STRUCTURED MESH™

SECURITY PRIMER

BACKHAUL SECURITY

- AES-CCMP ENCRYPTION
  - Implemented in hardware by chipset
- 128 bit TEMPORAL KEY
  - Generated during link formation
  - Key distribution via secure AES channel
  - Every backhaul link uses different key
  - Periodically updated

BACKHAUL SECURITY

CHILD NODE

PARENT NODE

802.11 ASSOCIATION REQUEST CONTAINING MESHDYNAMICS I.E

802.11 ASSOCIATION RESPONSE CONTAINING ENCRYPTED TEMPORAL AES-CCMP KEY

- ALL NODES IN A MESH NETWORK SHARE A 128-bit AES KEY
  - ENCRYPTS MESHDYNAMICS CONTROL PACKETS
- 128-bit AES-CCMP KEY ENCRYPTED USING SHARED AES KEY

ACCESS SECURITY

- USES THE 802.11i STANDARD
  - a.k.a. WPA version 2
  - Hardware based Encryption/Decryption
- LEGACY 40-bit and 64-bit WEP ALSO SUPPORTED
- CLIENT AUTHENTICATION USING 802.1x
  - RADIUS (RFC 3139) supported as backend

ACCESS SECURITY

- WPA PERSONAL
  - 802.1x authentication not used
  - 256-bit master key shared between AP and clients (a.k.a. PSK [pre-shared key])
- WPA ENTERPRISE
  - No keys shared
  - 802.1x authentication generates 256-bit master key (a.k.a. PMK [pairwise master key])

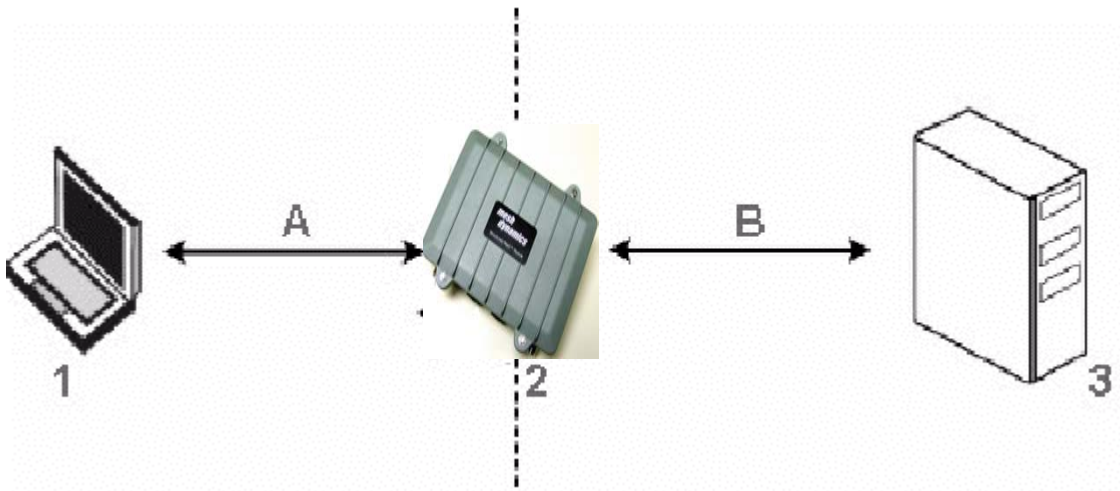
802.1x OPERATION

- Definitions:

SUPPLICANT – the client (1)

AUTHENTICATOR – the AP (2)

AUTHENTICATION SERVER – RADIUS server (3)



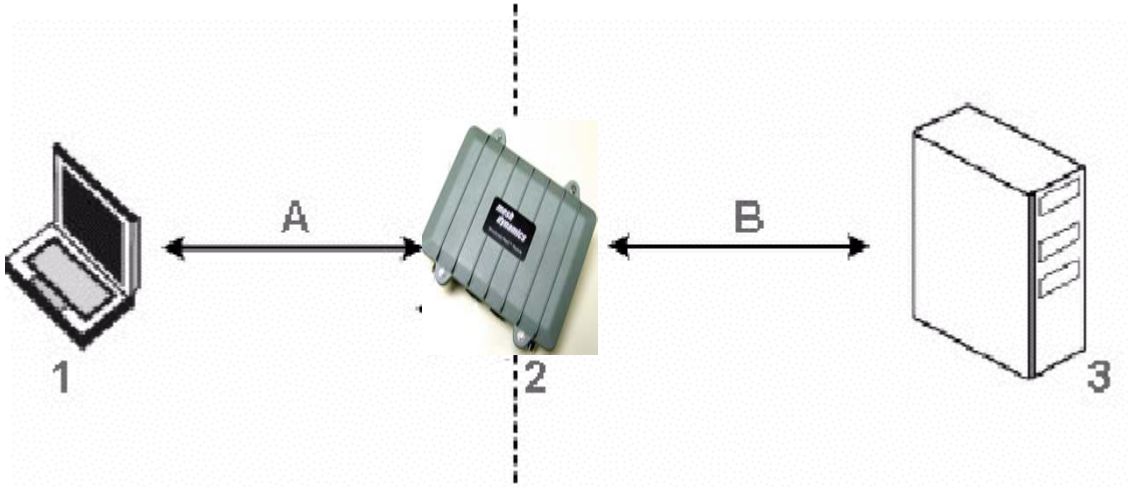
EXTENSIBLE AUTHENTICATION PROTOCOL

- DEFINED IN RFC 3748
- MULTIPLE AUTHENTICATION METHODS
- RUNS DIRECTLY OVER PPP OR IEEE 802
- INDEPENDENT OF LAYER 3

EAP METHODS

- PEAP (Protected EAP)
  - Uses a digital certificate on the network side, and a password or certificate on the client side
  - Provides for mutual authentication
  - Supported by Windows 2000 (SP4 or later), Windows XP, many 3rd-party software packages
- EAP-TLS (EAP with Transport Level Security)
  - RFC 2716
  - Uses certificates on both the client and network side
  - Provides for mutual authentication
  - Supported by Windows 2000, XP, and 3rd-party packages
- EAP-TTLS (EAP with Tunneled Transport Layer Security)
  - Uses a certificate on the network side, and a password, token, or certificate on the client side
  - Provides for mutual authentication
  - Encrypts entire exchange – including the username
  - Supported by 3rd-party applications, but not by Microsoft directly

802.1x OPERATION



- Supplicant communicates with authentication server *through* authenticator
- Protocol used is EAP
- Authenticator will re-package EAP messages from client inside a RADIUS exchange, forward to authentication server
- Authentication either succeeds or fails – RADIUS message will indicate which
- Details of exchange are encrypted and thus hidden from the AP

EAP OPERATION

- In wireless networks, EAP exchange is accomplished using the ***EAPOL* protocol (**EAP over LAN**)** – EAPOL is defined as part of the 802.1x specification

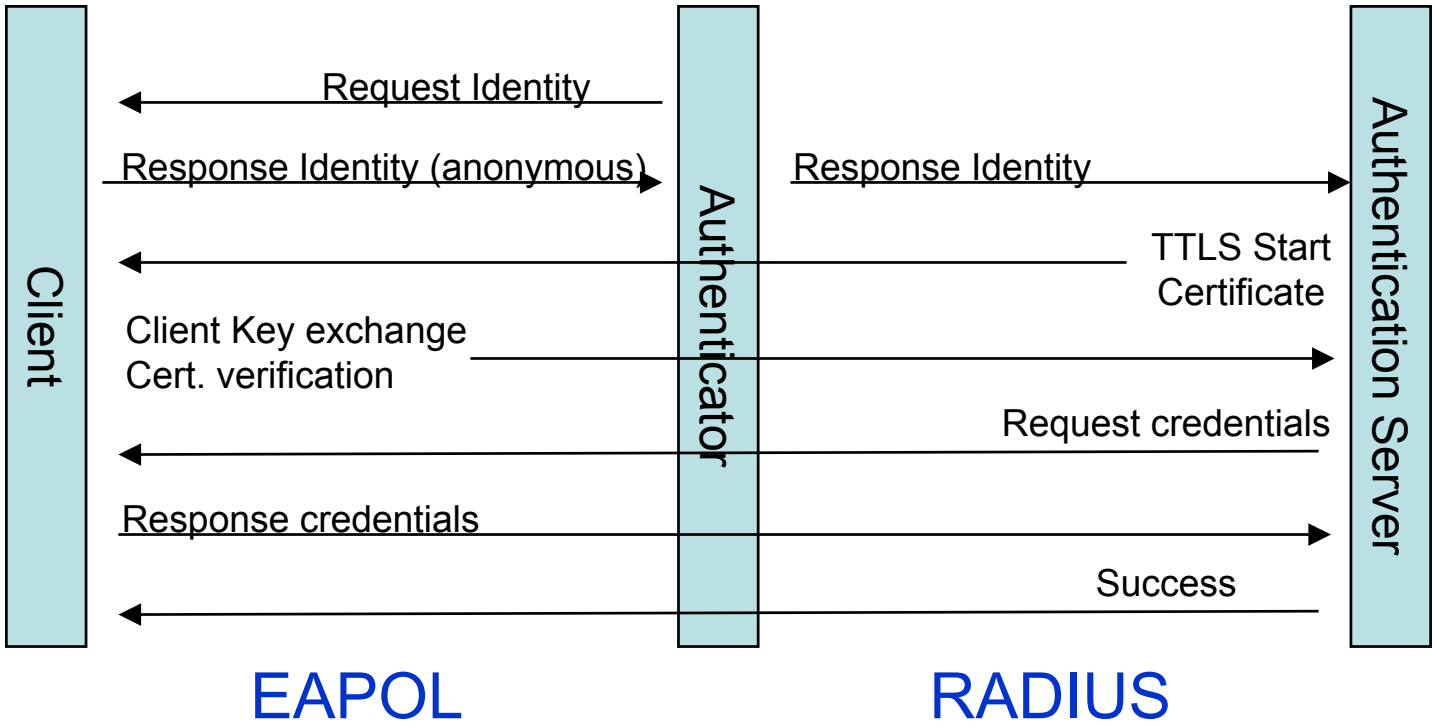
- EAPOL runs between the client and the AP

- EAPOL messages are converted to RADIUS messages on the back end for communication between the AP and the authentication server

EAP-TTLS EXAMPLE

- Authentication server contains a certificate
  - This is the same concept as SSL – the certificate must be trusted by the client, or signed by a certificate authority that is trusted by the client
  - The certificate is the AP's authentication credential. If the AP has the wrong certificate, the client should drop communication
- Client usually contains no information – relies on the user entering a password
  - The username/password is the client's authentication credential. If the username/password is wrong, the AP should drop communication
  - Client may also use a certificate, token card, etc.

EAP-TTLS EXAMPLE



- 2-stage process
  - Outer tunnel establishment
  - Credential exchange happens inside the encrypted tunnel

PMK GENERATION

- User authenticates using 802.1x
  - The EAP type does not matter – the end result is the same
  - Authentication exchange takes place between the supplicant (client) and the authentication server (RADIUS) – the AP is merely a pass-through device and "gatekeeper" at this point
- After authentication succeeds, RADIUS server derives a large random number. **Independently**, the client also derives the same large random number.
- This large random number is called the "Pairwise Master Key" (PMK)
- RADIUS server passes the PMK to the 802.1x authenticator (AP) – there is risk here!
  - The value is passed via a RADIUS attribute

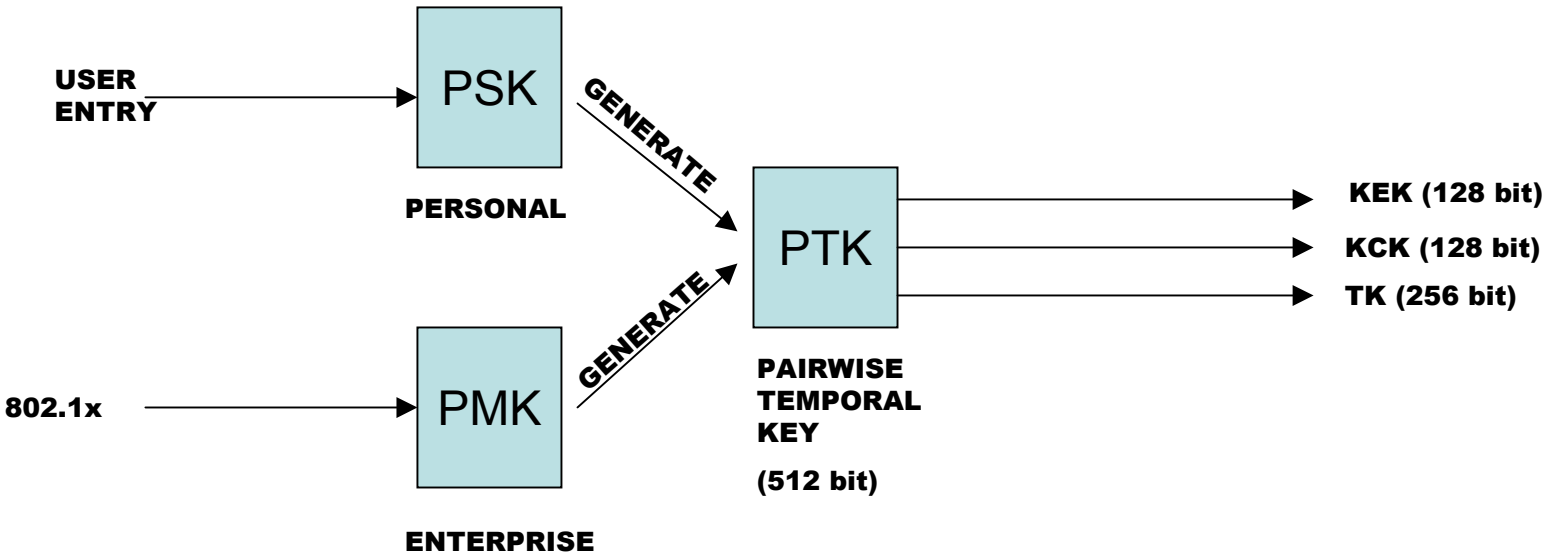
802.1x SUMMARY

- The AS and STA have established a session
- The AS and STA possess a mutually authenticated Master Key
  - Master Key represents decision to grant access based on authentication
- STA and AS have derived PMK
  - PMK is an authorization token to enforce access control decision
- AS has distributed PMK to the AP

WPA

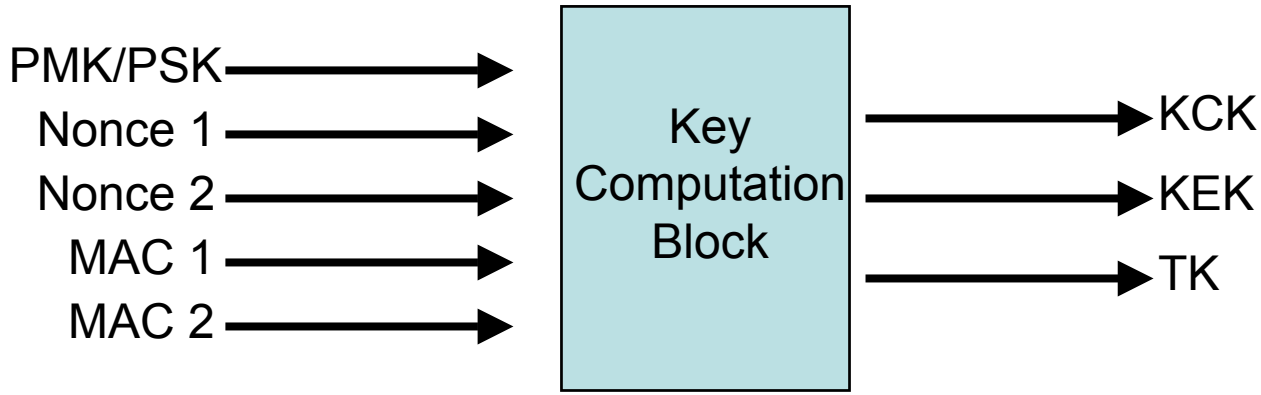
- At this stage, the client and AP both have the PMK. The goal now is for both sides to derive the PTK so that data encryption can begin.
- In addition to deriving the PTK, both sides need to prove to each other that they know the PMK
  - If this didn't happen, it's possible that one side could have a different key – data encryption would not work and network connectivity would not exist
- The process of deriving the PTK and proving knowledge of the PMK to the other side is known as the **4-way handshake**

WPA



- KEK : KEY ENCRYPTION KEY
- KCK : KEY CONFIRMATION KEY
- TK : TEMPORAL KEY

PTK GENERATION



- A “nonce” is simply a random number – usually based on something “timely” or “live” such as the time of day
  - In theory, a nonce is a value that will be used only once and will never be generated again
- Nonce 1 is generated by the AP. Nonce 2 is generated by the client.

4-WAY HANDSHAKE

- Message 1: AP → Client

  An EAPOL message is sent that contains Nonce 1. This message is not encrypted or integrity-verified.

  After this message is sent, the client has all the pieces necessary to generate the PTK.

4-WAY HANDSHAKE

- Message 2: Client→AP

  An EAPOL message is sent that contains Nonce 2. This message is not encrypted, but contains a MIC (Message Integrity Check) calculated using the KCK.

  After this message is sent, the AP can now compute the PTK. It does so, and then verifies the MIC using the newly-generated KCK.

  If this step succeeds, the AP has verified that the client has the correct PMK and PTK, and that there is no **man-in-the-middle**.

4-WAY HANDSHAKE

- Message 3: AP → Client

  This message tells the client that the AP is ready to begin encryption. The message is not encrypted, but contains a MIC (Message Integrity Check) calculated using the KCK.

  If this step succeeds, the client has verified that the AP has the correct PMK and PTK and that there is no **man-in-the-middle**.

4-WAY HANDSHAKE

- Client →AP

  Finishes the handshake. Indicates that the client is ready to begin data encryption.

  After this message is sent, both sides install the PTK and begin using it for data encryption

GROUP KEYS

- The pairwise keys are used for unicast communication
- Encrypting the same message separately for each client would be inefficient, hence multicast and broadcast traffic uses a *group key*
- Group key hierarchy is similar to pairwise key hierarchy
  - GMK (Group Master Key)
  - GTK (Group Transient Key)

GROUP KEYS

- Group keys are derived *after* pairwise keys. This makes distribution much easier
- AP generates a 256-bit GMK based on random numbers
- AP derives GTK
- After each client joins the network, AP communicates the GTK to each client using the pairwise keys
- If group key changes, AP just creates a new one and distributes it again

VLANs

- Multiple SSIDs
  - Each VLAN can have its own unique SSID
  - Each VLAN can have 802.1q packet tagging
  - Helps backhaul traffic distribution
- Multiple Security profiles (one per VLAN)
  - WEP,PSK,802.1x defined per VLAN
  - Mixture of open, WEP, WPA networks possible using same device