 02 00 00 dd 8b 00 50	f2 04 10 4a 00 01 10 10
> Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) PSK	0150	44 00 01 02 10 3b 00 01	03 10 47 00 10 87 65 43
Auth Key Management (AKM) Suite: 00:0f:ac (Ieee 802.11) PSK	0160	21 9a bc de f0 12 34 50	d4 f7 81 32 d9 10 21 00
Auth Key Management (AKM) OUI: 00:0f:ac (Ieee 802.11)	0170	07 54 50 2d 4c 69 6e 6b	10 23 00 09 41 72 63 68
Auth Key Management (AKM) type: PSK (2)	0180	65 72 5f 41 39 10 24 00	03 31 2e 30 10 42 00 0c
	0190	41 72 63 68 65 72 20 41	39 20 76 36 10 54 00 08
	01a0	00 06 00 50 f2 04 00 01	10 11 00 0a 41 72 63 68
	01b0	65 72 5f 41 39 10 24 00	03 31 2e 30 10 42 00 0c



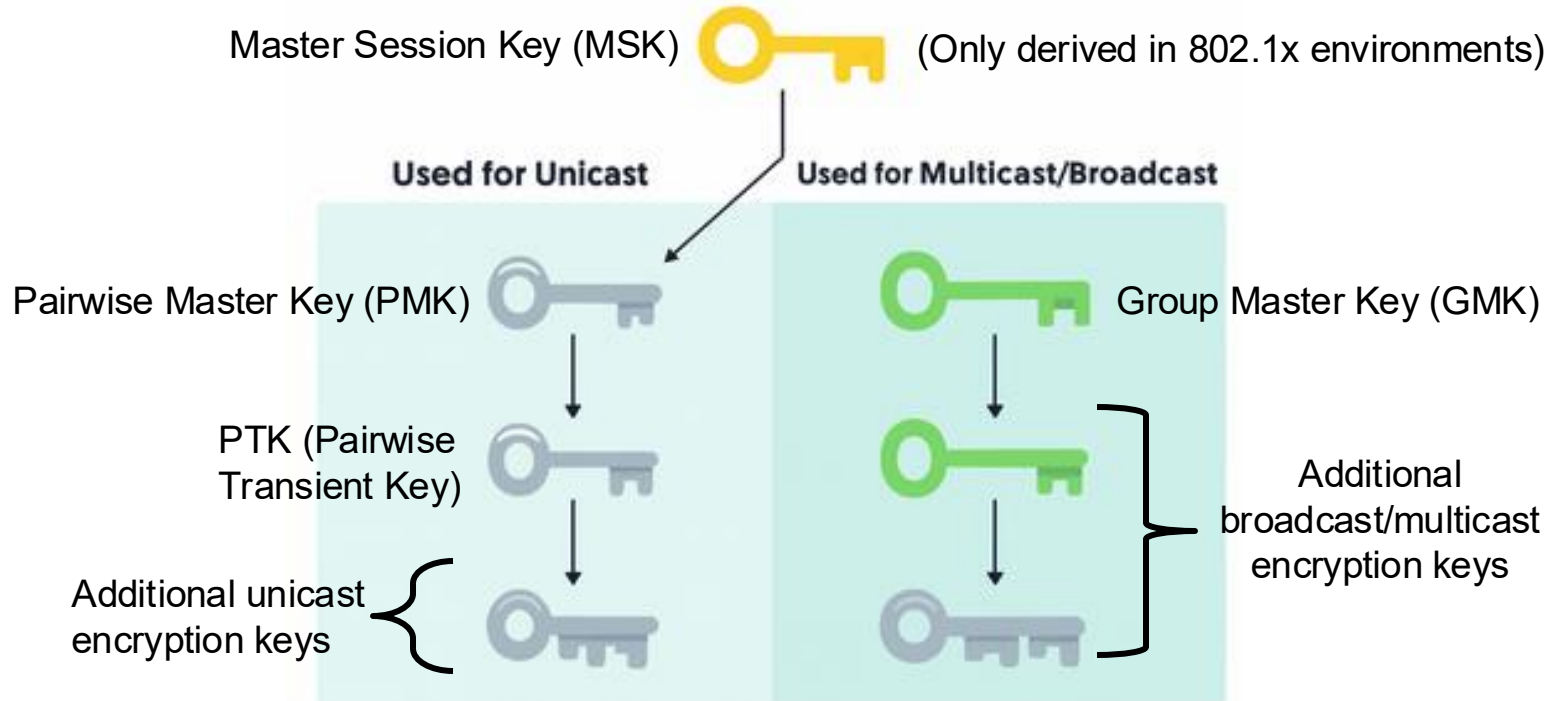
**Thank you for
watching!!**



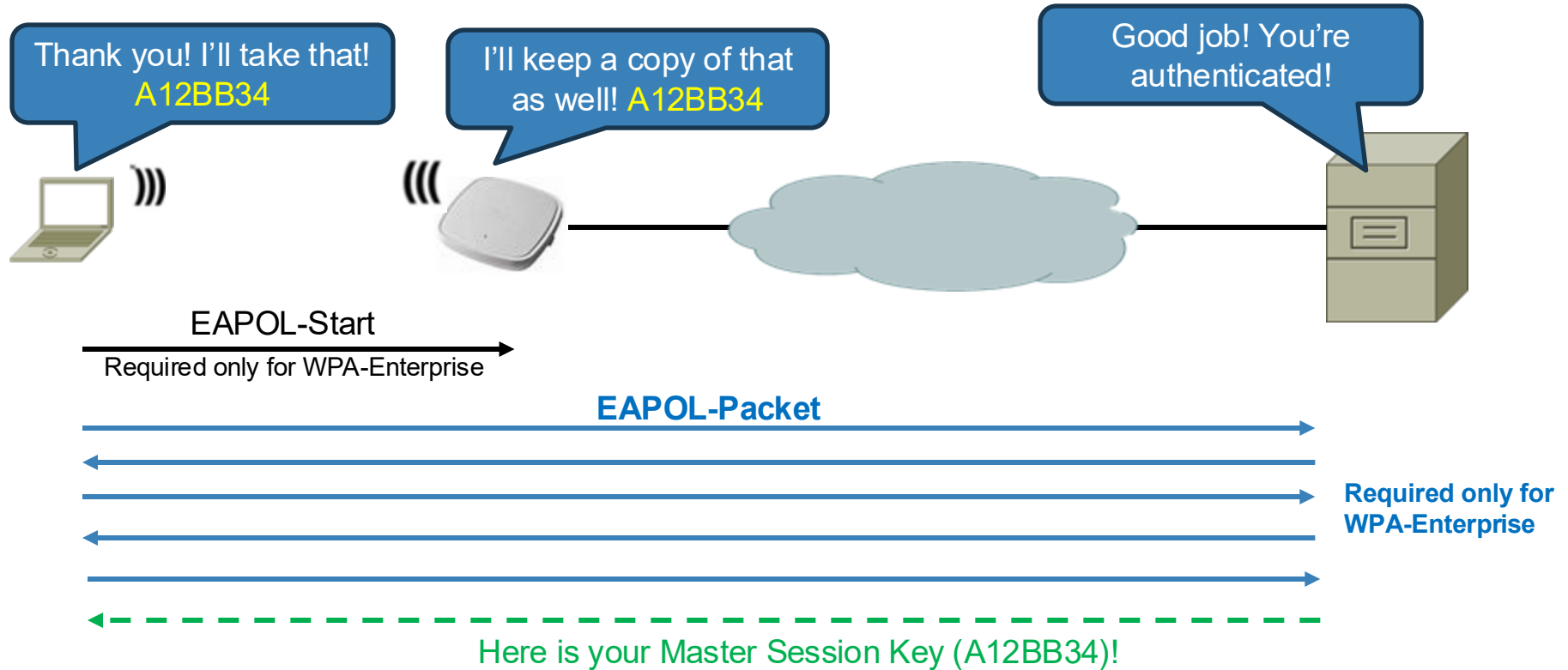
Breaking Down RSN Key Derivation

A Review of RSN Keys

- Once authentication has been completed, RSN dictates the generation and distribution of several types of keys



Propagating the MSK



Master Session Key

- + RFC 5247 specifies that a successful EAP authentication should result in the creation of a **Master Session Key**
 - + Derived from EAP authentication by Radius server
 - + 512 bits (64-bytes) in length
 - + Sent to the supplicant and WLC
- + WPA2/WPA3-Enterprise standards take that MSK and use it as the original keying material for creation of other Wi-Fi keys

Pairwise Master Key (PMK)

- + First defined by IEEE 802.11i specification
- + Three places it can originate from;
 - + Derived from MSK in Enterprise (first half of MSK)
 - + Equal to PSK in WPA2-Personal
 - + Derived from SAE handshake in WPA3-Personal
- + Used by client, and AP
- + Root for session key creation
- + PMK and MSK are VERY secret and thus kept hidden from any “show” commands

Pairwise Transient Key (PTK)

- + Once the AP and WLAN client obtain the PMK, a 4-way EAPOL-Key “handshake” commences.
- + During the handshake, the Pairwise Transient Key is derived.
- + PTK is considered a “Derived Key Vector”;
 - + **Derived** from materials exchanged during the handshake
 - + **Key vector** = an ordered bundle of subkeys
- + PTK itself is not used for any kind of encryption

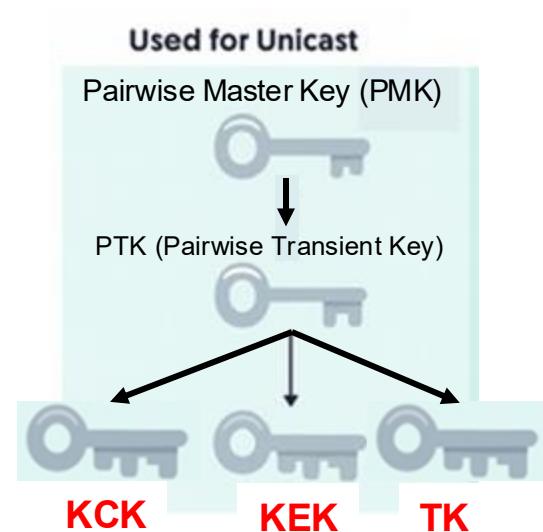
Used for Unicast

Pairwise Master Key (PMK)

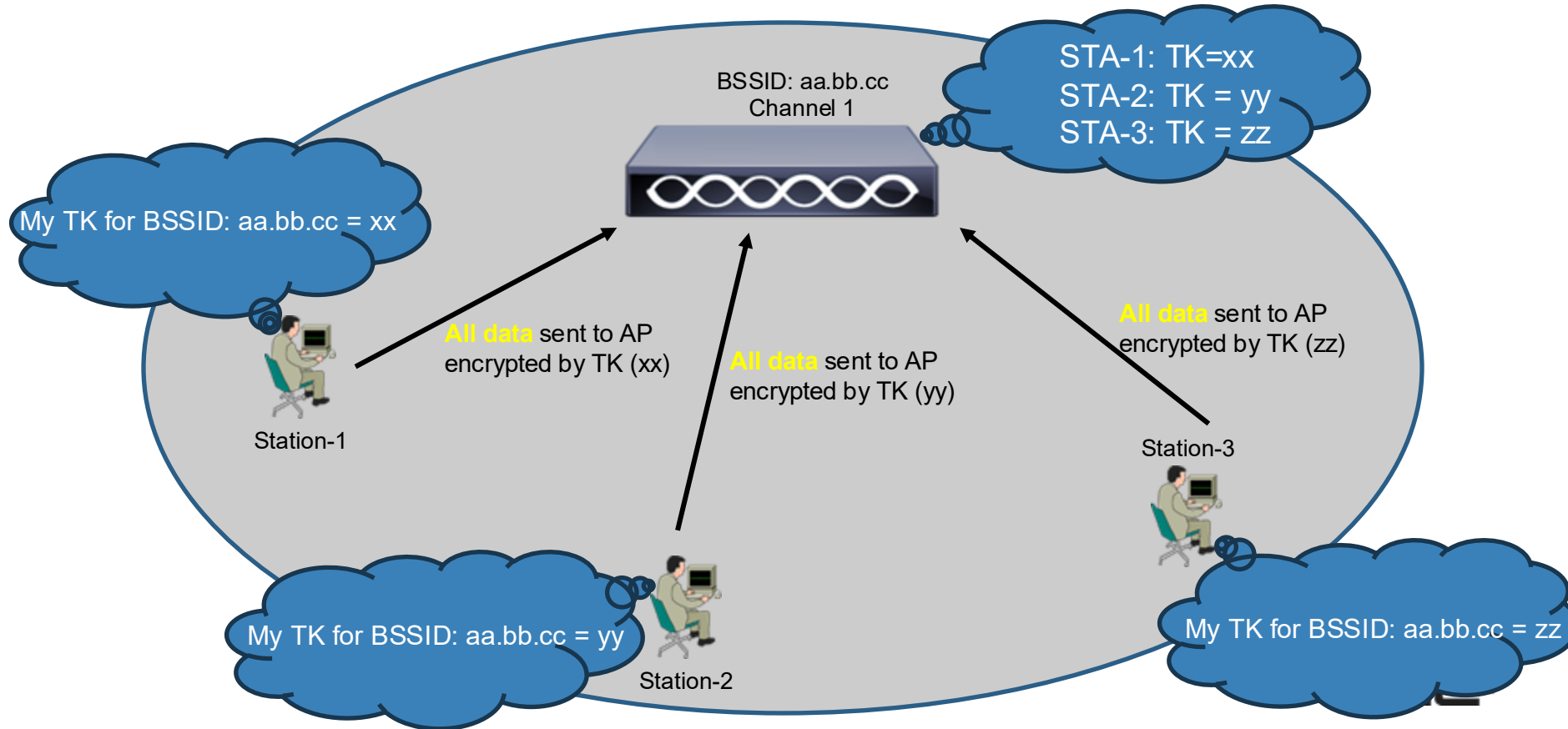


Keys Derived from PTK

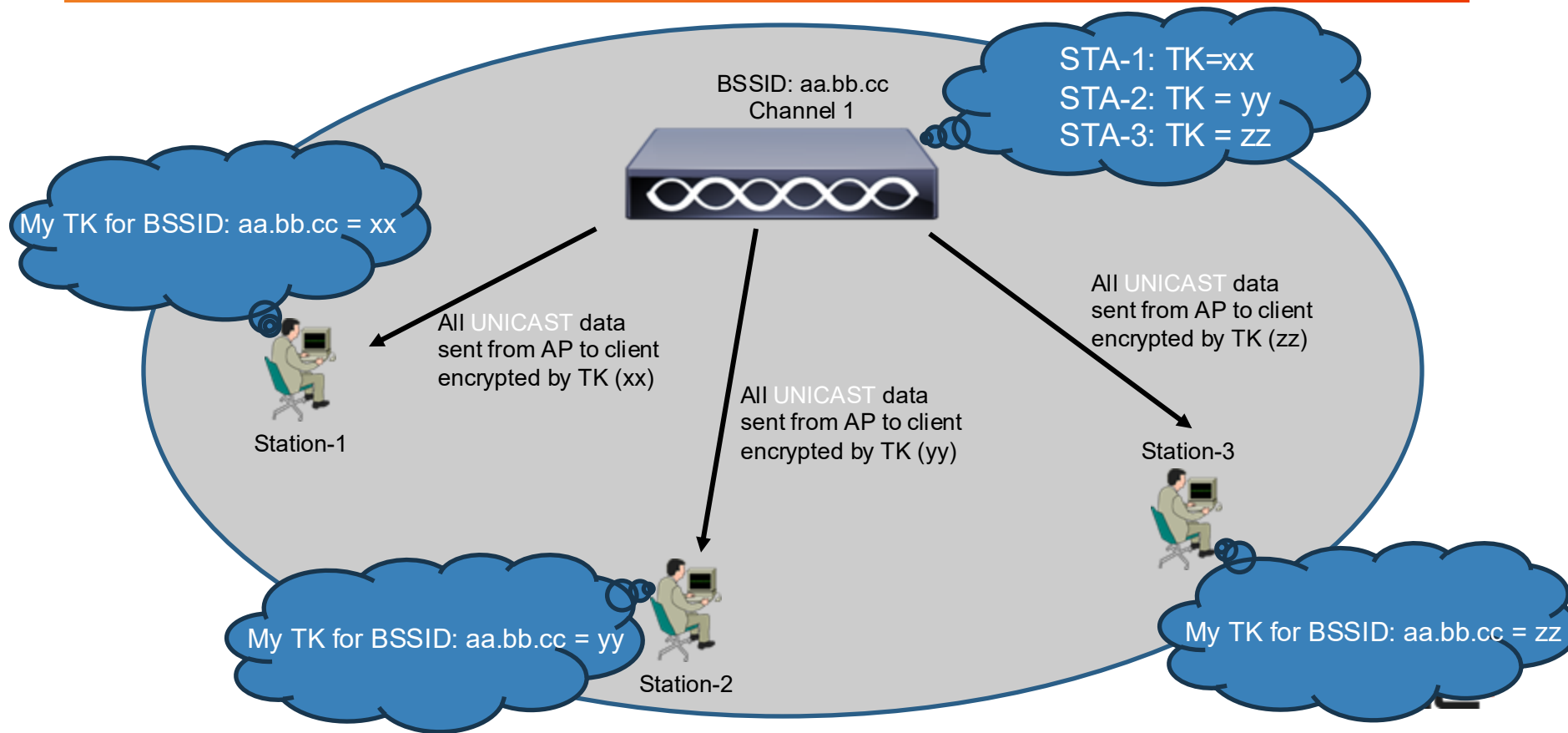
- + Once the PTK is derived, it is split into three equal-sized pieces
 - + KCK (Key Confirmation Key) – protects (MICs) the EAPOL-Key frames.
 - + KEK (Key Encryption Key) – wraps keys the AP sends (e.g., GTK/IGTK).
 - + **TK (Temporal Key)** – encrypts/decrypts unicast data frames.



Unicast Keys Recap



Unicast Keys Recap (2)



Broadcast & Multicast Keys

- + Access points may need to transmit broadcast or multicast frames into the BSS
 - + ARP Requests
 - + Multicast DNS (mDNS)
 - + Multicast Audio/Video
- + This type of traffic is also encrypted

GMK & GTK

- + Group Master Key (GMK)
- + Derived ONLY by the access point
 - + Strong 256-bit random number
 - + Typically, only computed once upon AP bootup
- + PRF (Pseudo Random Function) applied against GMK to derive **GTK (Group Temporal Key)**
- + GTK used to encrypt downstream (AP-> Clients) broadcast/multicast frames
- + GTK securely transmitted to clients by encrypting it with each client's KEK (Key Encryption Key)

Used for Multicast/Broadcast

Group Master Key (GMK)



Forwarding of the GMK & GTK

Source	Destination	Protocol	Length	Info
Cisco_75:9d:cf	b2:b7:4b:ab:29:65	EAPOL	193	Key (Message 1 of 4)
b2:b7:4b:ab:29:65	Cisco_75:9d:cf	EAPOL	202	Key (Message 2 of 4)
Cisco_75:9d:cf	b2:b7:4b:ab:29:65	EAPOL	299	Key (Message 3 of 4)
b2:b7:4b:ab:29:65	Cisco_75:9d:cf	EAPOL	171	Key (Message 4 of 4)

> Frame 152: Packet, 299 bytes on wire (2392 bits), 299 bytes captured (2392 bits)

> Radiotap Header v0, Length 34

> 802.11 radio information

> IEEE 802.11 QoS Data, Flags:F.C

▼ Logical-Link Control

> DSAP: SNAP (0xaa)

> SSAP: SNAP (0xaa)

> Control field: U, func=UI (0x03)

Organization Code: 00:00:00 (Officially Xerox, but 0:0:0:0:0:0 is more common)

Type: 802.1X Authentication (0x888e)

▼ 802.1X Authentication

Version: 802.1X-2004 (2)

Type: Key (3)

Length: 223

Key Descriptor Type: EAPOL RSN Key (2)

[Message number: 3]

> Key Information: 0x13c8

Key Length: 16

Replay Counter: 1

WPA Key Nonce: d0dd292433eaa924b151a1e0578d0cf6b07ba878ab65a550445215e93061c493

Key IV: 00000000000000000000000000000000

WPA Key RSC: 0000000000000000

WPA Key ID: 0000000000000000

WPA Key MIC: a4fc68d102efef4d5c05788068447048

WPA Key Data Length: 128

WPA Key Data [...]: 2f7e1c8a9881a694a1b746c9a1e941e1ac8cb2708f33e26014d675d8ee279fb6e04

GTK is
encrypted & then
transmitted



**Thank you for
watching!!**

WPA2 Personal PMK Derivation

Overview of PMK Derivation

- + In WPA2/3 Enterprise implementations, the Authentication Server derives the MSK
 - + First half of MSK is used by AP and Client as the PMK
- + *In WPA2/3 Personal, the Pre-Shared Key (PSK) is used as an input to the PMK derivation*
- + WPA2 Personal derives the PMK differently from WPA3 Personal
 - + WPA-2 Personal uses a PBKDF2 key derivation function
 - + WPA-3 Personal uses an SAE exchange function

WPA2 PMK Derivation

- + WPA2 Personal uses a method of PMK derivation called the, “PBKDF2 key derivation function”
 - + PBKDF2 = Password-Based Key Derivation Function 2
 - + It's defined in [PKCS #5 v2.0 / RFC 2898] and updated in [RFC 8018].
- + WPA2 allows users to create a PSK passphrase between 8-63 ASCII characters in length
- + Cryptography requires a fixed-length binary key (256 bits for the PMK).
 - + PBKDF2 bridges this gap by stretching the password into a full-strength key using repeated hashing.

How PBKDF2 Works

- + Start with the passphrase
- + Include a “salt” (SSID).
- + Run them through an HMAC-SHA1 function (a keyed hash) repeatedly 4096 times.
 - + Each iteration takes the result of the previous hash as input for the next one.
 - + Concatenate outputs until 256 bits are produced.
- + The result is the Pairwise Master Key (PMK).

Why Do We Care?

- + WPA2-Personal converts a passphrase into a 256-bit key via PBKDF2.
- + Because the iteration count is fixed (4096) and SHA-1 is relatively fast, weak passphrases are easily brute-forced with modern GPUs.
- + Short or dictionary-based PSKs are still risky, even though they “become 256 bits long” internally.
- + Summary: Short or easy to guess PSKs in WPA2 Personal can be easy to crack with offline attacks



**Thank you for
watching!!**

Introduction to SAE

(Simultaneous Authentication of Equals)

PMK Review

- + Secure Wi-Fi requires the use of several keys
- + Pairwise Master Key is a fundamental base key
- + Other critical encryption and integrity keys are derived from the PMK
- + Typically derived from either Pre-Shared Key (WPA Personal) or Master Session Key (from the Radius Server, WPA Enterprise)
- + Shared between WLAN Client and Access Point

From WPA2 to WPA3

+ In WPA2-Personal


- + *Pairwise Master Key (PMK) is derived directly from the Pre-Shared Key (PSK) using PBKDF2*
- + PBKDF2 is crackable when predictable PSKs are used.
- + Results in the same PMK for each WPA2 client

+ In WPA3-Personal

- + PBKDF2 mechanism is replaced with the Simultaneous Authentication of Equals (SAE) handshake
- + Also known as Dragonfly
- + Utilizes Diffie-Hellman-style key exchange to create unique PMKs per client.

WPA3 Authentication Frame Changes

- + WPA3 SAE changes the 802.11 Authentication frames (used by WPA and WPA2) in two significant ways:
 - + Changes the “authentication algorithm” to a value of “3” to indicate SAE



```
> 802.11 radio information
> IEEE 802.11 Authentication, Flags: .....C
v IEEE 802.11 Wireless Management
  v Fixed parameters (104 bytes)
    Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)
```

- + Exchanges four authentication messages instead of two

SAE Frame Exchange Overview

- + WPA3 authentication exchanges four frames in total;
 - + Two SAE “Commit” frames
 - + Two SAE “Confirm” frames
- + SAE Commit frames exchange data used for peers to derive a shared secret key
- + SAE Confirm frames verify that both sides have the same shared key.
- + Shared secret key is then mathematically changed into the PMK

SAE Message Exchanges

Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	493	Probe Response, SN=1337, FN=0, Flags=.....C, BI=100, SSID="CWAP-TEST"
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	194	Authentication, SN=1, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	194	Authentication, SN=0, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	130	Authentication, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	130	Authentication, SN=1, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	295	Association Request, SN=2335, FN=0, Flags=.....C, SSID="CWAP-TEST"
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	230	Association Response, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	EAPOL	221	Key (Message 1 of 4)
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	EAPOL	227	Key (Message 2 of 4)
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	EAPOL	295	Key (Message 3 of 4)
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	EAPOL	199	Key (Message 4 of 4)

Frame 122: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits) on interface \Device\NPF_{EE27CC7B-E19A-4000-8000-000000000000}	0000	18 db f2 34
Ethernet II, Src: Cisco_83:62:80 (1c:df:0f:83:62:80), Dst: Dell_34:92:f7 (18:db:f2:34:92:f7)	0010	00 b4 62 df
Internet Protocol Version 4, Src: 10.10.0.100, Dst: 10.181.33.228	0020	21 e4 15 b3
User Datagram Protocol, Src Port: 5555, Dst Port: 5000	0030	00 00 00 00
AiroPeek/OmniPeek encapsulated IEEE 802.11	0040	2c 00 0c d0
802.11 radio information	0050	f8 95 60 ae
IEEE 802.11 Authentication, Flags:C	0060	db 43 18 f8
IEEE 802.11 Wireless Management	0070	07 82 2d 53
Fixed parameters (104 bytes)	0080	a9 7a 13 54
Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)	0090	a5 d2 b4 ce
Authentication SEQ: 0x0001	00a0	31 65 97 1d
Status code: Successful (0x0000)	00b0	75 be f0 1c
SAE Message Type: Commit (1)	00c0	51 66
Group Id: 256-bit random ECP group (19)		
Scalar: be51db4318f8ee43264ab89e9741d42cc64007822d53f6754560abb68ab422d1		
Finite Field Element: 16c2a97a1354eb497d4e1ec2b1b739537c02a5d2b4cea679def9f9d35f110299e3933165971d730c6af317c		



**Thank you for
watching!!**

WPA3 SAE PMK Derivation

SAE Message Review

- + WPA2 PSK starts by exchanging two 802.11 “Authentication” messages
- + WPA3 SAE increased this to four 802.11 “Authentication” messages and changes the authentication algorithm from “Open” to “SAE”

Packet list:

No.	Time	Source	Destination	Protocol	Length	Info
70	0.000000	DexatekTechn_19:fa:...	TpLinkTechno_81:3...	802.11	70	Authentication, SN=25
70	0.000000	TpLinkTechno_81:32:...	DexatekTechn_19:f...	802.11	70	Authentication, SN=25
154	0.000000	DexatekTechn_19:fa:...	TpLinkTechno_81:3...	802.11	154	Association Request,
139	0.000000	TpLinkTechno_81:32:...	DexatekTechn_19:f...	802.11	139	Association Response,
139	0.000000	TpLinkTechno_81:32:...	DexatekTechn_19:f...	802.11	139	Association Response,

Frame 9: Packet, 70 bytes on wire (560 bits), 70 bytes captured (560 bits)

Radiotap Header v0, Length 36

802.11 radio information

IEEE 802.11 Authentication, Flags:C

IEEE 802.11 Wireless Management

Fixed parameters (6 bytes)

Authentication Algorithm: Open System (0)

Authentication SEQ: 0x0001

Status code: Successful (0x0000)

Authentication SEQ: 0x0001

Status code: Successful (0x0000)

SAE Message Type: Commit (1)

Group Id: 256-bit random ECP group (19)

Scalar: be51db4318f8ee43264ab89e9741d42cc64007822d53f6754560abb68ab422d1

Finite Field Element: 16c2a97a1354eb497d4e1ec2b1b739537c02a5d2b4cea679def9f9d35f110299e3933165971d730c6af317c

Packet bytes:

Offset	Hex	ASCII
0000	18 db f2 34	
0010	00 b4 62 df	
0020	21 e4 15 b3	
0030	00 00 00 00	
0040	2c 00 0c d0	
0050	f8 95 60 ae	
0060	db 43 18 f8	
0070	07 82 2d 53	
0080	a9 7a 13 54	
0090	a5 d2 b4 ce	
00a0	31 65 97 1d	
00b0	75 be f0 1c	
00c0	51 66	

Shared Secret Generation

- + WPA3 SAE improved upon WPA2 PSK by changing to a process like Diffie-Hellman for key generation.
- + Uses some known public information, mathematically computed with huge random numbers and prime numbers to compute the PMK
- + Changes the random numbers on each successive authentication attempt
- + Derives a new, unique shared secret (and PMK) for each connection.

A Review of Modular Arithmetic

- + Sometimes you want to create a function that;
 - + Divides numbers by a certain value “x” (“4” in the examples below)
 - + Looks for the remainder of that division “y”
 - + Uses that remainder within a formula or algorithm

$$25 / 4 = 6 \text{ (remainder 1)} \quad 34 / 4 = 8 \text{ (remainder 2)} \quad 31 / 4 = 7 \text{ (remainder 3)}$$

This is what I'm after!



- + In this case, your values (“25”, “34”, “31”) applied against “modulo 4” (what you’re dividing against) would yield the remainders you’re looking for.
- + Expressed mathematically as $25 \% 4 = 1$

A Review of Classic Diffie-Hellman

Peer-A

Peer-B

Try It Yourself!

Try this yourself with the following values:

Generator = 5

Modulus = 23

Peer-A Exponent = 7 Peer-B Exponent = 11

Peer-A

Generator/Base = g

- 1 Randomly selected exponent = a
Modulus (static/well-known) = m

Initial computation

$$5^7 \% 23 = 17$$

- 2 $g^a \text{ Mod } m = A$

My results (public key) = A

My results (public key) = B

Shared Secret (S) computation

- 4 $B^a \text{ Mod } m = S$

$$22^7 \% 23 = 22$$

Peer-B

Generator/Base = g

- 1 Randomly selected exponent = b
Modulus (static/well-known) = m

Initial computation

$$5^{11} \% 23 = 22$$

- 2 $g^b \text{ Mod } m = B$

Shared Secret computation

- 4 $A^b \text{ Mod } m = S$

$$17^{11} \% 23 = 22$$

SAE Scalar & Finite Field Elements

Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	493	Probe Response, SN=1337, FN=0, Flags=.....C, BI=100, SSID="CWAP-TEST"
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	194	Authentication, SN=1, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	194	Authentication, SN=0, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	130	Authentication, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	130	Authentication, SN=1, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	295	Association Request, SN=2335, FN=0, Flags=.....C, SSID="CWAP-TEST"
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	230	Association Response, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	EAPOL	221	Key (Message 1 of 4)
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	EAPOL	227	Key (Message 2 of 4)
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	EAPOL	295	Key (Message 3 of 4)
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	EAPOL	199	Key (Message 4 of 4)

Frame 122: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits) on interface \Device\NPF_{EE27CC7B-E19A-403D-B7E1-0800200C9A66}	0000	18 db f2 34
Ethernet II, Src: Cisco_83:62:80 (1c:df:0f:83:62:80), Dst: Dell_34:92:f7 (18:db:f2:34:92:f7)	0010	00 b4 62 df
Internet Protocol Version 4, Src: 10.10.0.100, Dst: 10.181.33.228	0020	21 e4 15 b3
User Datagram Protocol, Src Port: 5555, Dst Port: 5000	0030	00 00 00 00
AiroPeek/OmniPeek encapsulated IEEE 802.11	0040	2c 00 0c d0
802.11 radio information	0050	f8 95 60 ae
IEEE 802.11 Authentication, Flags:C	0060	db 43 18 f8
IEEE 802.11 Wireless Management	0070	07 82 2d 53
Fixed parameters (104 bytes)	0080	a9 7a 13 54
Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)	0090	a5 d2 b4 ce
Authentication SEQ: 0x0001	00a0	31 65 97 1d
Status code: Successful (0x0000)	00b0	75 be f0 1c
SAE Message Type: Commit (1)	00c0	51 66
Group Id: 256-bit random ECP group (19)		
Scalar: be51db4318f8ee43264ab89e9741d42cc64007822d53f6754560abb68ab422d1		
Finite Field Element: 16c2a97a1354eb497d4e1ec2b1b739537c02a5d2b4cea679def9f9d35f110299e3933165971d730c6af317c		

SAE Commit Phase

WPA3 Client

WPA3 Access Point

SAE utilizes Elliptic Curve Diffie-Hellman (ECDH)

SAE Commit Phase

5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	194	Authentication, SN=1, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	194	Authentication, SN=0, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	130	Authentication, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	130	Authentication, SN=1, FN=0, Flags=.....C

Commit messages

```
> IEEE 802.11 Authentication, Flags: .....C
< IEEE 802.11 Wireless Management
  < Fixed parameters (104 bytes)
    Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)
    Authentication SEQ: 0x0001
    Status code: Successful (0x0000)
    SAE Message Type: Commit (1)
    Group Id: 256-bit random ECP group (19)
    Scalar: be51db4318f8ee43264ab89e9741d42cc64007822d53f6754560abb68ab422d1
    Finite Field Element: 16c2a97a1354eb497d4e1ec2b1b739537c02a5d2b4cea679def9f9d35f110299e3933165971d730c6af317c
```

```
> IEEE 802.11 Authentication, Flags: .....C
< IEEE 802.11 Wireless Management
  < Fixed parameters (104 bytes)
    Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)
    Authentication SEQ: 0x0001
    Status code: Successful (0x0000)
    SAE Message Type: Commit (1)
    Group Id: 256-bit random ECP group (19)
    Scalar: ef7ff1f05a4ae8e908eb678522d9752a37b0452c967f572d95f4513dc6d65489
    Finite Field Element: 4408afbda3d873ee1fa4db3abcac954fafa721e639797769919b97213f5b866c958f9e7887ed65ffcfabab2
```

SAE Confirm Messages

- + Confirm messages prove each side has correctly computed the same shared secret.
- + Confirm messages contain a **confirm token** derived from an HMAC hash digest value that takes as inputs;
 - + Both peer's "Scalar" and "Element" values
 - + The shared secret that was derived
 - + A “confirm counter” value (client = “1”, access point = “2”)
- + The process to confirm the token always produces the same output value given the same input values
- + **PMK can be derived from shared secret after Confirm messages have been exchanged**

SAE Confirm Phase

5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	194 Authentication, SN=1, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	194 Authentication, SN=0, FN=0, Flags=.....C
5e:a7:ec:a8:33:ab	Cisco_95:60:ae	802.11	130 Authentication, SN=2, FN=0, Flags=.....C
Cisco_95:60:ae	5e:a7:ec:a8:33:ab	802.11	130 Authentication, SN=1, FN=0, Flags=.....C

Confirm messages

```
> IEEE 802.11 Authentication, Flags: .....C
< IEEE 802.11 Wireless Management
  < Fixed parameters (40 bytes)
    Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)
    Authentication SEQ: 0x0002
    Status code: Successful (0x0000)
    SAE Message Type: Confirm (2)
    Send-Confirm: 0
    Confirm: caf92b6e1ff5e243d288940bd195171387e524e68f709e76c351a61df4258c2
```

```
> IEEE 802.11 Authentication, Flags: .....C
< IEEE 802.11 Wireless Management
  < Fixed parameters (40 bytes)
    Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)
    Authentication SEQ: 0x0002
    Status code: Successful (0x0000)
    SAE Message Type: Confirm (2)
    Send-Confirm: 0
    Confirm: a55e1aff4a9cc75783dd6eaa618f4e8bd36bfd32f2f8a694b71cc2770225e1cf
```

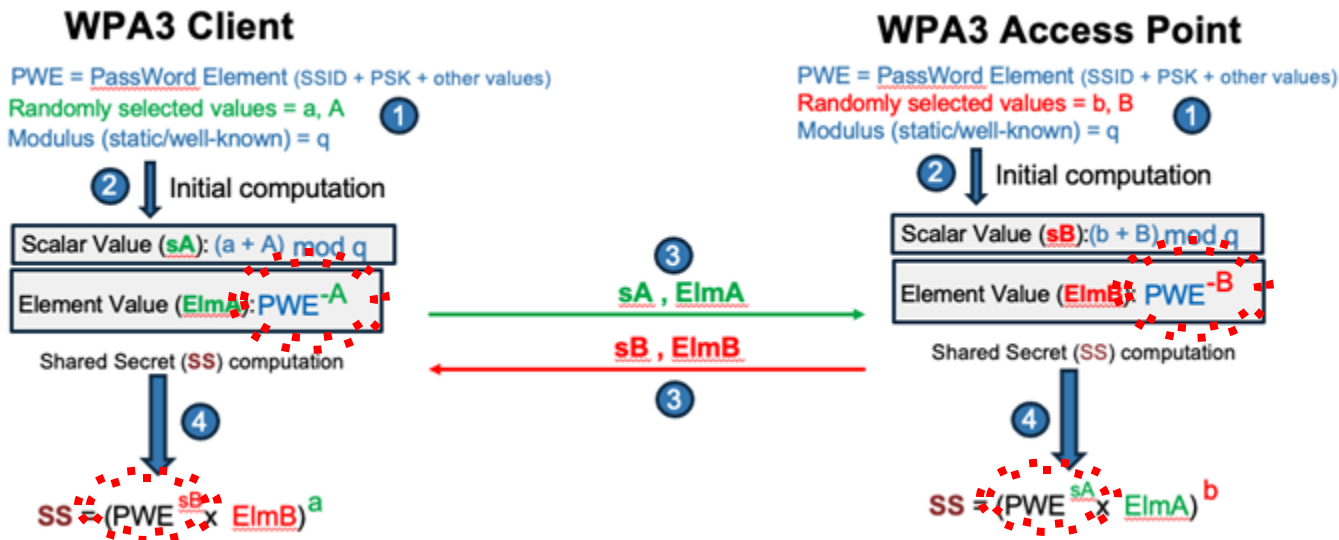



**Thank you for
watching!!**

WPA-3 Password Element Generation

PWE Introduction

- + WPA3 SAE requires the computation of a Password Element (PWE)
 - + PWE is used as one input within math functions to create a shared secret

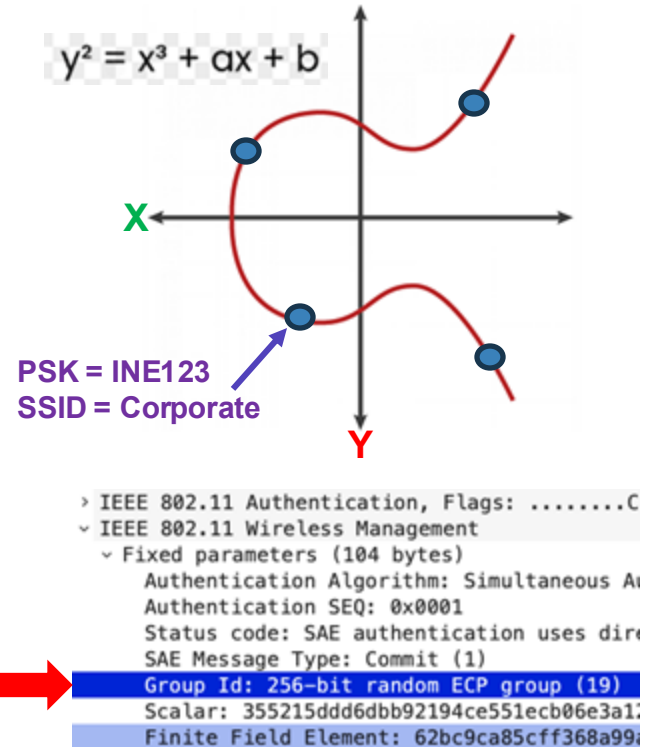


PWE Introduction

- + PWE uses various inputs such as SSID, Client MAC and AP MAC in its derivation
- + Each client connection derives a unique PWE value
- + PWE is a value mapped to a point on an elliptic curve
- + Technically called a “*mapped group element derived from the SSID + passphrase*”

PWE & Elliptic Curves

- + WLAN client will decide on a specific Elliptic Curve Group it wishes to use
 - + Client and AP use the same ECDH Group and thus the same curved shape
- + Every point on an elliptic curve matches a mathematical formula
- + Client maps PSK+SSID (and other things) to a point on the Elliptic Curve
- + First “Commit” message from WLAN client indicates specific Elliptic Curve group it wishes to use.



Why Do We Care?

- + Two (configurable) methods exist to map PSK+SSID to a point on an elliptic curve (called an “element”)
 - + Hash to Element (H2E)
 - + Hunting and Pecking (HnP)
- + Client and Access Point MUST use the same method

H2E or HNP: Cisco Catalyst 9800

Edit WLAN

General **Security** Advanced Add To Policy Tags

Layer2 Layer3 AAA

☐ WPA + WPA2 ☐ WPA2 + WPA3 ☒ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
		Transition Disable	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF Required ☐

Association Comeback Timer* 1

SA Query Time* 200

Fast Transition

Status Disabled

Over the DS ☐

Reassociation Timeout* 20

Auth Key Mgmt

SAE	<input checked="" type="checkbox"/>	FT + SAE	<input type="checkbox"/>
OWE	<input type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>		

Anti Clogging Threshold* 1500

Max Retries* 5

Retransmit Timeout* 400

PSK Format ASCII

PSK Type Unencrypted

Pre-Shared Key* *****

SAE Password Element Hash to Elem...

Hash to Elem...

Both H2E and HnP

Hash to Element Only

Hunting and Pecking Only



**Thank you for
watching!!**



HnP and H2E

H2E or HNP: Cisco Catalyst 9800

Edit WLAN

General **Security** Advanced Add To Policy Tags

Layer2 Layer3 AAA

☐ WPA + WPA2 ☐ WPA2 + WPA3 ☒ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
		Transition Disable	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF Required ☐

Association Comeback Timer* 1

SA Query Time* 200

Fast Transition

Status Disabled

Over the DS ☐

Reassociation Timeout* 20

Auth Key Mgmt

SAE	<input checked="" type="checkbox"/>	FT + SAE	<input type="checkbox"/>
OWE	<input type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>		

Anti Clogging Threshold* 1500

Max Retries* 5

Retransmit Timeout* 400

PSK Format ASCII

PSK Type Unencrypted

Pre-Shared Key* *****

SAE Password Element Hash to Elem...

Hash to Elem...

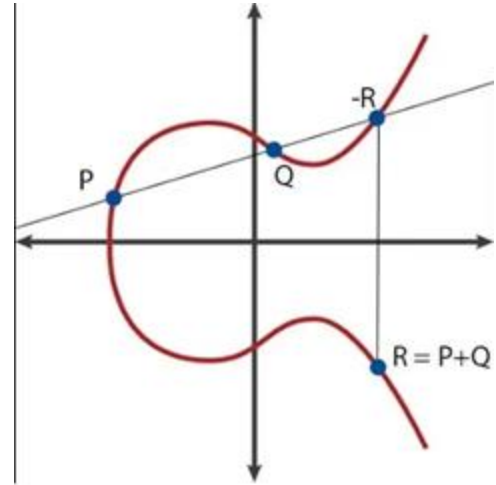
Both H2E and HnP

Hash to Element Only

Hunting and Pecking Only

What is an Elliptic Curve?

- + Graphical curves are made up of an infinite quantity of points
- + Each point on a curve has a value for the X-axis (horizontal) and Y-axis (vertical)
- + Although Elliptic Curves are often displayed as two-dimensional curved shapes, they are simply mathematical equations in which the X and Y coordinates must adhere to a specific equation.



Elliptic Curve Math

- + Elliptic Curve Diffie-Hellman (ECDH) is the basis for the key agreement in WPA3-SAE
- + The equation used is dictated by the Elliptic Curve Group that is selected:

The NIST P-256 Curve (Used by WPA3...“Group 19”)

$$y^2 = x^3 + ax + b \pmod{p}$$

- + For two values (representing “X” and “Y”) to be considered valid Elliptic curve points they must adhere to the formula.

The NIST P-256 Curve (Used by WPA3...“Group 19”)

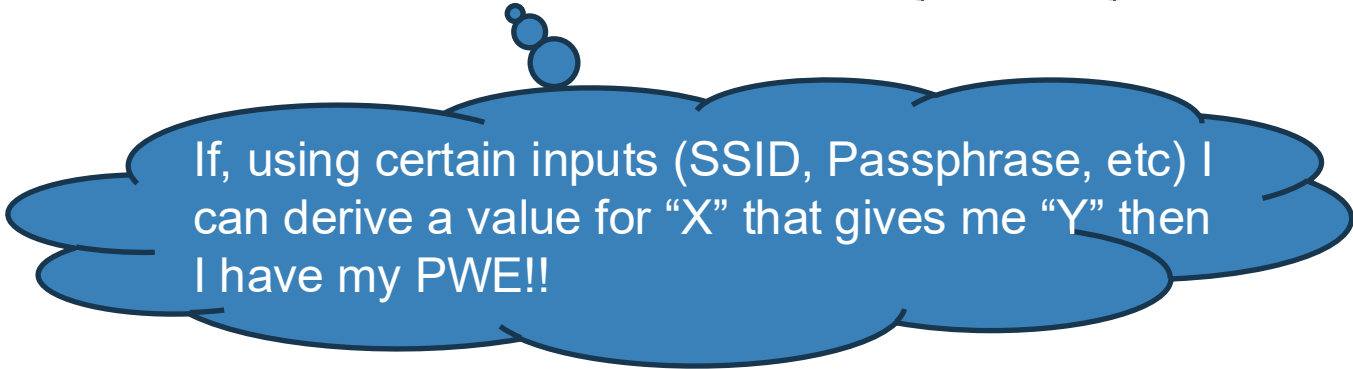
Symbol	Meaning
x, y	Coordinates of a point on the curve (these are variables you solve for).
a, b	Fixed curve parameters that define the specific shape of the elliptic curve.
p	A large prime modulus , meaning all arithmetic is done modulo p .



Mapping to the Element

- + WPA3 SAE must first derive a PWE (PassWord Element) value before performing SAE exchange
- + The PWE must be a valid point on the agreed-upon Elliptic Curve

$$y^2 = x^3 + ax + b \pmod{p}$$



If, using certain inputs (SSID, Passphrase, etc) I can derive a value for “X” that gives me “Y” then I have my PWE!!

A Simplistic Example

Imagine if the Elliptic Curve Formula was: $Y^2 = X - 1$

SSID = "ACE"
Passphrase = "B55"



ASCII values

"ACE" = 65, 67, and 69 (added together = 201)
"B55" = 66, 53, and 53 (added together = 172)
[201 + 172 = 373]



SSID = "ACE"
Passphrase = "B55"

How do I change "373" into a suitable value for "X" that will satisfy the equation? Let's try hashing it!

Attempt-1: Hash(373) = 501

Attempt-2: Hash(501) = 207

Attempt-3: Hash(207) = 145

PWE = 145:12

X = 145, Y = 12

Mapping to the Element

- + WPA3 SAE must first derive a PWE (PassWord Element) value before performing SAE exchange
- + This is done by finding a common “element” value shared between Access Point and WLAN clients
- + The element can be derived by one of two methods:
 - + Hash to Element (H2E)
 - + Hunting and Pecking (HnP)
- + When configurable, client and AP must use the same method

H2E or HNP: Cisco Catalyst 9800

Edit WLAN

General **Security** Advanced Add To Policy Tags

Layer2 Layer3 AAA

☐ WPA + WPA2 ☐ WPA2 + WPA3 ☒ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
		Transition Disable	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF Required ☐

Association Comeback Timer* 1

SA Query Time* 200

Fast Transition

Status Disabled

Over the DS ☐

Reassociation Timeout* 20

Auth Key Mgmt

SAE	<input checked="" type="checkbox"/>	FT + SAE	<input type="checkbox"/>
OWE	<input type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>		

Anti Clogging Threshold* 1500

Max Retries* 5

Retransmit Timeout* 400

PSK Format ASCII

PSK Type Unencrypted

Pre-Shared Key* *****

SAE Password Element Hash to Elem...

Hash to Elem...

Both H2E and HnP

Hash to Element Only

Hunting and Pecking Only

Hunting & Pecking

- + Historically, HnP came out first
 - + Uses a hashing method to convert passphrase (and additional elements) to a point on the curve.
 - + If the resulting candidate is not a valid point (or doesn't satisfy certain mathematical properties), it “pecks” (increments a counter) and tries again.
 - + Iterates until it finds a candidate that works.
- + Potentially many iterations & weaker security

Hash To Element

- + Newer, came out after HnP
- + Uses a cryptographically specified hash-to-curve (or hash-to-group) algorithm
- + Maps the input values directly to a valid group element *in a way that doesn't require trial-and-error loops.*
- + More efficient and safer than HnP
- + Some older clients and APs don't support it



PWE Algorithm Advertisement

- + Access Points advertise which PWE capability they support in Beacons and Probe responses
- + If a WLAN supports H2E and clients do as well, clients will prefer H2E
- + H2E is mandatory for WiFi 6e and WiFi 7
- + *If unsure about client capability, select HnP for WiFi 6 (and lower) WLANs*



**Thank you for
watching!!**

Potential Problems with H2E

Mapping to the Element

- + WPA3 SAE must first derive a PWE (PassWord Element) value before performing SAE exchange
- + This is done by finding a common “element” value shared between Access Point and WLAN clients
- + The element can be derived by one of two methods:
 - + Hash to Element (H2E)
 - + Hunting and Pecking (HnP)
- + H2E is newer and more secure than HnP
- + If WLAN advertises H2E capability and clients support it, they will prefer H2E

H2E “Gotcha”

- + Some older Cisco access points support Hunting and Pecking (for PWE generation) **but not Hash to Element (H2E)**
- + Cisco 9800 WLC will still allow selection of H2E and will not warn of AP incompatibility
- + H2E capability must be advertised in Beacons, but older APs don't support this.
- + Then what happens?

H2E Information Elements

- + This is what you SHOULD see in Beacons when using H2E

```

  ✓ IEEE 802.11 Wireless Management
    ✓ Fixed parameters (12 bytes)
      Timestamp: 319795211
      Beacon Interval: 0.102400 [Seconds]
      > Capabilities Information: 0x1111
    ✓ Tagged parameters (299 bytes)
      > Tag: SSID parameter set: "WPA3-Test"
      > Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
      > Tag: DS Parameter set: Current Channel: 36
      > Tag: Traffic Indication Map (TIM): DTIM 0 of 1 bitmap
      > Tag: Country Information: Country Code US, Environment All
      > Tag: Power Constraint: 0
      > Tag: RSN Information
    ✓ Tag: RSN eXtension (1 octet)
      Tag Number: RSN eXtension (244)
      Tag length: 1
      ✓ RSNX: 0x20 (octet 1)
        .... 0000 = RSNX Length: 0
        ...0 .... = Protected TWT Operations Support: False
        ..1. .... = SAE Hash to element: True
        .0.. .... = SAE-PK: False
        0... .... = Protected WUR Frame Support: False

```

Both H2E & HnP (Sniffer)

Source	Destination	Protocol	Length	Info
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	168	Authentication, SN=2390, FN=0, Flags=.....C
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	168	Authentication, SN=153, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	104	Authentication, SN=2391, FN=0, Flags=.....C
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	104	Authentication, SN=154, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	285	Association Request, SN=2392, FN=0, Flags=.....C, SSID="WPA3-Test"
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	204	Association Response, SN=155, FN=0, Flags=.....C
Cisco_89:09:ef	b2:b7:4b:ab:29:65	EAPOL	193	Key (Message 1 of 4)
b2:b7:4b:ab:29:65	Cisco_89:09:ef	EAPOL	202	Key (Message 2 of 4)
Cisco_89:09:ef	b2:b7:4b:ab:29:65	EAPOL	267	Key (Message 3 of 4)
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	66	Disassociate, SN=2393, FN=0, Flags=.....C

> Frame 1181: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)	0000
> Radiotap Header v0, Length 36	0010
> 802.11 radio information	0020
> IEEE 802.11 Disassociate, Flags:C	0030
> IEEE 802.11 Wireless Management	0040
< Fixed parameters (2 bytes)	
Reason code: Element in 4-way handshake different from (Re)Association Request/Probe Response/Beacon frame (0x0011)	

This is what you'll see when "**Both H2E and HnP**" are selected in the Controller for the WLAN.

Both H2E & HnP (Radioactive Trace)

```
: [client-keymgmt] [14472]: (info): MAC: b2b7.4bab.2965 EAP key M1 Sent successfully  
: [client-keymgmt] [14472]: (info): MAC: b2b7.4bab.2965 Client key-mgmt state transition: S_INITPMK -> S_PTK_START  
}: [mm-dgram-io] [15658]: (debug): MAC: 0000.0000.0000 Sending message: pmk_update to group: default  
: [client-keymgmt] [14472]: (info): MAC: b2b7.4bab.2965 M2 Status: EAP key M2 validation success  
: [client-keymgmt] [14472]: (info): MAC: b2b7.4bab.2965 EAP key M3 Sent successfully  
: [client-keymgmt] [14472]: (info): MAC: b2b7.4bab.2965 Client key-mgmt state transition: S_PTK_START -> S_PTKINITNEGOTIATING  
: [dot11] [14472]: (info): MAC: b2b7.4bab.2965 DOT11 state transition: S_DOT11_ASSOCIATED -> S_DOT11_TO_DELETE  
: [client-orch-sm] [14472]: (info): MAC: b2b7.4bab.2965 Deleting the client, reason: 69, CO_CLIENT_DELETE_REASON_CLIENT_EAP_TIMEOUT_FAILURE, Client state S_CO_L2_AUTH_IN_PROGRESS  
: [client-orch-sm] [14472]: (note): MAC: b2b7.4bab.2965 Client delete initiated. Reason: CO_CLIENT_DELETE_REASON_CLIENT_EAP_TIMEOUT_FAILURE, details: , fsm-state transition 00|00|00|00|00|00|00|00|00|00|01|07|15|a2]
```

This is what you'll see when “**Both H2E and HnP**” are selected in the Controller for the WLAN.



H2E Only (Sniffer)

Source	Destination	Protocol	Length	Info
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	189	Probe Request, SN=2571, FN=0, Flags=.....C, SSID="WPA3-Test"
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	368	Probe Response, SN=729, FN=0, Flags=.....C, BI=100, SSID="WPA3-Test"
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	168	Authentication, SN=2572, FN=0, Flags=.....C
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	70	Authentication, SN=731, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	66	Deauthentication, SN=2573, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	189	Probe Request, SN=2584, FN=0, Flags=.....C, SSID="WPA3-Test"
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	368	Probe Response, SN=742, FN=0, Flags=.....C, BI=100, SSID="WPA3-Test"
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	168	Authentication, SN=2585, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	168	Authentication, SN=2586, FN=0, Flags=.....C
Cisco_89:09:ef	b2:b7:4b:ab:29:65	802.11	70	Authentication, SN=743, FN=0, Flags=.....C
b2:b7:4b:ab:29:65	Cisco_89:09:ef	802.11	66	Deauthentication, SN=2587, FN=0, Flags=.....C

> Frame 773: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)	0000
> Radiotap Header v0, Length 36	0010
> 802.11 radio information	0020
> IEEE 802.11 Authentication, Flags:C	0030
> IEEE 802.11 Wireless Management	0040
< Fixed parameters (6 bytes)	
Authentication Algorithm: Simultaneous Authentication of Equals (SAE) (3)	
Authentication SEQ: 0x0001	
Status code: Unspecified failure (0x0001)	
SAE Message Type: Commit (1)	

This is what you'll see when "**H2E Only**" is selected in the Controller for the WLAN.

H2E Only (Radioactive Trace)

```
[dot11][14472]: (info): MAC: b2b7.4bab.2965 Dot11 SAE AUTH COMMIT message received in NOTHING State  
[dot11][14472]: (ERR): MAC: b2b7.4bab.2965 Parsing SAE COMMIT Message- PWE mismatch, expected Hash to Element status code (0)  
[dot11][14472]: (ERR): MAC: b2b7.4bab.2965 Dot11 send SAE auth COMMIT message with failure status code  
[dot11][14472]: (info): MAC: b2b7.4bab.2965 Decrement sae open session counter  
[dot11][14472]: (info): MAC: b2b7.4bab.2965 Dot11 process SAE Commit Failure Handling  
[dot11][14472]: (info): MAC: b2b7.4bab.2965 Dot11 SAE AUTH COMMIT message received in NOTHING State  
[dot11][14472]: (ERR): MAC: b2b7.4bab.2965 Parsing SAE COMMIT Message- PWE mismatch, expected Hash to Element status code (0)  
[dot11][14472]: (ERR): MAC: b2b7.4bab.2965 Dot11 send SAE auth COMMIT message with failure status code
```

This is what you'll see when “**H2E Only**” is selected in the Controller for the WLAN.

How to Avoid These H2E “Gotchas”

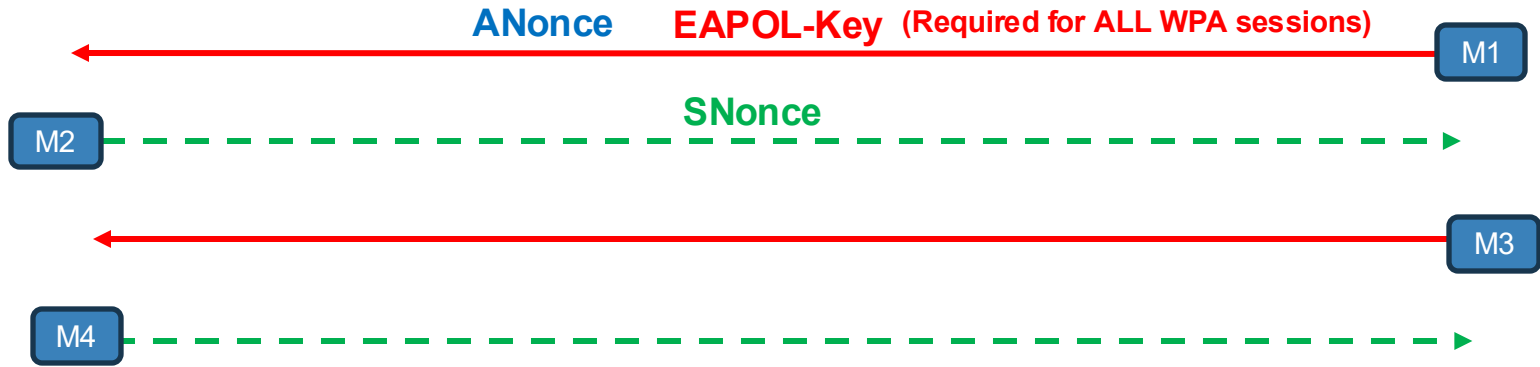
- + H2E is required for Wi-Fi 6e and Wi-Fi 7
- + If you purchase an access point that supports these Wi-Fi Alliance certifications, it will support H2E
- + Other access points that support Wi-Fi 6 (or lower) may NOT support H2E.
- + If NONE of your AP's support WiFi-6e configure WLANs with HnP



**Thank you for
watching!!**

WPA Transient & Temporal Key Generation

What's After the PMK?



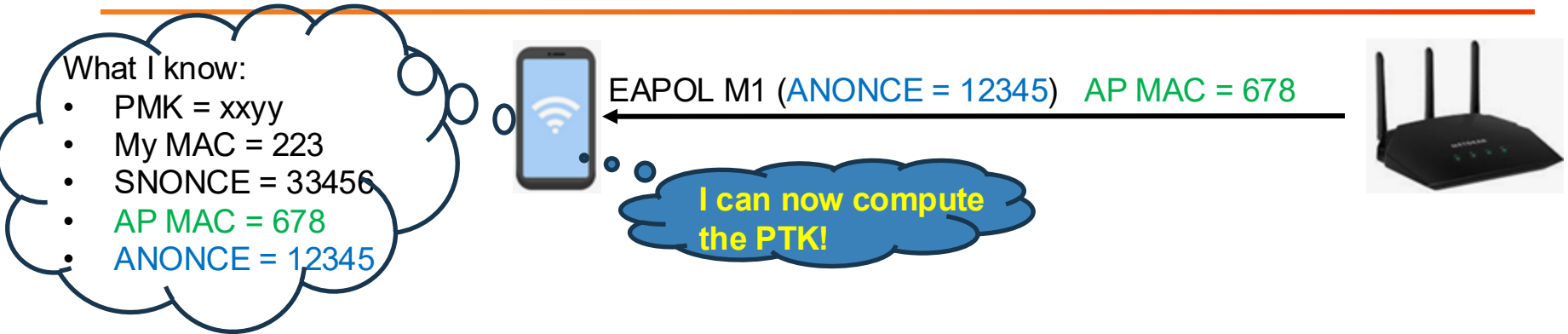
Nonces & Security

- + Nonces are exchanged in the first two EAPOL-Key messages
- + Fresh nonces are created whenever a client associates (and reassociates) with an access point
- + This allows fresh pairwise keys to be derived, ensuring secure communications.
- + Nonces are used as inputs to Key Derivation Functions (KDF) along MAC addresses and PMK.
- + Nonces are produced by a cryptographically secure pseudo random number generators (CSPRNG)

```
fd          ttywf
fsevents    uart.debug-console
klog        urandom
lockstat    zero
keithbogart@Keith-Bogart /dev %
```



PTK Inputs



+ After WLAN Client receives EAPOL M1 (message 1) it has the following information;

- + The PMK
- + Its own MAC
- + The SNONCE it is about to send to the AP
- + The ANONCE from the AP
- + The MAC of the AP

WPA2 PTK Generation

What I know:

- PMK = xxyy
- My MAC = 223
- SNONCE = 33456
- AP MAC = 678
- ANONCE = 12345



EAPOL M1 (ANONCE = 12345) AP MAC = 678



Invoke PRF-512

$(\text{PMK} || \text{"Pairwise Key Expansion"} || \text{Min}(\text{MAC_A}, \text{MAC_B}) || \text{Max}(\text{MAC_A}, \text{MAC_B}) || \text{Min}(\text{ANonce}, \text{SNonce}) || \text{Max}(\text{ANonce}, \text{SNonce}) || \text{Counter})$

xxyy || "Pairwise Key Expansion" || 223 || 678 || 12345 || 33456 || 0x04

HMAC-SHA1

Output-1: 123456789 (160-bits)
Output-2: abcdef123 (160-bits)
Output-3: 987654321 (160-bits)
Output-3: 9876abcde (160-bits)

160-bits x 4 = 640-bits

Output-1 || Output-2 || Output-3 || Output-4

Trim last 128-bits

FINAL 512-bit PTK!!



Exchange of Values

Message 1

AP → **Client**: ANonce



Message 2

Client → **AP**: SNonce & MIC



Message 3

AP → **Client**: GTK & MIC



Message 4

Client → **AP**: ACK & MIC

- + PMK originates from PSK or EAP-derived MSK
- + **PTK computed from PMK, ANonce, SNonce & MAC addresses**
- + GTK sent in Message 3 enables broadcast encryption
- + Secure data session begins after Message 4 (keys installed)

WPA3 PTK Generation

What I know:

- PMK = xxyy
- My MAC = 223
- SNONCE = 33456
- AP MAC = 678
- ANONCE = 12345



EAPOL M1 (ANONCE = 12345) AP MAC = 678



Invoke PRF-256

$(\text{PMK} || \text{"Pairwise Key Expansion"} || \text{Min}(\text{MAC_A}, \text{MAC_B}) || \text{Max}(\text{MAC_A}, \text{MAC_B}) || \text{Min}(\text{ANonce}, \text{SNonce}) || \text{Max}(\text{ANonce}, \text{SNonce}) || \text{Counter})$

$\text{xxyy} || \text{"Pairwise Key Expansion"} || 223 || 678 || 12345 || 33456 || 0x02$

HMAC-SHA256

Output-1: 123456789 (256-bits)

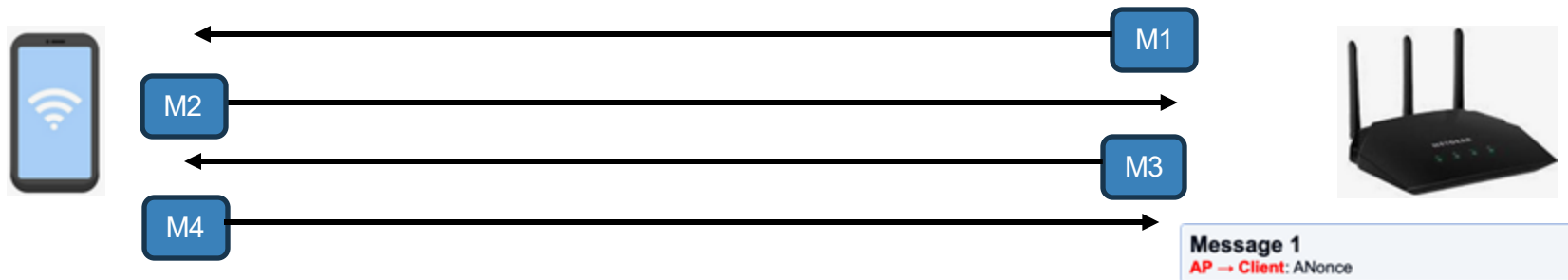
Output-2: abcdef123 (256-bits)

256-bits x 2 = 512-bits

Output-1 || Output-2

FINAL 512-bit PTK!!

Temporal Keys (WPA2 & WPA3)



Divide 512-bit PTK into sub-keys

123456789a b c d e f 123998765465432987abc

KCK (Key Confirmation Key) = 16-bytes

TK (Temporal Key) = 16-bytes

Remaining bytes = Optional
MIC keys for TKIP

KEK (Key Encryption Key) = 16-bytes
[Encrypts GTK sent in M3]



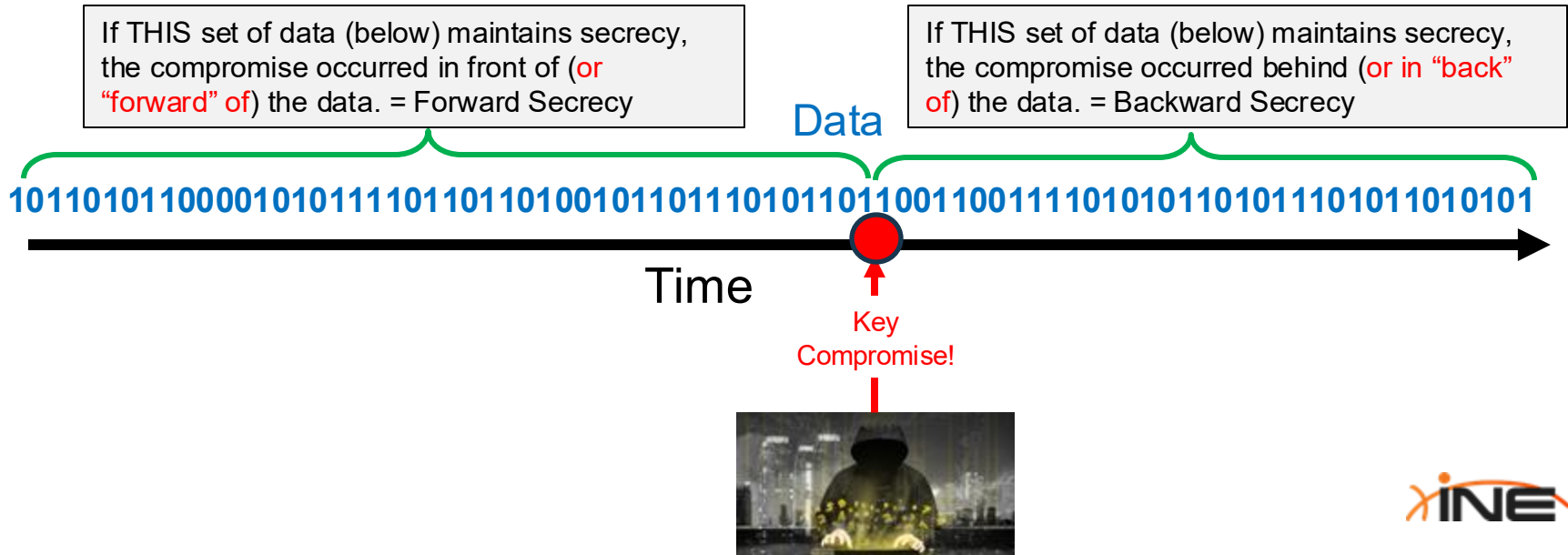
**Thank you for
watching!!**



WPA3 Perfect Forward Secrecy

Forward & Backward

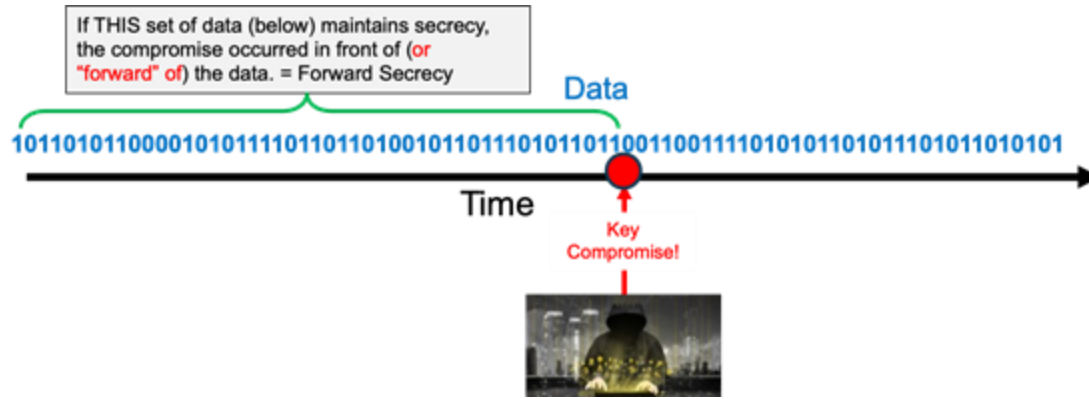
- + In cryptography, the terms “forward” or “backward” as applied to secrecy is determined if the moment of compromise is in front of, or behind, the data that remains protected.



WPA3 Perfect Forward Secrecy

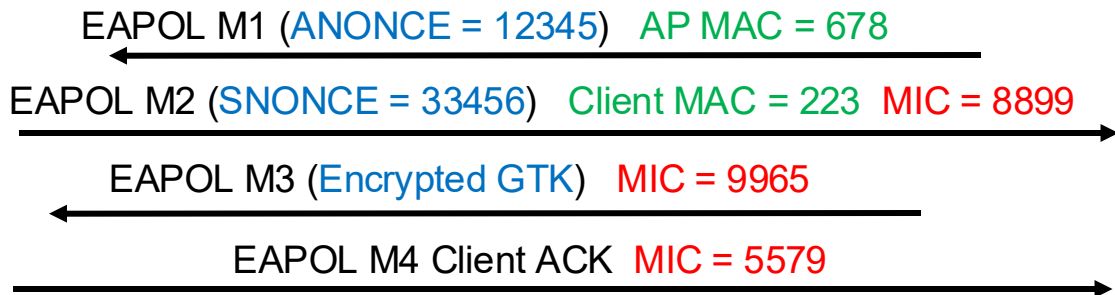
- + Perfect Forward Secrecy (PFS) refers to cryptographic key-exchange protocols (e.g., ephemeral Diffie-Hellman) where;
 - + *Compromise of long-term keys* cannot be used to recover any past session keys
 - + Even if an attacker recorded the EAPOL-Key handshake messages.
- + In this context, “long-term keys” refers to secret information that doesn’t change over time;
 - + WPA2 PSK
 - + WPA2 PMK
 - + WPA3 Passphrase

WPA3 PFS



- + If an attacker captures your data for some time and then later cracks the encryption key, only data AFTER the key has been cracked is at risk.
- + Data from WLAN sessions prior to the hack is safe, even if prior 4-way EAPOL-Key handshakes are captured.

Why WPA2 Doesn't Provide PFS



Malicious Actor starts collecting data

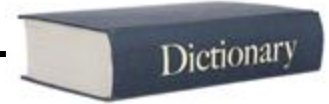
What I know:

- PMK = ???
- Client MAC = 223
- SNONCE = 33456
- AP MAC = 678
- ANONCE = 12345
- Various MIC values

WPA2 Offline Dictionary Attack



Dictionary Item#1 (*maybe this is the WPA2 PSK??*)



Pass Item#1 into
PBKDF2 algorithm

Potential PMK = *xxyy??*

Invoke PRF-512

$(\text{PMK} || \text{"Pairwise Key Expansion"} || \text{Min}(\text{MAC_A}, \text{MAC_B}) || \text{Max}(\text{MAC_A}, \text{MAC_B}) || \text{Min}(\text{ANonce}, \text{SNonce}) || \text{Max}(\text{ANonce}, \text{SNonce}) || \text{Counter})$

xxyy || "Pairwise Key Expansion" || 223 || *678* || *12345* || 33456 || 0x01

HMAC-SHA1

Output-1: 123456789 (160-bits)
Output-2: abcdef123 (160-bits)
Output-3: 987654321 (160-bits)
Output-3: 9876abcde (160-bits)

160-bits x 4 = 640-bits

Output-1 || Output-2 || Output-3 || Output-4

Trim last 128-bits

(Potential) 512-bit PTK!!



Cracking WPA2 Personal

Divide 512-bit PTK into sub-keys

123456789a bcdef123998765465432987abc

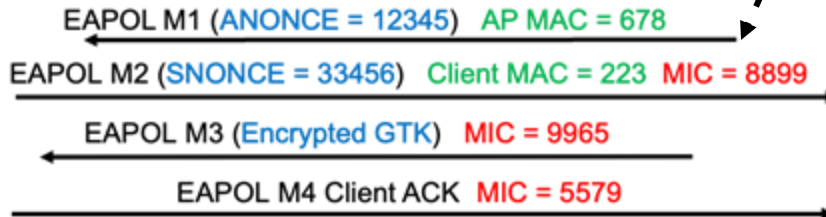
KCK (Key Confirmation Key) = 16-bytes

Can I compute the same MIC with my KCK?

Malicious Actor



1. If YES...I've got the PMK and WPA2 PSK!!
2. If NO...Try again with another dictionary item!



WPA2 Does Not Have PFS

- + Once the malicious actor has decoded the WPA2 Personal PSK, he knows that the **PMK will always be static and unchanging.**
- + He can go back in time and look at all captured Wi-Fi sessions for this client and;
 - + As long as he has the EAPOL 4-way handshake
 - + Crack all of those sessions.
 - + Decrypt all past traffic from that client

WPA3 PFS

- + The key to cracking a Wi-Fi session (and decrypting data) is to obtain the PMK
- + WPA-2's PMK is directly computed from the PSK and does not change
- + WPA-3's PMK;
 - + Derived from SAE/Dragonfly EDCH method
 - + Virtually impossible to crack
 - + Changes on every session
- + Even if a PMK was cracked for THIS session what would not help decrypt traffic from previous sessions.



**Thank you for
watching!!**

WPA Protected Management Frames (PMF)

Wi-Fi Frame Types Review

- + The 802.11 standard defines three types of Wi-Fi frames;
 - + Data
 - + Control (e.g., RTS, CTS)
 - + Management
- + Management frames include;
 - + Beacons
 - + Probe Requests (and Responses)
 - + Authentication
 - + Association
 - + Action Frames

Management Frame Vulnerability

- + Management frames are used to initiate and tear down sessions for network services
- + Management frames can be spoofed, leading to;
 - + Denial of service
 - + Luring clients to Evil Twin access points
 - + Slowing down Wi-Fi traffic



802.11w

- + Management frames can be categorized into two types:
 - + Those sent prior to EAPOL handshakes
 - + Those sent after
- + 802.11w specified several types of Management frames that should be protected as “Robust Management Frames”
- + Robust Management Frames are protected via PMF (“Protected Management Frames”)
- + Robust Management Frames are only used between Wireless stations after the EAPOL handshake is done.

What Frames are Protected?

- + Robust Management frames include;
 - + Disassociation
 - + De-authentication
 - + Robust Action frames
- + Robust “Action” frames include:
 - + Spectrum Management
 - + CSA (Channel Switch Announcements)
 - + QoS
 - + Block Ack
 - + Fast BSS Transition
 - + ...and others

Where PMF Applies

- + PMF allows the addition of a cryptographic signature to Robust Management frames sent after EAPOL handshake
- + Adds a MIC ([Message Integrity Code](#)) field these frames
 - + The MIC allows the receiver to verify that the frame wasn't tampered with in transit and came from the legitimate peer (AP or STA).
- + For unicast Robust Management frames, the MIC is computed using the client's KCK (Key Confirmation Key)
- + For broadcast frames (Channel Switch Announcement, etc) the MIC is computed using the access point's IGTK (Integrity Group Temporal Key) which is derived from the AP's Group Master Key ([More on this later](#))

Comparing Management Frames

Source	Destination	Protocol	Length	Info
Cisco_75:9d:cf	MotorolaMobi_ba:e...	802.11	66	Disassociate, SN=2655, FN=0, Flags=.....F.C
Cisco_75:9d:cf	MotorolaMobi_ba:e...	802.11	66	Disassociate, SN=2655, FN=0, Flags=....R.F.C
Cisco_75:9d:cf	MotorolaMobi_ba:e...	802.11	66	Deauthentication, SN=2656, FN=0, Flags=.....C

- > Frame 2106: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
- > Radiotap Header v0, Length 36
- > 802.11 radio information
- > IEEE 802.11 Deauthentication, Flags:C
- ▼ IEEE 802.11 Wireless Management
 - ▼ Fixed parameters (2 bytes)
 - Reason code: Unknown (0x00fc)

Source	Destination	Protocol	Length	Info
Cisco_75:9d:cd	b2:b7:4b:ab:29:65	802.11	82	Disassociate, SN=6, FN=0, Flags=.p....F.C
Cisco_75:9d:cd	b2:b7:4b:ab:29:65	802.11	82	Deauthentication, SN=7, FN=0, Flags=.p..R...C
Cisco_75:9d:cd	b2:b7:4b:ab:29:65	802.11	66	Deauthentication, SN=3037, FN=0, Flags=.....C

- > Frame 2598: 82 bytes on wire (656 bits), 82 bytes captured (656 bits)
- > Radiotap Header v0, Length 36
- > 802.11 radio information
- > IEEE 802.11 Deauthentication, Flags: .p..R...C
- ▼ Data (10 bytes)
 - Data: 9ff21e29545dd451f143
 - [Length: 10]

MIC added by PMF!

Protected Frame Indicators

```
Cisco_75:9d:cd    b2:b7:4b:ab:29:65  802.11    82 Disassociate, SN=6, FN=0, Flags=.p....F.C
Cisco_75:9d:cd    b2:b7:4b:ab:29:65  802.11    82 Deauthentication, SN=7, FN=0, Flags=.p..R...C
Cisco_75:9d:cd    b2:b7:4b:ab:29:65  802.11    66 Deauthentication, SN=3037, FN=0, Flags=.....C

> Frame 2598: 82 bytes on wire (656 bits), 82 bytes captured (656 bits)
> Radiotap Header v0, Length 36
> 802.11 radio information
  > IEEE 802.11 Deauthentication, Flags: .p..R...C
    Type/Subtype: Deauthentication (0x000c)
  > Frame Control Field: 0xc048
    .... ..00 = Version: 0
    .... 00.. = Type: Management frame (0)
    1100 .... = Subtype: 12
  > Flags: 0x48
    .... ..00 = DS status: Not leaving DS or network is operating in AD-HOC mode (To DS: 0 From DS: 0) (0x0)
    .... .0.. = More Fragments: This is the last fragment
  > .... 1... = Retry: Frame is being retransmitted
    > [Expert Info (Note/Sequence): Retransmission (retry)]
    ...0 .... = PWR MGT: STA will stay up
    ..0. .... = More Data: No data buffered
    .1.. .... = Protected flag: Data is protected
    0... .... = +HIC/Order flag: Not strictly ordered
    .000 0000 0011 1100 = Duration: 60 microseconds
  > Receiver address: b2:b7:4b:ab:29:65 (b2:b7:4b:ab:29:65)
  > Destination address: b2:b7:4b:ab:29:65 (b2:b7:4b:ab:29:65)
  > Transmitter address: Cisco_75:9d:cd (78:72:5d:75:9d:cd)
  > Source address: Cisco_75:9d:cd (78:72:5d:75:9d:cd)
  > BSS Id: Cisco_75:9d:cd (78:72:5d:75:9d:cd)
    .... .... 0000 = Fragment number: 0
    0000 0000 0111 .... = Sequence number: 7
    Frame check sequence: 0x24b116a6 [unverified]
    [FCS Status: Unverified]
    [WLAN Flags: .p..R...C]
  > CCMP parameters
    CCMP Ext. Initialization Vector: 0x000000000000
    Key Index: 0
  > Data (10 bytes)
    Data: 9ff21e29545dd451f143
    [Length: 10]
```

Beacons Indicate PMF Requirements

```
Cisco_75:9d:cd Broadcast 802.11 373 Beacon frame, SN=1333, FN=0, Flags=.....C, BI=100, SSID="WPA3-Test"

> Tag: Power Constraint: 0
> Tag: RSN Information
  Tag Number: RSN Information (48)
  Tag length: 26
  RSN Version: 1
> Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)
  Pairwise Cipher Suite Count: 1
> Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)
  Auth Key Management (AKM) Suite Count: 1
> Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) SAE (SHA256)
> RSN Capabilities: 0x00e8
  .... = RSN Pre-Auth capabilities: Transmitter does not support pre-authentication
  .... = RSN No Pairwise capabilities: Transmitter can support WEP default key 0 simultaneously wit
  .... = RSN PTKSA Replay Counter capabilities: 4 replay counters per PTKSA/GTKSA/STakeySA (0x2)
  .... = RSN GTKSA Replay Counter capabilities: 4 replay counters per PTKSA/GTKSA/STakeySA (0x2)
  .... = Management Frame Protection Required: True
  .... = Management Frame Protection Capable: True
  .... = Joint Multi-band RSNA: False
  .... = PeerKey Enabled: False
  ..0. = Extended Key ID for Individually Addressed Frames: Not supported
  .0.. = OCVC: False
  PMKID Count: 0
  PMKID List
> Group Management Cipher Suite: 00:0f:ac (Ieee 802.11) BIP (128)
```

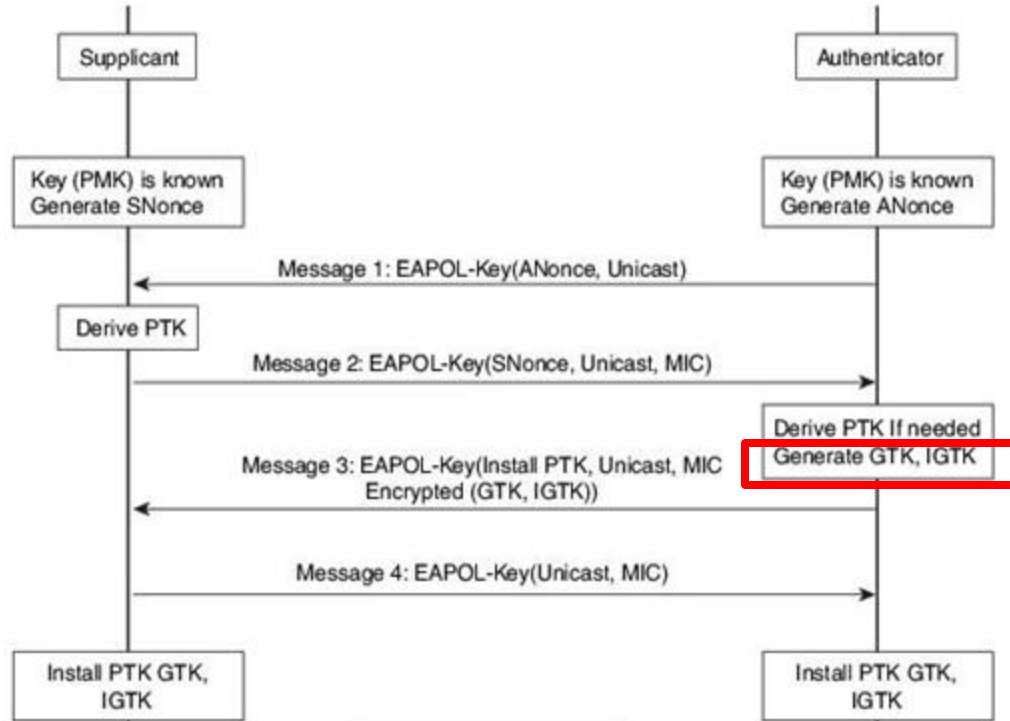
What about BIP?

- + Some Robust Management Frames are sent as broadcasts or multicasts;
 - + Multicast deauths
 - + Channel switch announcements
 - + Group addressed action frames
- + These frames can't use unicast PTK-based protection, since they are sent to multiple clients at once.
- + Instead, BIP is used (*Broadcast/Multicast Integrity Protocol*)

BIP & the IGTK

- + 802.11w introduced a new key for use with BIP, the IGTK
 - + Integrity Group Temporal Key
- + IGTK is derived by the Access Point
- + IGTK is given to WLAN clients (encrypted) in EAPOL Message-3
- + Neither CCMP nor GCMP can be used to derive MICs for broadcast/multicast frames due to its intended use for unicast
- + BIP is a new protocol that (used with IGTK) creates a MIC for broadcast/multicast Robust Management Frames

IGTK Propagation



PMF Final Thoughts

- + PMF MIC generation
 - + For unicast robust management frames → TK + CCMP/GCMP → MIC
 - + For broadcast/multicast robust management frames → GTK + BIP → MIC
- + WPA2 first offered PMF as an optional service
- + WPA3 requires PMF





**Thank you for
watching!!**

WPA3 Beacon Protection

Beacon Protection

- + 802.11 Beacon frames are;
 - + Used by clients to discover WLANs and their capabilities
 - + Maintain associations to WLANs
 - + Sent in plaintext and unencrypted
- + Once associated to a WLAN, Beacon spoofing can result in;
 - + Disassociation of clients
 - + Rate-limiting of clients (throttling of transmit power)
 - + Rogue CSAs forcing clients to an unauthorized channel
- + WPA3 added Beacon Protection to reduce some of these risks.

How Beacons are Protected

- + Beacon Protection adds a computed MIC (*message integrity code*) as an informational element to beacons
- + Relies on a *Beacon Integrity Group Temporal Key* (BIGTK)
 - + Carried inside EAPOL-Key Message-3 frames (encrypted by the KEK)
 - + This key is only for integrity, not encryption.
 - + Shared among all clients associated with that SSID
 - + Requires PMF
- + Beacon protection is optional in Wi-Fi 6e and lower, but mandatory in Wi-Fi 7.

Catalyst 9800: Enabling Beacon Protection

Edit WLAN

⚠ Changing WLAN parameters while it is enabled will result in loss of connectivity for clients connected to it.

General **Security** Advanced Add To Policy Tags

Layer2 **Layer3** AAA

☐ WPA + WPA2 ☐ WPA2 + WPA3 ☒ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
Transition Disable	<input type="checkbox"/>	Beacon Protection	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF Required ▼

Fast Transition

Status Disabled ▼

Over the DS ☐

Reassociation Timeout * 20

Auth Key Mgmt (AKM)

FT + 802.1X	<input type="checkbox"/>	802.1X-SHA256	<input type="checkbox"/>
OWE	<input type="checkbox"/>	SAE	<input checked="" type="checkbox"/>
FT + SAE	<input type="checkbox"/>	SAE-EXT-KEY	<input type="checkbox"/>
FT + SAE-EXT-KEY	<input type="checkbox"/>		

Requires 17.15 code or later.

Validating Beacon Protection

```
> IEEE 802.11 Beacon frame, Flags: .....
> IEEE 802.11 Wireless Management
  > Fixed parameters (12 bytes)
    Timestamp: 56
    Beacon Interval: 0.102400 [Seconds]
    > Capabilities Information: 0x1511
  > Tagged parameters (524 bytes)
    > Tag: SSID parameter set: " "
    > Tag: Extended Capabilities (11 octets)
      Tag Number: Extended Capabilities (127)
      Tag length: 11
      > Extended Capabilities: 0x04 (octet 1)
      > Extended Capabilities: 0x10 (octet 2)
      > Extended Capabilities: 0x0f (octet 3)
      > Extended Capabilities: 0x06 (octet 4)
      > Extended Capabilities: 0x01 (octet 5)
      > Extended Capabilities: 0x00 (octet 6)
      > Extended Capabilities: 0x00 (octet 7)
      > Extended Capabilities: 0x0040 (octets 8 & 9)
      > Extended Capabilities: 0x40 (octet 10)
      > Extended Capabilities: 0x10 (octet 11)
      .... ..0 = Complete List of NonTxBSSID Profiles: False
      .... ..0. = SAE Password Identifiers In Use: False
      .... .0.. = SAE Passwords Used Exclusively: False
      .... 0... = Enhanced Multi-BSSID Advertisement Support: False
      ...1 .... = Beacon Protection Enabled: True
      ..0. .... = Mirrored SCS: False
      .0.. .... = OCT: False
      0... .... = Local MAC Address Policy: False
```

Management MIC Element (MME)

```
> Tag: Vendor Specific: Qualcomm
> Tag: Management MIC
  Tag Number: Management MIC (76)
  Tag length: 16
  KeyID: 7
  IPN: 0x000000b2c4e0
  MIC: 3b359d05c1f82527
```

Transmission of the BIGTK

- + The BIGTK (along with other keys such as the GTK) is encrypted by the Key Encryption Key (KEK) when transmitted by the AP.

```
> 802.11 radio information
> IEEE 802.11 QoS Data, Flags: .....F.
> Logical-Link Control
v 802.1X Authentication
  Version: 802.1X-2004 (2)
  Type: Key (3)
  Length: 407
  Key Descriptor Type: EAPOL RSN Key (2)
  [Message number: 3]
> Key Information: 0x13c8
  Key Length: 32
  Replay Counter: 2
  WPA Key Nonce: e6e4c574e343c086405719511030eb9b53adbc6caa330aedb548897a61c990c
  Key IV: 00000000000000000000000000000000
  WPA Key RSC: 138c060000000000
  WPA Key ID: 0000000000000000
  WPA Key MIC: 0ac977462b2579919da64f71f61bcf187f11ee86b3fe701f
  WPA Key Data Length: 304
  WPA Key Data [...]: f0d0bde1489e57b8b72753c4d52915c195292a2707d3c836fc9d0370db5355e54e53ac7612cd603a83b327a7a497edf3a58...
```

Beacon Protection: Final Thoughts

- + Beacon Protection ONLY protects clients from rogue AP's attempting to impersonate (via MAC duplication) the legitimate AP.
- + Does NOT protect against APs with same SSID but different MAC address.
- + Clients are still free to implement roaming logic to migrate to other (bad/evil) access points.
- + *Older Cisco Wave 2 Aironet access point models (3700/3800/1810/1830/1850 series) do not have the hardware capability to support this feature.*



**Thank you for
watching!!**

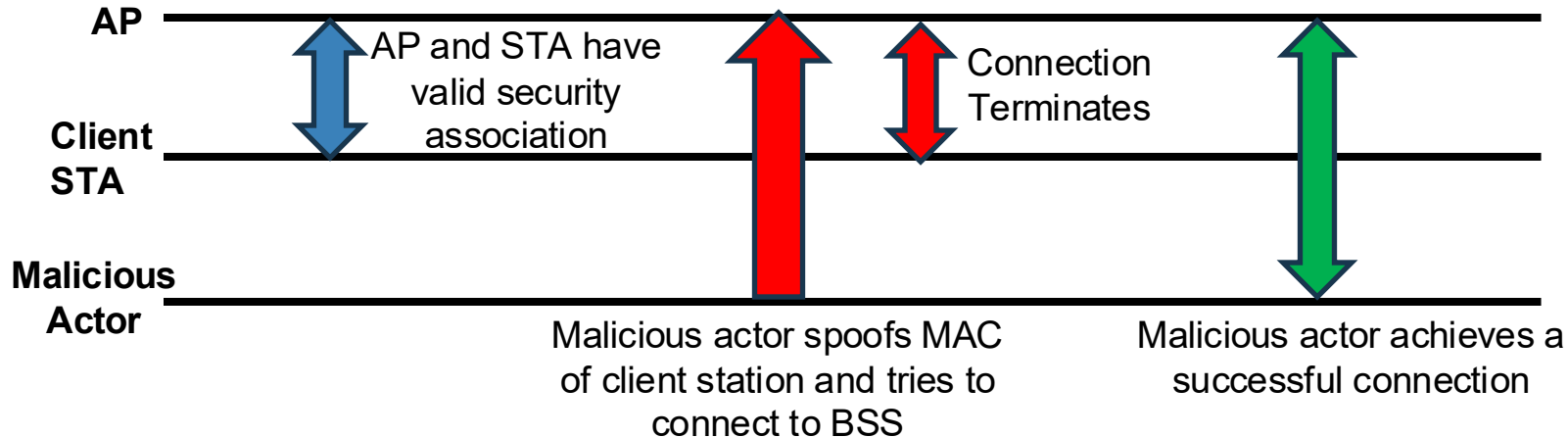


PMF & SA-Queries

Utilizing PMF to Protect against Spoofed Association Requests

Attack by Association

- + Prior to PMF, associated clients could spoof other associated clients and knock them off of the Wi-Fi
- + Simply sending an “Association Request” frame spoofed from an existing client could accomplish this.



Prevention against Spoofed Associations

- + PMF includes two mechanisms to prevent Association Attacks
 - + An Association Comeback Timer
 - + SA Query procedure
- + Association Comeback Timer
 - + Association request received
 - + AP/WLC checks for presence of current association from same client
 - + If yes, AP denies Association Request with Status Code 30 “Comeback Later”

Denying Spoofed Association Requests

- Frame 252: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface en0, id 0
- Radiotap Header v0, Length 25
- 802.11 radio information
- IEEE 802.11 Association Response, Flags:C
- IEEE 802.11 Wireless Management

- Fixed parameters (6 bytes)

- Capabilities Information: 0x0411

Status code: Association request rejected temporarily; try again later (0x001e)
..00 0000 0000 0001 = Association ID: 0x0001

Status Code: 30

- Tagged parameters (33 bytes)

- Tag: Supported Rates 1(B), 2(B), 5.5(B), 11(B), 6, 9, 12, 18, [Mbit/sec]
- Tag: Extended Supported Rates 24, 36, 48, 54, [Mbit/sec]

- Tag: Timeout Interval

Tag Number: Timeout Interval (56)

Tag length: 5

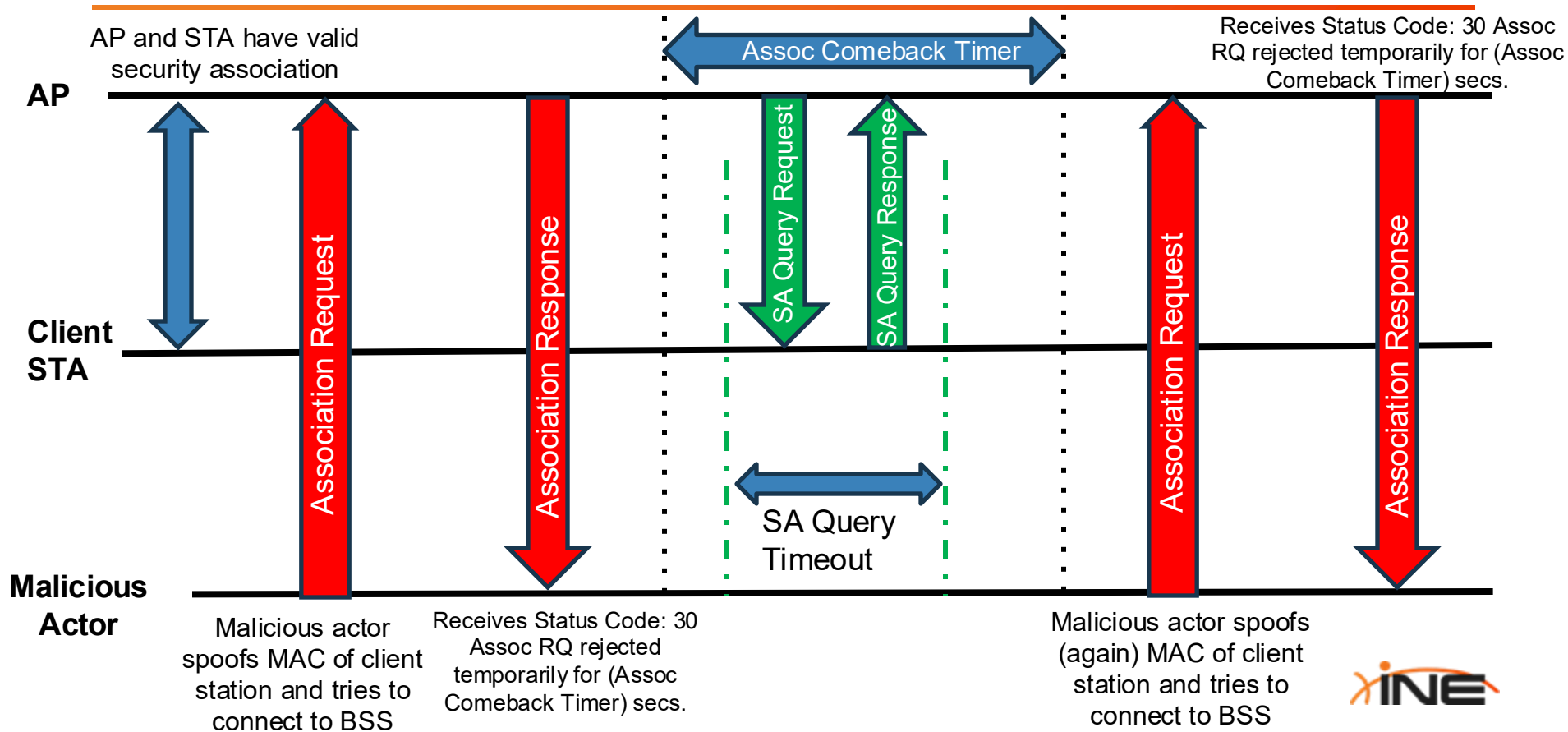
Timeout Interval Type: Association Comeback time (TUs) (3)
Timeout Interval Value: 1000

- Tag: Extended Capabilities (8 octets)

PMF SA Queries

- + Security Association Queries
- + Triggered upon WLC/AP starting the Association Comeback Timer
 - + WLC/AP transmits a “SA Query Request” within an Action frame to the currently-associated client (for which attacker was attempting to spoof)
 - + If legitimate client still exists, it must respond with “SA Query Response” within the SA Query interval
- + The data within the SA Query frames is encrypted using CCMP and the client’s TK (Temporal Key)

SA Queries and Response Process



Cisco 9800 WLC PMF Configuration

- + Timers and values for both of these PMF features can be configured within the Cisco 9800 WLC under “edit > WLAN”

Protected Management Frame

PMF	Required ▼
Association Comeback Timer*	1 - 20 secs
SA Query Time*	100 - 500 msecs



**Thank you for
watching!!**

WPA Transition Mode

The Problem Defined

- + WPA3 has several mechanisms that make it inherently stronger and more secure than WPA2
- + However, not all WLAN clients support WPA3
 - + Older smartphones and tablets (Android 9 or older)
 - + IoT devices
 - + Home Automation platforms (i.e. “Home Assistant”)
- + How can you offer an SSID that works for both WPA3 and WPA2 clients?

Introduction to WPA Transition Mode

- + WPA Transition mode = *A single SSID that advertises both WPA2 and WPA3 capability*
- + Clients should connect via WPA3 (SAE) unless they only support WPA2 (PSK)
- + PMF should be set to “Optional”
 - + Some WPA2 support PMF
 - + WPA2 clients that don't support PMF won't be able to connect to your WLAN if PMF set to “Required”

Creating a Transition Mode WLAN

Edit WLAN

General **Security** Advanced Add To Policy Tags

Layer2 Layer3 AAA

☐ WPA + WPA2 ☒ WPA2 + WPA3 ☐ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input checked="" type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
Transition Disable	<input type="checkbox"/>	Beacon Protection	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF

Association Comeback Timer*

SA Query Time*

Fast Transition

Status

Over the DS ☐

Reassociation Timeout *

Auth Key Mgmt (AKM)

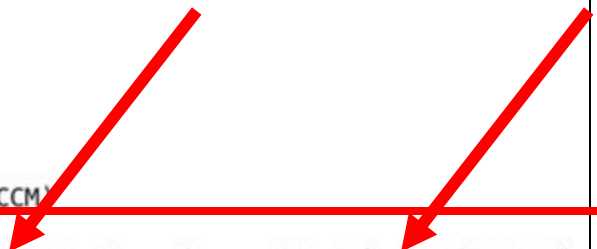
802.1X	<input type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>	CCKM ⚠	<input type="checkbox"/>
PSK	<input checked="" type="checkbox"/>	FT + PSK	<input type="checkbox"/>
PSK-SHA256	<input type="checkbox"/>	SAE	<input checked="" type="checkbox"/>
FT + SAE	<input type="checkbox"/>	SAE-EXT-KEY	<input type="checkbox"/>
FT + SAE-EXT-KEY	<input type="checkbox"/>		

Anti Clogging Threshold*

Max Retries*

WPA Transition Beacon

```
> IEEE 802.11 Beacon frame, Flags: .....C
~ IEEE 802.11 Wireless Management
  > Fixed parameters (12 bytes)
  ~ Tagged parameters (309 bytes)
    > Tag: SSID parameter set: "WPA-Transition"
    > Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
    > Tag: DS Parameter set: Current Channel: 100
    > Tag: Traffic Indication Map (TIM): DTIM 0 of 1 bitmap
    > Tag: Country Information: Country Code US, Environment All
    > Tag: Power Constraint: 0
    ~ Tag: RSN Information
      Tag Number: RSN Information (48)
      Tag length: 30
      RSN Version: 1
      > Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)
      Pairwise Cipher Suite Count: 1
      > Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)
      Auth Key Management (AKM) Suite Count: 2
      ~ Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) PSK 00:0f:ac (Ieee 802.11) SAE (SHA256)
        > Auth Key Management (AKM) Suite: 00:0f:ac (Ieee 802.11) PSK
        > Auth Key Management (AKM) Suite: 00:0f:ac (Ieee 802.11) SAE (SHA256)
      > RSN Capabilities: 0x00a8
      PMKID Count: 0
      PMKID List
      > Group Management Cipher Suite: 00:0f:ac (Ieee 802.11) BIP (128)
```



The diagram illustrates the structure of the WPA Transition Beacon's RSN Information field. Two red arrows point from the 'Auth Key Management (AKM) List' section to the two 'Auth Key Management (AKM) Suite' entries, highlighting the supported key management methods: PSK and SAE (SHA256).

Conflicting Cipher Suites

- + Can I use AES-CCMP128 *AND* GCMP128 in a Transition WLAN? **No.**
 - + WPA2 clients understand only CCMP128 (AES).
 - + WPA3 clients can use GCMP128, *but only when the SSID is WPA3-only.*
 - + WPA Transition WLANs are restricted to CCMP128 only.

Cipher Suite Combination Error

General **Security** Advanced Add To Policy Tags

Layer2 **Layer3** AAA

☐ WPA + WPA2 ☒ WPA2 + WPA3 ☐ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input checked="" type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
Transition Disable	<input type="checkbox"/>	Beacon Protection	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input checked="" type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF

Association Comeback Timer*

SA Query Time*

Fast Transition

Status

Over the DS ☐

Reassociation Timeout *

Auth Key Mgmt (AKM)

WPA2+WPA3 security valid Encryption and AKM combinations

- GCMP128 Cipher + SuiteB AKM

802.1X	<input type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>	SUITEB-1X	<input type="checkbox"/>
CCKM ⚠	<input type="checkbox"/>	PSK	<input checked="" type="checkbox"/>
FT + PSK	<input type="checkbox"/>	PSK-SHA256	<input type="checkbox"/>
SAE	<input checked="" type="checkbox"/>	FT + SAE	<input type="checkbox"/>
SAE-EXT-KEY	<input type="checkbox"/>	FT + SAE-EXT-KEY	<input type="checkbox"/>

Transition Disable

- + Catalyst 9800 WLCs offer a “Transition Disable” checkbox
- + If checked, Beacons will set the “transition disable” flag
- + Once a WLAN Client connects to this SSID via WPA3 it will;
 - + Cache the “transition disable” status for this WLAN
 - + Refuse to connect to this WLAN via WPA2 on any other access point
- + Good to select IF all your access points support WPA2 and WPA3

Transition Disable

Layer2 Layer3 AAA

☐ WPA + WPA2 ☒ WPA2 + WPA3 ☐ WPA

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input checked="" type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
Transition Disable	<input type="checkbox"/>	Beacon Protection	<input type="checkbox"/>

Transition Disable Caveats

- + The Transition Disable flag is part of a *post-2018 WPA3 specification update*
- + Cisco added controller-side awareness in later 9800 code (starting around 17.5 / 17.6)
- + Only newer AP platforms can propagate the RSN bit that advertise “Transition Disable” in their beacons.
- + Older Cisco Wave 2 Aironet access point models (3700/3800/1810/1830/1850 series) do not have the hardware capability to support this feature.

Transition Mode Risks

- + Transition Mode introduces security trade-offs:
 - + Downgrade Attacks:
 - + A malicious actor could spoof beacons attempting to influence WPA3 clients to connect using WPA2.
 - + Mixed Security:
 - + Since some devices may still use weaker WPA2-PSK, the overall network inherits WPA2's vulnerabilities.
- + Wi-Fi Alliance recommends:
 - + Using Transition Mode only temporarily.
 - + Once all clients support WPA3, switch to WPA3-only mode.



**Thank you for
watching!!**



WPA3 Enhanced Open

The Challenge

- + People want to be able to utilize public Wi-Fi networks
 - + Coffee Shops
 - + Hotels
 - + Airports
 - + Uncle Bob's house
- + People want their Wi-Fi traffic encrypted
- + People don't want to have to type in a passphrase, just click on the appropriate SSID and connect.
- + *How do we provide Wi-Fi encryption without a passphrase?*

Introducing Wi-Fi Enhanced Open

- + First introduced with WPA3
- + A security mode under the WPA3 umbrella aimed at open (no-password) Wi-Fi networks
- + *Goal is to encrypt traffic over the air even though there is no shared password or user authentication*
- + The mechanism used is Opportunistic Wireless Encryption (OWE)
 - + Cryptographic handshake that generates a unique encryption key between each client and the access point (AP)
 - + The network is “open” (i.e. no credentials)

High Level OWE Operation

- + Exchange of (empty) 802.11 “Authentication” frames.
- + Association / handshake:
 - + Client and the AP perform a Diffie-Hellman (or elliptic-curve) handshake.
 - + They exchange public parameters, compute a shared secret, and derive encryption keys.
 - + This is done without requiring a password.
- + After key derivation, standard 4-way EAPOL handshake ensues
 - + Derives unique encryption keys per client
- + Enhanced Open requires support of PMF

OWE Configuration (Cisco 9800 WLC)

Add WLAN

General **Security** Advanced

Layer2 **Layer3** AAA

☐ WPA + WPA2 ☐ WPA2 + WPA3 ☒ WPA3 ☐ Static WEP ☐ None

MAC Filtering ☐

Lobby Admin Access ☐

WPA Parameters

WPA Policy	<input type="checkbox"/>	WPA2 Policy	<input type="checkbox"/>
GTK Randomize	<input type="checkbox"/>	WPA3 Policy	<input checked="" type="checkbox"/>
		Transition Disable	<input type="checkbox"/>

WPA2/WPA3 Encryption

AES(CCMP128)	<input checked="" type="checkbox"/>	CCMP256	<input type="checkbox"/>
GCMP128	<input type="checkbox"/>	GCMP256	<input type="checkbox"/>

Protected Management Frame

PMF

Association Comeback Timer*

Fast Transition

Status

Over the DS ☐

Reassociation Timeout *

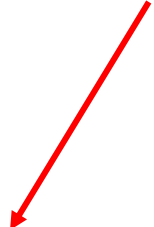
Auth Key Mgmt

SAE	<input type="checkbox"/>	FT + SAE	<input type="checkbox"/>
OWE	<input checked="" type="checkbox"/>	FT + 802.1X	<input type="checkbox"/>
802.1X-SHA256	<input type="checkbox"/>		
Transition Mode WLAN ID	<input type="text" value="0-4096"/>		

Notice that there is no place to input a passphrase.

OWE Beacons

```
> IEEE 802.11 Beacon frame, Flags: .....C
v IEEE 802.11 Wireless Management
  v Fixed parameters (12 bytes)
    Timestamp: 7558963211
    Beacon Interval: 0.102400 [Seconds]
    > Capabilities Information: 0x1111
  v Tagged parameters (296 bytes)
    > Tag: SSID parameter set: "OWE-Test"
    > Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
    > Tag: DS Parameter set: Current Channel: 100
    > Tag: Traffic Indication Map (TIM): DTIM 0 of 1 bitmap
    > Tag: Country Information: Country Code US, Environment All
    > Tag: Power Constraint: 0
  v Tag: RSN Information
    Tag Number: RSN Information (48)
    Tag length: 26
    RSN Version: 1
    > Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)
    Pairwise Cipher Suite Count: 1
    > Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)
    Auth Key Management (AKM) Suite Count: 1
  v Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) Opportunistic Wireless Encryption
    v Auth Key Management (AKM) Suite: 00:0f:ac (Ieee 802.11) Opportunistic Wireless Encryption
      Auth Key Management (AKM) OUI: 00:0f:ac (Ieee 802.11)
      Auth Key Management (AKM) type: Opportunistic Wireless Encryption (18)
```

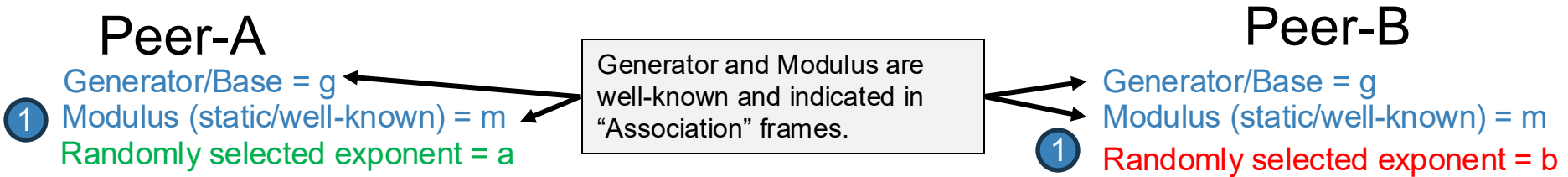


OWE Diffie-Hellman Group Indication

12:71:c8:91:9c:aa	Cisco_75:9d:cd	802.11	70 Authentication, SN=2654, FN=0, Flags=.....C
Cisco_75:9d:cd	12:71:c8:91:9c:aa	802.11	70 Authentication, SN=2113, FN=0, Flags=.....C
12:71:c8:91:9c:aa	Cisco_75:9d:cd	802.11	321 Association Request, SN=2655, FN=0, Flags=.....C, SSID="OWE-Test"
Cisco_75:9d:cd	12:71:c8:91:9c:aa	802.11	266 Association Response. SN=2114. FN=0. Flags=.....C

◦ Tag: RSN Information	00
Tag Number: RSN Information (48)	00
Tag length: 26	00
RSN Version: 1	00
> Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)	00
Pairwise Cipher Suite Count: 1	00
> Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)	00
Auth Key Management (AKM) Suite Count: 1	00
> Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) Opportunistic Wireless Encryption	00
> RSN Capabilities: 0x40c0	00
PMKID Count: 0	00
PMKID List	00
> Group Management Cipher Suite: 00:0f:ac (Ieee 802.11) BIP (128)	00
> Tag: RM Enabled Capabilities (5 octets)	00
> Tag: Supported Operating Classes	00
> Tag: HT Capabilities (802.11n D1.10)	01
> Tag: Extended Capabilities (8 octets)	01
> Tag: VHT Capabilities	01
> Ext Tag: OWE Diffie-Hellman Parameter	01
Ext Tag length: 34 (Tag len: 35)	
Ext Tag Number: OWE Diffie-Hellman Parameter (32)	
Group: 256-bit random ECP group (19)	
Public Key: dc17f66d7f8bd44e449cd7f506ca1b1401bbd2db513a3704e3503d99f162eaa6	
> Tag: Vendor Specific: Microsoft Corp.: WMM/WME: Information Element	
> Tag: Vendor Specific: Qualcomm Inc.	
> Tag: RSN eXtension (1 octet)	
Tag Number: RSN eXtension (244)	
Tag length: 1	
> RSNX: 0x00 (octet 1)	
.... 0000 = RSNX Length: 0	
...0 = Protected TWT Operations Support: False	
..0. = SAE Hash to element: False	

OWE Implementation of Diffie-Hellman



s entire process occurs during the
change of "Association" messages.

Full WPA3 OWE Connection

12:71:c8:91:9c:aa	Cisco_75:9d:cd	802.11	188	Probe Request, SN=2653, FN=0, Flags=.....C, SSID="OWE-Test"
Cisco_75:9d:cd	12:71:c8:91:9c:aa	802.11	366	Probe Response, SN=2112, FN=0, Flags=.....C, BI=100, SSID="OWE-Test"
12:71:c8:91:9c:aa	Cisco_75:9d:cd	802.11	70	Authentication, SN=2654, FN=0, Flags=.....C
Cisco_75:9d:cd	12:71:c8:91:9c:aa	802.11	70	Authentication, SN=2113, FN=0, Flags=.....C
12:71:c8:91:9c:aa	Cisco_75:9d:cd	802.11	321	Association Request, SN=2655, FN=0, Flags=.....C, SSID="OWE-Test"
Cisco_75:9d:cd	12:71:c8:91:9c:aa	802.11	266	Association Response, SN=2114, FN=0, Flags=.....C
Cisco_75:9d:cd	12:71:c8:91:9c:aa	EAPOL	193	Key (Message 1 of 4)
12:71:c8:91:9c:aa	Cisco_75:9d:cd	EAPOL	202	Key (Message 2 of 4)
Cisco_75:9d:cd	12:71:c8:91:9c:aa	EAPOL	275	Key (Message 3 of 4)
Cisco_75:9d:cd	12:71:c8:91:9c:aa	EAPOL	275	Key (Message 3 of 4)
12:71:c8:91:9c:aa	Cisco_75:9d:cd	EAPOL	171	Key (Message 4 of 4)

> Frame 219: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)	0000
> Radiotap Header v0, Length 36	0010
> 802.11 radio information	0020
> IEEE 802.11 Authentication, Flags:C	0030
> IEEE 802.11 Wireless Management	0040
< Fixed parameters (6 bytes)	
Authentication Algorithm: Open System (0)	
Authentication SEQ: 0x0001	
Status code: Successful (0x0000)	

OWE Transition Mode

- + Many legacy devices don't support OWE
- + WPA3 supports OWE Transition Mode (OWETM).
- + In this mode, the AP can broadcast both an open network (for legacy clients) and an OWE-enabled network.
- + Devices that support OWE use encryption; older ones fall back to plain open (no encryption).
- + Wi-Fi 6E and Wi-Fi 7 **do NOT allow OWETM**

Configuring OWETM

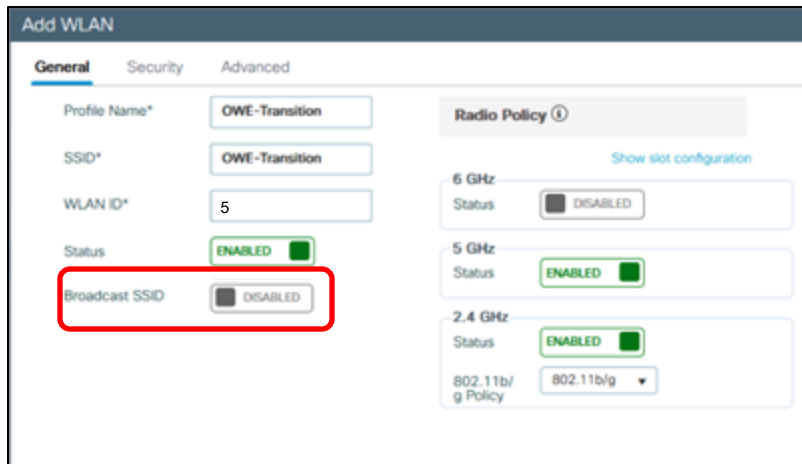
- + Step-1: Create an OPEN WLAN (no security) and mark it for OWE Transition usage.

The 'Add WLAN' window is shown with the 'General' tab selected. The 'Profile Name*' is 'Airport', 'SSID*' is 'Airport', and 'WLAN ID*' is '4'. The 'Status' is 'ENABLED' and 'Broadcast SSID' is also 'ENABLED'. On the right, the 'Radio Policy' section shows '6 GHz' as 'DISABLED', '5 GHz' as 'ENABLED', and '2.4 GHz' as 'ENABLED'. The '802.11b/g Policy' is set to '802.11g'.

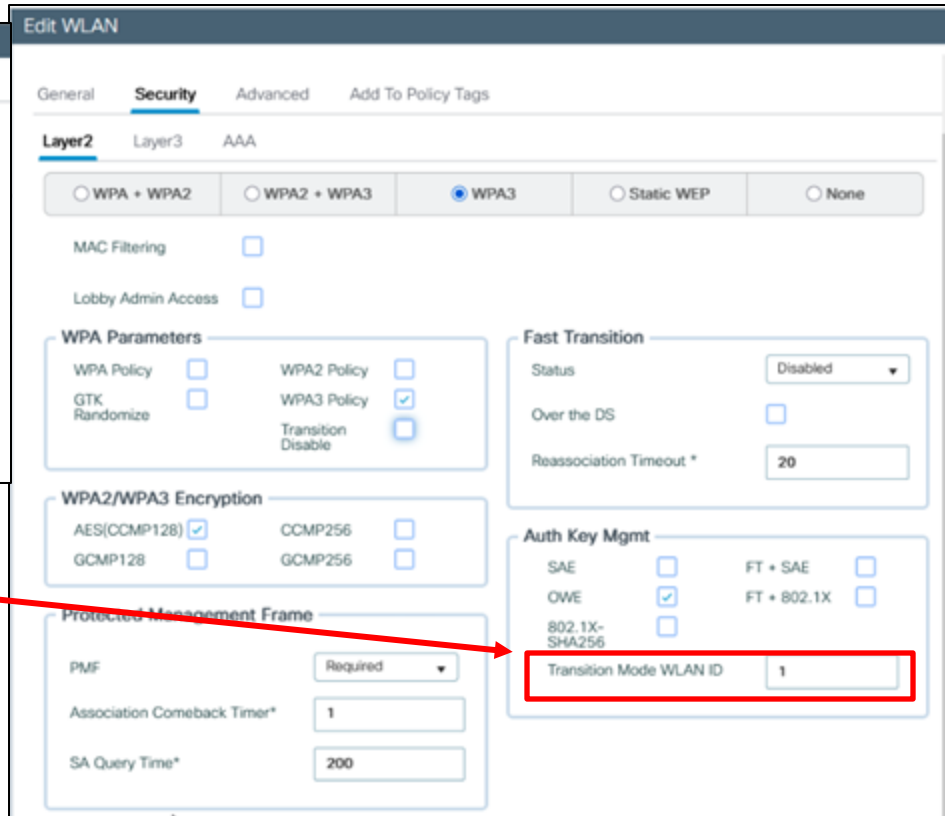
The 'Add WLAN' window is shown with the 'Security' tab selected. The 'Layer2' sub-tab is active. The security mode is set to 'None' (radio button selected). Below this, 'MAC Filtering' is unchecked. The 'OWE Transition Mode' checkbox is checked, and the 'Transition Mode WLAN ID*' is set to '1'. A red dashed arrow points from the '1' in the 'Transition Mode WLAN ID*' field to a text box above it.

Remember this number
for the next step.

Configuring OWETM



The 'Add WLAN' configuration screen shows the 'General' tab. The 'Profile Name' is 'OWE-Transition', 'SSID' is 'OWE-Transition', and 'WLAN ID' is '5'. The 'Status' is 'ENABLED'. The 'Broadcast SSID' option is highlighted with a red rectangle and is currently 'DISABLED'. The 'Radio Policy' section shows frequencies 6 GHz, 5 GHz, and 2.4 GHz, all with 'ENABLED' status. The 802.11b/g Policy is set to '802.11b/g'.



The 'Edit WLAN' configuration screen shows the 'Security' tab. The 'Layer2' section has 'WPA3' selected. The 'WPA Parameters' section shows 'WPA3 Policy' checked. The 'WPA2/WPA3 Encryption' section shows 'AES(CCMP128)' and 'GCMP128' both checked. The 'Protected Management Frame' section shows 'PMF' set to 'Required'. The 'Fast Transition' section shows 'Status' set to 'Disabled'. The 'Auth Key Mgmt' section shows 'OWE' checked. The 'Transition Mode WLAN ID' is highlighted with a red rectangle and set to '1'. A red arrow points from the text in the adjacent block to this field.

- + Step-2: Refer to your previously-configured transition WLAN ID within the WPA3 OWE WLAN.



**Thank you for
watching!!**



WPA3 Easy Connect

Back in the “Old Days”...

- + Early Wi-Fi setup (early 2000's) was confusing for non-technical users, requiring manual entry of long WPA/WPA2 passphrases.
- + Many devices (printers, cameras, TVs) had tiny screens or limited input controls, making passphrase entry tedious or error-prone.
- + Wi-Fi developers looked for a way to securely deliver Wi-Fi credentials to these types of devices/users in a simple way.

Passphrase:
1@33\$aAb1C4D&

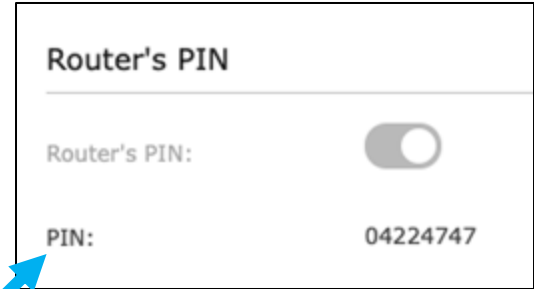


I have to type THAT
passphrase into THIS
tiny screen???!?



WPS...the Early Solution

- + *WPS = Wi-Fi Protected Setup*
- + Launched by Wi-Fi Alliance in 2007
- + Two device roles:
 - + Registrar (Access Point)
 - + Enrollee (Client)
- + Secure encrypted tunnel created between devices to pass WPA/WPA2 passphrase
- + WPS tunnel is encrypted by,
 - + WPS Pin (8-digit number)
 - + WPS Button



The Problems with WPS

- + *WPS not supported for WPA3 WLANs*
- + WPS PIN-derived shared secret proven crackable
- + WPS Pushbutton method makes WLAN available (within WPS time window) for ANY nearby device with WPS capability
- + *WPS is not supported on devices without GUI interface or pushbutton capability* (think Smart Lightbulbs)

Today's Problems to Solve

- + Only authorized users and devices should gain access to WLANs
- + Many Wi-Fi devices today are “headless” (no management Web-UI, or CLI is available)
 - + Wi-Fi Video Cameras and Doorbells
 - + Smart appliances
 - + Smart Bulbs
- + How can we **securely** provide Wi-Fi credentials to these types of devices?

Wi-Fi Easy Connect Introduction

- + Wi-Fi Alliance standard for secure, vendor-neutral device onboarding
- + Defines two device roles:
 - + Configurator – trusted device holding Wi-Fi network credentials
 - + Enrollee – device being provisioned (e.g., IoT sensor, thermostat)
- + Establishes a mutually authenticated, encrypted channel using public-key cryptography to securely deliver WPA3 credentials

Easy Connect Phases

- + Wi-Fi Easy Connect consists of three phases:
 - + *Bootstrap Information Exchange*
 - + *Device Provisioning Protocol (DPP)*
 - + *Normal 802.11 connection to BSS*
- + Bootstrap Information Exchange is the pre-DPP step where the Configurator learns the Enrollee's public key;
 - + Via QR code, NFC, BLE, etc).
- + The actual DPP protocol begins only after this step, handling secure mutual authentication and credential delivery.

Bootstrap Phase

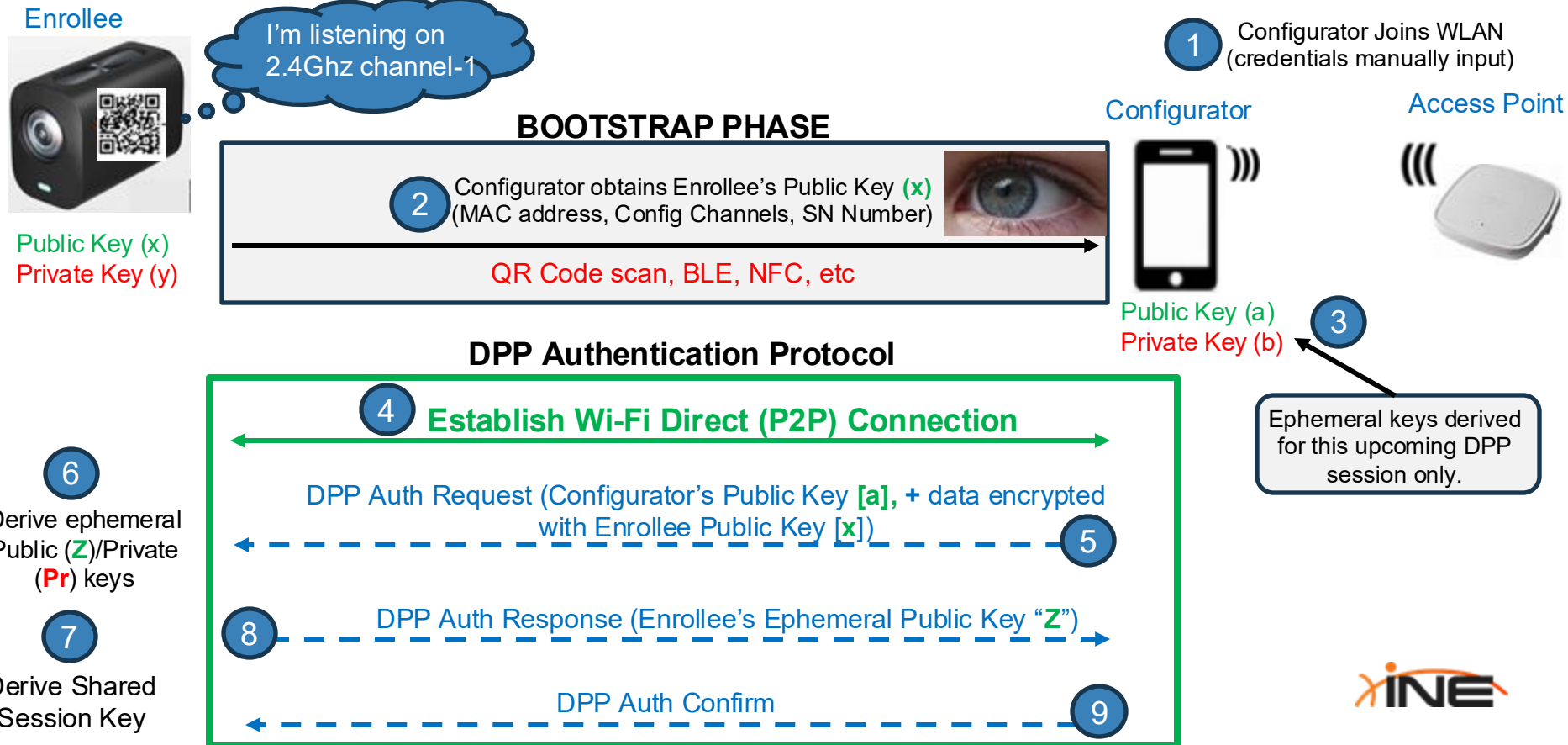
- + Allows headless clients (“*Enrollee*”) to present their public key (and other information) via an offline channel to an intermediary device that contains the Wi-Fi credentials (“*Configurator*”).
- + Easy Connect doesn't mandate HOW this is done but common methods include:
 - + QR Codes
 - + Bluetooth Low Energy (BLE)
 - + NFC tags



DPP Stages

- + The Device Provisioning Protocol is applied in two stages across two different DPP protocols
- + DPP Authentication Protocol
 - + Exchange of public keys
 - + Creation of a secure, encrypted tunnel
- + DPP Configuration Protocol
 - + Configurator transmits required Wi-Fi information to Enrollee

DPP Authentication Stage

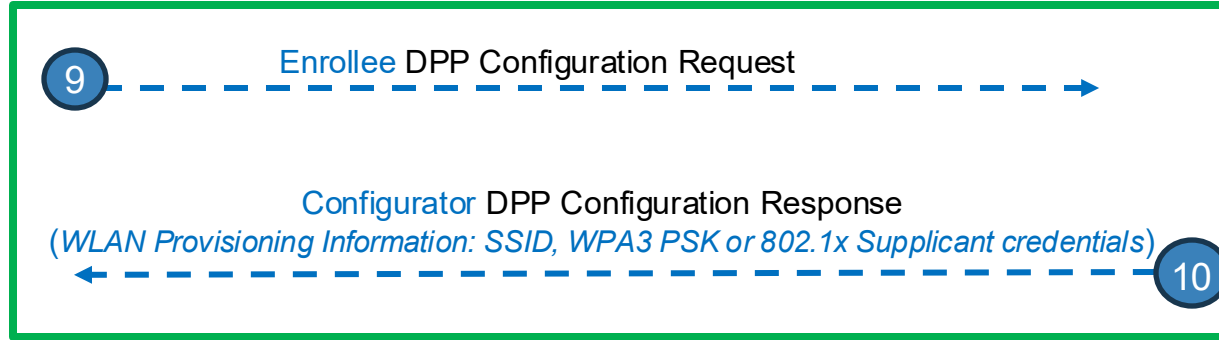


DPP Configuration Stage

Enrollee



DPP Configuration Protocol



Configurator



Public Key (a)
Private Key (b)



12 Enrollee Joins the WLAN



Access Point



Easy Connect Final Thoughts

- + Once provisioning completes, the Enrollee uses the delivered credentials to join the WPA3-Personal network via the standard SAE handshake
- + WPA3-Enterprise connections are also possible (though not widely implemented yet)
- + Not all devices support DPP yet; adoption began around 2019 (11ax chipsets onward)



**Thank you for
watching!!**

Getting Personal with WPA3

Wi-Fi Security

- Summary

Key Concepts - Recap

- + Wi-Fi Key Management and Cryptographic Foundations
- + Evolution of Wi-Fi Security Protocols (WPA2 → WPA3)
- + Protecting Management and Open-Network Communications
- + WPA3 Enhancements



Learning Outcomes Recap

- + Explain the Complete Wi-Fi Key Hierarchy and Its Functions
- + Differentiate Between WPA2 (PSK) and WPA3 (SAE) Key-Derivation Processes
- + Analyze the Importance and Operation of WPA3 Protected Management Frames (PMF) & Beacon Protection
- + Configure and Validate Modern Wi-Fi Security Enhancements

Next Steps

- + Check out some other Wireless content from INE such as:

**Getting There with Wireless
Mobility & Roaming**

**Fast-Track WLAN Deployment
Using Cisco Catalyst Center**

**Wireless Without the Headache:
Starting Out with the 9800 WLC**



THANKS FOR WATCHING!



EXPERTS AT MAKING YOU AN EXPERT

