# Wireless LAN Security and IEEE 802.11i*

Jyh-Cheng Chen[1,2], Ming-Chia Jiang[1], and Yi-Wen Liu[1]
[1]Department of Computer Science
[2]Institute of Communications Engineering
National Tsing Hua University
Hsinchu, Taiwan

**Abstract**

This paper reviews wireless LAN security with a focus on the evolving new standard of IEEE 802.11i. The major security enhancements in encryption and authentication defining by the IEEE 802.11i are illustrated. In addition, the newly introduced key management in IEEE 802.11i is also discussed. Because IEEE 802.11i incorporates IEEE 802.1X as its authentication enhancement, the IEEE 802.1X with the consideration of roaming users are depicted. Both intra-subnet roaming and inter-subnet roaming are illustrated.

**Keywords:** 802.11i, 802.1X, authentication, encryption, key management, wireless security, wireless LANs

**Contact Author:**
Prof. Jyh-Cheng Chen
Department of Computer Science
National Tsing Hua University
Hsinchu 300, Taiwan

Phone: +886-3-574-2961
Fax: +886-3-572-3694
Email: jcchen@cs.nthu.edu.tw

# 1 Introduction

Conventional Internet users have been bound to wired connections. Wireless communications, however, have broken this restriction and provide ubiquitous accessing to the Internet. In addition, the increase in flexibility also strongly motivates wireless network technologies. Today, the deployment of Wireless Local Area Networks (WLANs) sometimes is even more economical and efficient than installing wired networks in a whole building. With the promotion of wireless networking technologies and market, services and applications are increasing tremendously. The IEEE 802.11 standard [1] for WLANs is one of the most widely adopted standards for broadband wireless Internet access.

However, the security consideration over wireless environment is more complicated than that in wired environment. Due to the wide-open nature of wireless radio, many attacks could make the network insecure. The IEEE 802.11 standard has defined the following two basic security mechanisms for securing the access to IEEE 802.11 networks: (1) entity authentication including *open system* authentication and *shared key* authentication, and (2) Wired Equivalent Privacy (WEP). Nevertheless they are all proved to be vulnerable.

In order to enhance security of the IEEE 802.11 standard, a new standard called IEEE 802.11i [2] is being developing. The objective of IEEE 802.11i is to enhance the IEEE 802.11 security aspects. In addition to introducing *key management and establishment*, it also defines *encryption* and *authentication* improvements. In order to manage security keys automatically, the IEEE 802.11i defines key management and establishment algorithms, which are first introduced in the IEEE 802.11 standards. As conventional WEP is known to be vulnerable, the IEEE 802.11i is specifying the enhanced encryption to provide stronger privacy. IEEE 802.11i also incorporates the IEEE 802.1X [3] as its authentication enhancement. The IEEE 802.1X is now widely deployed in many IEEE 802 series standards with the Remote Authentication Dial In User Service (RADIUS, IETF RFC 2865) as the authentication server. RADIUS could provide Authentication, Authorization, Accounting (AAA) services, but it is still unlikely to resolve all security threats in wireless networks. Diameter (IETF RFC 3588) thus is evolving to improve RADIUS to provide better security.

This paper first discusses the IEEE 802.1X with the consideration of roaming users. It then reviews the (1) authentication enhancement, (2) key management and establishment, and (3) encryption enhancement defined in the IEEE 802.11i draft[†].

# 2 IEEE 802.1X

The IEEE 802.1X standard defines a mechanism for port-based network access control to provide compatible authentication and authorization mechanisms for devices interconnected by various IEEE 802 LANs. It could also be used to distribute security keys for IEEE 802.11 WLANs by enabling public key authentication and encryption between Access Points (APs) and Mobile Nodes (MNs). In IEEE 802.1X, the *port* represents the association between MN and AP. There are three main components in the IEEE 802.1X authentication

[†]IEEE 802.11i is still evolving and only draft was available when this paper was completed.

system: *Supplicant*, *Authenticator* and *Authentication Server (AS)*. Supplicant usually is a MN requesting for WLAN access. Authenticator represents the Network Access Server (NAS). In IEEE 802.11 networks it is normally an AP. A RADIUS server is commonly used as the authentication server although other type of AAA servers such as Diameter could also serve as the authentication server. In IEEE 802.11, the authentication server might be physically integrated into an AP.

### 2.1 IEEE 802.1X Framework

As indicated in Fig. 1 [3], both supplicant and authenticator have a PAE (Port Access Entity) that operates the algorithms and protocols associated with the authentication mechanisms. The authenticator PAE controls the authorized/unauthorized state of its *Controlled Port* depending on the outcome of the authentication processes. Before the supplicant is authenticated, the authenticator uses the *Uncontrolled Port* to communicate with the supplicant PAE. The authenticator will block all traffic except IEEE 802.1X messages before the supplicant is authenticated. The IEEE 802.1X standard leverages Extensible Authentication Protocol (EAP, IETF RFC 2284) to provide a number of authentication schemes, including Message Digest 5 (MD5, IETF RFC 1321), Transport Layer Security (TLS, IETF RFC 2716), Tunneled TLS (TTLS) [4], Protected Extensible Authentication Protocol (PEAP) [5], and smart cards such as EAP SIM [6]. IEEE 802.1X also defines EAPOL (EAP over LANs) that encapsulates EAP messages between the supplicant and the authenticator. EAP messages from the supplicant are relayed to the authentication server by the authenticator PAE. In order to let the RADIUS server authenticate users by using EAP, the authenticator PAE encapsulates the same EAP messages in a RADIUS packet format and sends them to the RADIUS server, assuming the RADIUS is adopted as the authentication server. The encapsulation is known as RADIUS-encapsulated EAP with the EAP-Message attribute, which is defined in RADIUS Extensions (IETF RFC 2869) for supporting EAP within RADIUS. Once the supplicant is authenticated successfully, the *Controlled Port* in the authenticator is authorized. Packets from the supplicant will now go through the *Controlled Port* of the authenticator to backend networks to acquire the necessary services.

Fig. 2 depicts a typical IEEE 802.1X message exchange with both the supplicant PAE and the authenticator PAE state transitions. The supplicant PAE and the authenticator PAE state machines are showed in Fig. 3 and Fig. 4, respectively. In Fig. 2, the digits associated with each flow do not represent the order of the flow. Instead the digits in rectangles refer to the supplicant PAE state in Fig. 3, and the digits in circles refer to the authenticator PAE state in Fig. 4.

After MN and AP complete the IEEE 802.11 association, both MN and AP will transit to the CONNECTING state in their PAE state machines. However the port is unauthorized at this moment, and the IEEE 802.1X authentication process just starts. As indicated in Fig. 2, MN (supplicant) sends an EAPOL-Start frame to the AP (authenticator) to initialize the authentication process. When the AP receives EAPOL-Start, it replies an EAP-Request/Identity to obtain MN's identity. MN transits to the ACQUIRED state when it receives EAP-Request/Identity from the AP. The MN then sends back an EAP-Response/Identity containing the MN's identity in response to the EAP-Request/Identity. If the AP receives the EAP-Response/Identity,

the authenticator PAE state will transit to the AUTHENTICATING state. In AUTHENTICATING state, the authenticator PAE encapsulates the EAP-Response/Identity message in RADIUS-Access-Request as an attribute (EAP-Message attribute) and sends it to the RADIUS server. In response to the RADIUS-Access-Request, the RADIUS server will challenge the MN by sending RADIUS-Access-Challenge to the AP, which then relays the message in the form of EAP-Request/Auth to the MN. When the MN receives EAP-Request/Auth, the supplicant PAE state transits to the AUTHENTICATING state and replies an EAP-Response/Auth, which is also relayed to the RADIUS server by the AP in the format of RADIUS-Access-Request. Depending on the authentication scheme, there might be some more message exchanges. The RADIUS server then determines whether the MN should be accepted or denied to access the network services. Fig. 2 depicts three cases thereafter which are separated by dotted-lines. If the authentication succeeds, the RADIUS server sends a RADIUS-Access-Accept to the AP. On receipt of RADIUS-Access-Accept the authenticator PAE state transits to the AUTHENTICATED state and sends an EAP-Success message to the MN to indicate the success of authentication. The *Controlled Port* of the AP shown in Fig. 1 thus is authorized. After receiving EAP-Success, the MN transits to the AUTHENTICATED state and the whole authentication process is completed. On the other hand, RADIUS-Access-Reject is sent by the RADIUS server and is relayed to the MN by the AP in the message of EAP-Failure if the authentication fails. In this case, both AP and MN transit to the HELD state, and the whole authentication fails. The *Controlled Port* thus is still unauthorized. If the MN is authenticated and it wants to perform a logoff procedure from current AP, the MN originates an EAPOL-Logoff packet to the AP. After that, the *Controlled Port* of current AP transits to unauthorized state immediately. The supplicant and authenticator will transit to the LOGOFF state and DISCONNECTED state, respectively.

Usually there are several APs connecting to a same authentication server. Because user profiles are stored in a centralized database, APs in the system could lighten the load of storing large amount of data such as MAC addresses, user names, and passwords. Administrators would not need to configure APs frequently when users are added or removed from the system. In addition, proxy RADIUS is also able to relay authentication messages to other RADIUS servers to authenticate the users whose profiles are maintained in other servers.

## 2.2 IEEE 802.1X with Roaming Users

This section discusses the mobility issues in IEEE 802.1X enabled networks in terms of intra-subnet roaming and inter-subnet roaming, respectively. In intra-subnet roaming, MN hands off to a different AP but still within the same IP subnet. It would need to re-authenticate with the new AP. Because MN is still in the same IP subnet, it does not involve the change of IP address. The inter-subnet handoff, however, would need to get a new IP address and keep the ongoing IP connections alive. This paper uses Mobile IP (IETF RFC 3344) as an example to illustrate the inter-subnet handoff in IEEE 802.1X networks.

- **Intra-subnet roaming**
  According to IEEE 802.1X standard, a MN should re-authenticate with new authenticator or authen-

tication server when it roams to another IEEE 802.1X enabled network. Fig. 5 depicts a typical architecture in which a MN moves from a Home AP to a Foreign AP within the same IP subnet. Fig. 5 also indicates the state transitions of supplicant and authenticator during the handoff.

Assuming the MN is already authenticated successfully with Home AP, therefore both supplicant and authenticator PAE states are in AUTHENTICATED state as indicated in Fig. 5. When MN roams to Foreign AP, it first proceeds the IEEE 802.11 reassociation process with the Foreign AP. The PAE state machine of the Foreign AP will transit to CONNECTING state once the reassociation is successfully completed. The Foreign AP then sends an EAP-Request/Identity to the MN. After receiving the EAP-Request/Identity, the MN will transit from AUTHENTICATED state to ACQUIRED state. It then sends back an EAP-Response/Identity. Afterwards, the message exchange follows the standard IEEE 802.1X authentication described earlier. Both Foreign AP and MN will proceed to AUTHENTICATED state eventually if the MN is authenticated successfully. The re-authentication then is completed.

- **Inter-subnet roaming**

  When roaming to a new IP subnet, the MN's IP address in old subnet is invalid in the new subnet. With only the IEEE 802.1X re-authentication, MN is able to reassociate with the Foreign AP but it cannot connect to the new IP subnet. Therefore, protocol such as Mobile IP should be used to support mobility management in IP layer. After the IEEE 802.1X authentication with Foreign AP described earlier is completed, both MN and AP are in AUTHENTICATED state. Therefore, MN is able to send Mobile IP registration messages and perform subsequent Mobile IP procedures. Basically, the IEEE 802.1X and Mobile IP could work independently.

### 2.3 IEEE 802.1X with Diameter

Because the network entities have increased in complexity, the deficiencies in RADIUS have been identified (IETF RFC 3127). Because IEEE 802.1X does not mandate the type of authentication server, Diameter (IETF RFC 3588) could be used to improve security weakness in RADIUS.

Diameter provides a base protocol that can be extended from various applications to support AAA services. The Diameter-EAP application [7] could be employed for an IEEE 802.1X network. The usage of Diameter in IEEE 802.1X system is similar to that with RADIUS. The major difference is in the replacement of RADIUS messages with Diameter messages. For inter-subnet roaming, Diameter also specifies a Mobile IPv4 application [8].

## 3 IEEE 802.11i

The IEEE 802.11 Working Group has been working on MAC enhancement for several years. In May 2001, the MAC enhancement was split into different task groups. The Task Group E (TGe) is responsible for Quality of Service (QoS). The Task Group I (TGi) is working on security.

One of the major missions of IEEE 802.11 TGi is to define a Robust Security Network (RSN). The definition of RSN according to IEEE 802.11i draft [2] is *a Security Network which only allows the creation of Robust Security Network Associations (RSNA)*. That is, in a RSN the associations between all stations including Access Points (APs) are built upon a strong association/authentication called RSNA, which is also defined by the IEEE 802.11 TGi as: *a RSNA depends upon IEEE 802.1X to transport its authentication services and to deliver key management services*. A security association refers to as *the context providing the state (cryptographic keys, counters, sequence spaces, etc.) needed for correct operation of the IEEE 802.11 cipher suites.* RSNA includes a novel *4-way handshake* mechanism to provide robust session key management. By leveraging IEEE 802.1X, the 4-way handshake, and the enhanced cryptographic algorithms, communication links in the IEEE 802.11 wireless are securely protected.

### 3.1 IEEE 802.11i Framework

The IEEE 802.11i standard defines two classes of security framework for IEEE 802.11 WLANs: RSN and pre-RSN security frameworks. A station is called a RSN-capable equipment if it is capable of creating RSNAs. Otherwise, it is a pre-RSN equipment. The network that only allows RSNA in associations with RSN-capable equipments is called a RSN security framework. The network that allows pre-RSNA associations between stations is called a pre-RSN security framework. The major difference between RSNA and pre-RSNA is in the 4-way handshake. If the 4-way handshake is not included in the authentication/association procedures, stations are said to use pre-RSNA.

- **Pre-RSN**

  The pre-RSN security consists of two security subsystems: (1) IEEE 802.11 entity authentication, and (2) WEP. The IEEE 802.11 entity authentication includes *Open System* authentication and *Shared Key* authentication. In open system authentication, there is no authentication algorithm. A station is authenticated simply based on its identity. Shared key authentication, on the other hand, authenticates a station based on a secret key known to authentication *requester* and *responder*. It requires the privacy mechanism implemented in WEP.

- **RSN**

  In addition to enhancing the security in pre-RSN, the RSN security defines key management procedures for IEEE 802.11 networks. It also enhances the authentication and encryption in pre-RSN.

  – **Authentication Enhancement:** IEEE 802.11i utilizes IEEE 802.1X for its authentication and key management services. It incorporates two components into the IEEE 802.11 architecture: *IEEE 802.1X Port* and *Authentication Server (AS)*. IEEE 802.1X port represents the association between two peers. There is a one-to-one mapping between IEEE 802.1X Port and association.

As discussed earlier, IEEE 802.1X port will allow general traffic to pass only when the authentication is successfully completed. The AS could be a stand-alone server or it could be integrated into AP. Although the protocol between AS and AP is not recommended by IEEE 802.11i, there should be a secure channel such as TLS (IETF RFC 2246) or IPsec (IETF RFC 2401) between AP and AS. EAP that supports mutual authentication should be used in RSN. That is, the authentication requester and responder must be able to authenticate each other. EAP-MD5, for instance, cannot meet this requirement.

– **Key Management and Establishment:** Two ways to support key distribution are introduced in IEEE 802.11i: *manual key management* and *automatic key management*. Manual key management requires the administrator to manually configure the key. The automatic key management is available only in RSNA. It relies on IEEE 802.1X to support key management services. More specifically, the 4-way handshake is used to establish each transient key for packet transmission.

– **Encryption Enhancement:** In order to enhance confidentiality, two advanced cryptographic algorithms are developed: Counter-Mode/CBC-MAC Protocol (CCMP) and Temporal Key Integrity Protocol (TKIP). In RSN, CCMP is mandatory. TKIP is optional and is recommended only to patch pre-RSNA equipment.

IEEE 802.11i specifies *RSN Information Element (RSN IE)* which carries RSN security information including RSN capabilities, authentication and cipher key selectors. RSN IE could be used to distinguish pre-RSN stations and RSN-capable stations. RSN-capable stations shall include the RSN IE in beacons, probe response, association and reassociation request, and in the second and third messages of the 4-way handshake. On the other hand, there is no RSN IE in messages sent by pre-RSN stations. As shown in Fig. 6, the RSN IE contains a list of authentication and cipher selector fields for communications. The value of *Element ID* field in Fig. 6 should always be $48$ in decimal. The *Length* field indicates the number of octets in the information fields excluding the *Element ID* and *Length* fields. *Version* field shows the version number of the RSNA protocol. The *Pairwise Key Cipher Suite Count* indicates the number of Pairwise Key Cipher Suites that are contained in the field of *Pairwise Key Cipher Suite List*. The *Pairwise* refers to two entities that are associated with each other. The *Pairwise Key Cipher Suite*, therefore, is the cipher suite which is being or to be associated between communicating peers. Similarly, the *Authentication and Key Management Suite Count* indicates the number of Authentication and Key Management Suites that are contained in the *Authentication and Key Management Suite List* field. In the *RSN Capabilities* field, the requested or advertised capabilities are filled in. By using this field, the receiver could know the security mechanisms the sender supports or is requesting.

Generally speaking, the RSN IE carries the robust security information that indicates the authentication and cipher algorithms the communicating parties would use. Stations and APs could learn the security capabilities of the communicating peers and negotiate with each other by the RSN IE carried in association/reassociation request, probe response, beacons, or other messages. Correspondent security procedures will then be executed.

Fig. 7 shows an example to establish RSNA between supplicant (station) and authenticator (AP) in a Basic Service Set (BSS). It assumes there is no pre-shared key. *Flows 1–6* are the IEEE 802.11 association and authentication process prior to attaching to the authenticator. During the process, security information and capabilities could be negotiated by using the RSN IE. The authentication in *Flows 3 and 4* refer to the IEEE 802.11 open system authentication. After the IEEE 802.11 association is completed, the IEEE 802.1X authentication indicated in *Flow 7* of Fig. 7 is initiated. EAP messages will be exchanged between supplicant, authenticator, and authentication server, although authentication server is not depicted in Fig. 7. If the supplicant and the authentication server authenticate each other successfully, both of them independently generate a *Pairwise Master Key (PMK)*. The authentication server then transmits the PMK to the authenticator through a secure channel (for example, IPsec or TLS). The 4-way handshake then uses the PMK to derive and verify a *Pairwise Transient Key (PTK)*. Therefore, the session key between the supplicant and the authenticator is guaranteed to be fresh. After that, the group key handshake is proceeded as indicated in *Flow 9*. The group key handshake is used to generate and refresh the group key, which is shared between a group of stations and APs. By using this key, broadcast and multicast messages can be securely exchanged over the air.

The following sections review the authentication enhancement, key management and establishment, and encryption enhancement, respectively, defined in IEEE 802.11i.

### 3.2 Authentication Enhancement

In the original IEEE 802.11 standard, a station should first associate with an IEEE 802.11 AP. It then is able to access to the WLAN service. An example of the process is shown in *Flows 1–6* in Fig. 7. After finding an AP by receiving the Probe Response, the mobile station needs to proceed to the following two steps: *IEEE 802.11 entity authentication* and *association*. Before associating with an AP, the station needs to accomplish the IEEE 802.11 entity authentication. As discussed earlier, there are two authentication schemes: *open system* authentication and *shared key* authentication. The open system authentication allows a station to be authenticated without having a correct WEP key. There are two-message exchanges. The first message sending from supplicant (mobile station) to authenticator (AP) is used to expose the identity of the station. Based on the identity, the authentication result is sent from the authenticator back to the station. There is no authentication algorithm. In shared key authentication, there are four-message exchanges. The first message containing the identity of the station is delivered from the station to the AP. The AP will then send a challenge packet to the mobile station. The mobile station is required to encrypt the challenge packet using the shared WEP key and send the encrypted result back to the AP. If the challenge packet is encrypted correctly, the supplicant is authenticated successfully. The authentication result is sent to the station in the fourth message. If the station is authenticated successfully, it proceeds to the IEEE 802.11 association. The mobile station should transmit an Association Request to the AP. The AP then sends back an Association Reply to the station.

The shared key authentication in IEEE 802.11 is not adopted by the IEEE 802.11i. Instead, it incorpo-

rates IEEE 802.1X standard as the authentication solution for RSN. As depicted in Fig. 7, IEEE 802.1X is performed after IEEE 802.11 open system authentication and association. IEEE 802.1X provides a *port-based* network access control mechanism to protect against unauthorized access. Details of IEEE 802.1X have been discussed. Please note Fig. 7 depicts an establishment of RSN. The two-message exchanges of *Flows 3 and 4* for open system authentication should not be replaced by the four-message exchanges of the shared key authentication.

The IEEE 802.11i also specifies a more robust security framework by utilizing IEEE 802.1X, 4-way handshake, and group key handshake to authenticate and authorize stations. The 4-way handshake and group key handshake will be described in next session. After the station is authenticated successfully, the cryptographic keys are configured as well. The station, therefore, is able to send and receive unicast and broadcast frames in a secure manner. Moreover, IEEE 802.11i also supports pre-authentication. A station could pre-authenticate with an AP before roaming. A station could initiate EAPOL-Start message through the serving AP to inform the new AP to start the IEEE 802.1X authentication. Therefore, the handoff latency could be reduced.

### 3.3 Key Management and Establishment

This section discusses the 4-way handshake and group key handshake, respectively.

- **4-Way Handshake**

  RSNA defines a 4-way handshake to perform several functions such as confirming the liveness of the communicating stations, guaranteeing the freshness of session key, installing the cryptographic key, and confirming the installation of the key. The 4-way handshake is achieved by using IEEE 802.1X. Specifically, messages exchanged in the 4-way handshake are in the *EAPOL-Key* format. EAPOL-Key is defined in IEEE 802.1X that could be used to exchange cryptographic keying information. Fig. 7 depicts the message flows of 4-way handshake.

  In the 4-way handshake, the authenticator first sends out a message to the supplicant. The first message contains key information and an *Anonce*. Anonce is a nonce, which is also called key material, generated by the authenticator. A nonce essentially is a random or pseudo-random value. In the 4-way handshake, Anonce will never be reused. Therefore, the RSN can against replay attack.

  After receiving the first message, the supplicant validates the message by checking the *Replay Counter* field in the message. The Replay Counter is a sequence number, which shall be incremented by each EAPOL-Key message. If the Replay Counter is less than or equal to the value kept in the supplicant, the supplicant discards the message. Otherwise, the supplicant generates a new nonce called *Snonce*. By using an algorithm called *Pseudo-Random Function (PRF)* with Anonce, Snonce, Pairwise Master Key (PMK), and other information as the inputs, the supplicant derives a *Pairwise Transient Key (PTK)*. The supplicant then sends back the second message containing key information, Snonce, the

supplicant's RSN IE, and the *Message Integrity Code (MIC)* back to the authenticator. The MIC is a cryptographic digest used to provide integrity service.

Upon receiving the second message, the authenticator validates the message by checking the Replay Counter. The process is similar to that in the supplicant when receiving the first message. It then derives the PTK if the second message is validated. Because the authenticator uses the same algorithm and the same inputs, the PTK derived by the authenticator will be same as the one in the supplicant. The authenticator also verifies the MIC. The packet is discarded silently if the MIC is not valid. In addition, the authenticator compares the received RSN IE bit-wise with the one contained in Association/Reassociation Request received earlier from the supplicant. If these are not exactly identical, the association is terminated. Otherwise, the authenticator sends the third message to the supplicant. The third message includes the key information, Anonce, MIC, and the authenticator's RSN IE.

On reception of the third message, the supplicant first verifies the message by checking the Replay Counter and Anonce fields. It then compares the RSN IE with the one received previously in the Beacon or Probe Response. The supplicant will disassociate from this AP if the RSN IEs are different. If the RSN IE is correct, the supplicant further checks the MIC. The supplicant sends back the fourth message to the authenticator if the MIC is valid. The fourth message comprises the key information and MIC.

Once the fourth message is received by the authenticator, the authenticator checks the Replay Counter as before. The authenticator then verifies the MIC if the Replay Counter is valid. The 4-way handshake is completed if the MIC is valid. The fourth message is used to acknowledge the authenticator that the supplicant has installed the temporal key, PTK. The PTK is only known by the supplicant, authenticator, and authentication server. It is used as a key to encrypt data.

- **Group Key Handshake**

  RSNA also defines a group key handshake for authenticator to deliver the Group Transient Key (GTK) to supplicant such that the supplicant could receive broadcast messages. Like the 4-way handshake, the messages exchanged in the group key handshake also use the EAPOL-Key format. Fig. 7 depicts the message flows of the group key handshake.

  As indicated in Fig. 7, the group key handshake is performed after the 4-way handshake. The authenticator first sends a message which includes key information, MIC, and GTK to the supplicant. The GTK is encrypted by using the EAPOL-Key Encryption Key (KEK), and the MIC is computed over the body of the EAPOL-Key message by using the EAPOL-Key Confirmation Key (KCK). Both KEK and KCK are parts of the PTK. Upon receiving the message, the supplicant checks the Replay Counter. It then uses the KCK to verify the MIC. The supplicant will decrypt the GTK with KEK if the Replay Counter and MIC are valid. The supplicant then configures the GTK into its IEEE 802.11 MAC. In addition, it also replies a message, which includes key information and MIC, back to the authenticator. Similarly, the authenticator validates the Replay Counter and the MIC.

### 3.4 Encryption Enhancement

The WEP algorithm is primarily used to protect wireless communications from eavesdropping. It is also capable of preventing unauthorized access. Thus WEP provides both confidentiality and integrity services. WEP relies on the secret key shared between mobile station and AP. WEP uses the RC4 stream cipher. Before sending a data, the sender needs to compute the Integrity Check Value (ICV) with CRC-32 algorithm. The sender then encrypts the data frame and ICV. The ciphertext consists of the encrypted data and ICV. The WEP bit in the MAC header should be set as well. When the receiver receives a MAC frame with WEP bit set, it will use the shared WEP key to decrypt the payload.

It is known that WEP has been cracked. WEP is vulnerable because of the short length of Initialization Vector (IV) and the static secret key. IVs are used to concatenate the shared secret key to produce different RC4 key sequences for each packet. The IV is generated at random and is also included in the packet. With only 24 bits of IV, WEP will eventually use the same IV for different data packets, which is known as *IV collision*. When collecting enough packets based on the same IV, an attacker could find out the shared value, i.e. the key sequence or the secret key, among the communicating parties. The static nature of the shared secret key causes another security issue. Because the original IEEE 802.11 does not provide any mechanism for key management, the system administrator and a user in general use the same shared key for a long period of time. Even the same WEP key is shared between all stations in the same Basic Service Set (BSS) or Extended Service Set (ESS). This nature provides attackers plenty of time to monitor and hack into WEP-enabled WLANs.

To amend the flaws in WEP, the IEEE 802.11i develops a better algorithm called Temporal Key Integrity Protocol (TKIP) as an interim standard. TKIP, initially referred to as WEP2, is also based on RC4 encryption. It, however, is implemented in a different way that addresses the vulnerabilities of WEP. The TKIP defines a Temporal Key (TK) which is a 128-bit secret key shared by encryptor and decryptor. The TK might be common among many parties. The encryptor and decryptor must use the RC4 stream cipher. Each party must ensure that no IV value is used more than once with current TK. The IV is expected to be implemented as a 16-bit counter starting with zero. Implementations must ensure that the TK is updated before the full 16-bit IV space is exhausted. TKIP also employs a packet *sequence counter* to order the MAC Protocol Data Unit (MPDU). The receiver should drop the out-of-order MPDUs. It thus could protect the replay attack. Moreover, TKIP combines the temporal key with the client's MAC address and then adds a relatively large 16-bit IV to produce the key to encrypt data. This ensures that each computer uses different key for encryption. TKIP basically applies the same encryption as WEP, but it utilizes the IEEE 802.1X EAPOL protocol to refresh the temporal keys to prevent key reuse. This provides a dynamic key distribution that significantly enhances the security provided by WEP. TKIP can be adapted into the older IEEE 802.11 products by just upgrading through relatively simple firmware patches. This is especially favorable for vendors. In addition, equipments that only support the old WEP will still be capable of interoperating with the TKIP-enabled devices. TKIP is optional in IEEE 802.11i.

Because TKIP uses the same RC4 encryption as WEP, it is considered as a short-term solution for

WLAN security. In addition to TKIP, the IEEE 802.11i standard also defines Counter-Mode/CBC-MAC Protocol (CCMP) as a long-term solution. The CCMP employs the stronger encryption of Advanced Encryption Algorithm (AES) which uses the CCM mode (IETF RFC 3610) with a 128-bit key and a 128-bit block size of operation. The CCM mode combines Counter-Mode (CTR) and Cipher Block Chaining Message Authentication Code (CBC-MAC). The CTR is used to encrypt the payload and the MIC to provide confidentiality service. The CBC-MAC computes the MIC to provide authentication and integrity services. The CCM requires a fresh Temporal Key (TK) for every session and needs to refresh TK when the Packet Number (PN) is repeated. The PN is incremented for each MPDU and can be used to prevent the replay attack with the receiver's Replay Counter. The PN and key Id are encoded in the CCMP header. Although the CCMP could provide much stronger security services, it requires additional hardware (co-processor) to improve encryption performance. Therefore, the older IEEE 802.11 hardware will not be upgradeable in many cases. CCMP is mandatory in IEEE 802.11i.

## 4 Summary

Due to the nature of wireless media, unauthorized access is easier than that in wired networks. Although the IEEE 802.11 is proverbial insecure, it is widely deployed. The IEEE 802.11i, therefore, is aiming to enhance the security in IEEE 802.11 networks.

This paper presents the security enhancements in encryption and authentication developed by the IEEE 802.11i. In addition, the newly introduced key management in IEEE 802.11i is also discussed. The RSN is expected to fulfill many security requirements. However, the coordination of the whole systems is still a challenge. It involves inter-compatibility between different domains, as well as backward-compatibility between new and old systems. The usability of the new software and hardware will also determine the acceptance of the new standard by end users. Although the emerging IEEE 802.11i standard potentially could improve the security services in today's IEEE 802.11 wireless LANs, it is expected that more work is still needed to lead us to a more secure WLAN environment.

## Acknowledgments

This work was sponsored in part by MOE Program for Promoting Academic Excellent of Universities under the grant number 89-E-FA04-1-4, National Science Council under the grant numbers 91-2219-E-007-023 and 92-2213-E-007-019, and Industrial Technology Research Institute under the contracts of T1-92019-3 and 2F-92050-4.

## References

[1] ANSI/IEEE Std 802.11, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," 1999.

[2] IEEE Std 802.11i/D4.1, "Wireless medium access control (MAC) and physical layer (PHY) specifications: medium access control (MAC) security enhancements," July 2003.

[3] IEEE Std 802.1X-2001, "Port-based network access control," June 2001.

[4] P. Funk and S. Blake-Wilson, "EAP tunneled TLS authentication protocol (EAP-TTLS)." IETF Internet Draft, <draft-ietf-pppext-eap-ttls-03.txt>, work in progress, Aug. 2003.

[5] A. Palekar, D. Simon, J. Salowey, H. Zhou, and S. Josefsson, "Protected EAP protocol (PEAP) version 2." IETF Internet Draft, <draft-josefsson-pppext-eap-tls-eap-07.txt>, work in progress, Oct. 2003.

[6] H. Haverinen and J. Salowey, "EAP SIM authentication." IETF Internet Draft, <draft-haverinen-pppext-eap-sim-12.txt>, work in progress, Oct. 2003.

[7] P. Eronen, T. Hiller, and G. Zorn, "Diameter extensible authentication protocol (EAP) application." IETF Internet Draft, <draft-ietf-aaa-eap-04.txt>, work in progress, Feb. 2004.

[8] P. R. Calhoun, T. Johansson, C. E. Perkins, and T. Hiller, "Diameter Mobile IPv4 application." IETF Internet Draft, <draft-ietf-aaa-diameter-mobileip-16.txt>, work in progress, Feb. 2004.

## Biographies

Jyh-Cheng Chen [SM] (jcchen@cs.nthu.edu.tw) is an Associate Professor in the Department of Computer Science and the Institute of Communications Engineering, National Tsing Hua University, Hsinchu, Taiwan. Prior to joining National Tsing Hua University as an Assistant Professor, he was a Research Scientist at Telcordia Technologies (formerly Bellcore), Morristown, NJ, from August 1998 to August 2001. He received his Ph.D. degree from the State University of New York at Buffalo in 1998. Dr. Chen is coauthor of the book *IP-Based Next-Generation Wireless Networks* published by Wiley in January 2004.

Ming-Chia Jiang (jmc@wire.cs.nthu.edu.tw) received his B.S. degree in Computer Science and Information Engineering from the National Central University, Chungli, Taiwan in 2001, and M.S. degree in Computer Science from the National Tsing Hua University, Hsinchu, Taiwan in 2003. He is now with Ambit Mircosystems Corporation in Hsinchu, Taiwan.

Yi-Wen Liu (timl@wire.cs.nthu.edu.tw) received his B.S. degree from the Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan in 2002. He is now a MS student in the same department.

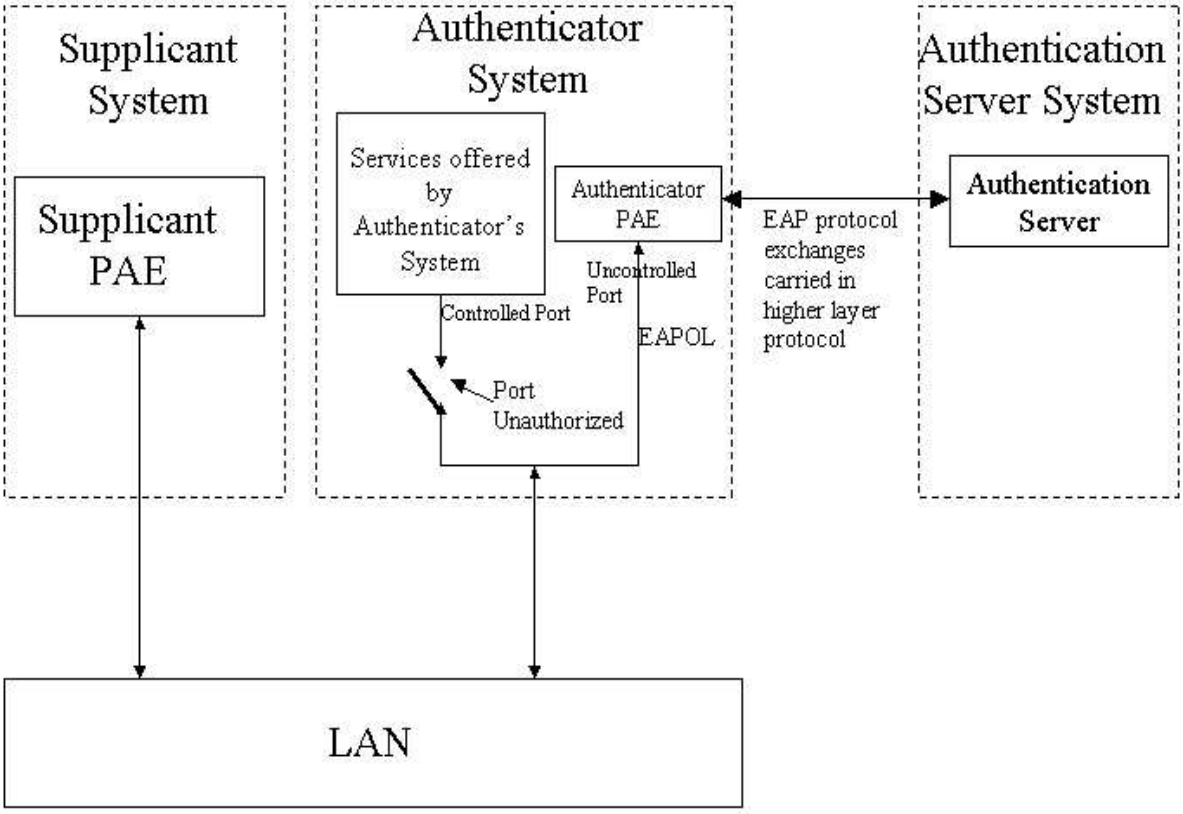


Figure 1: IEEE 802.1X framework

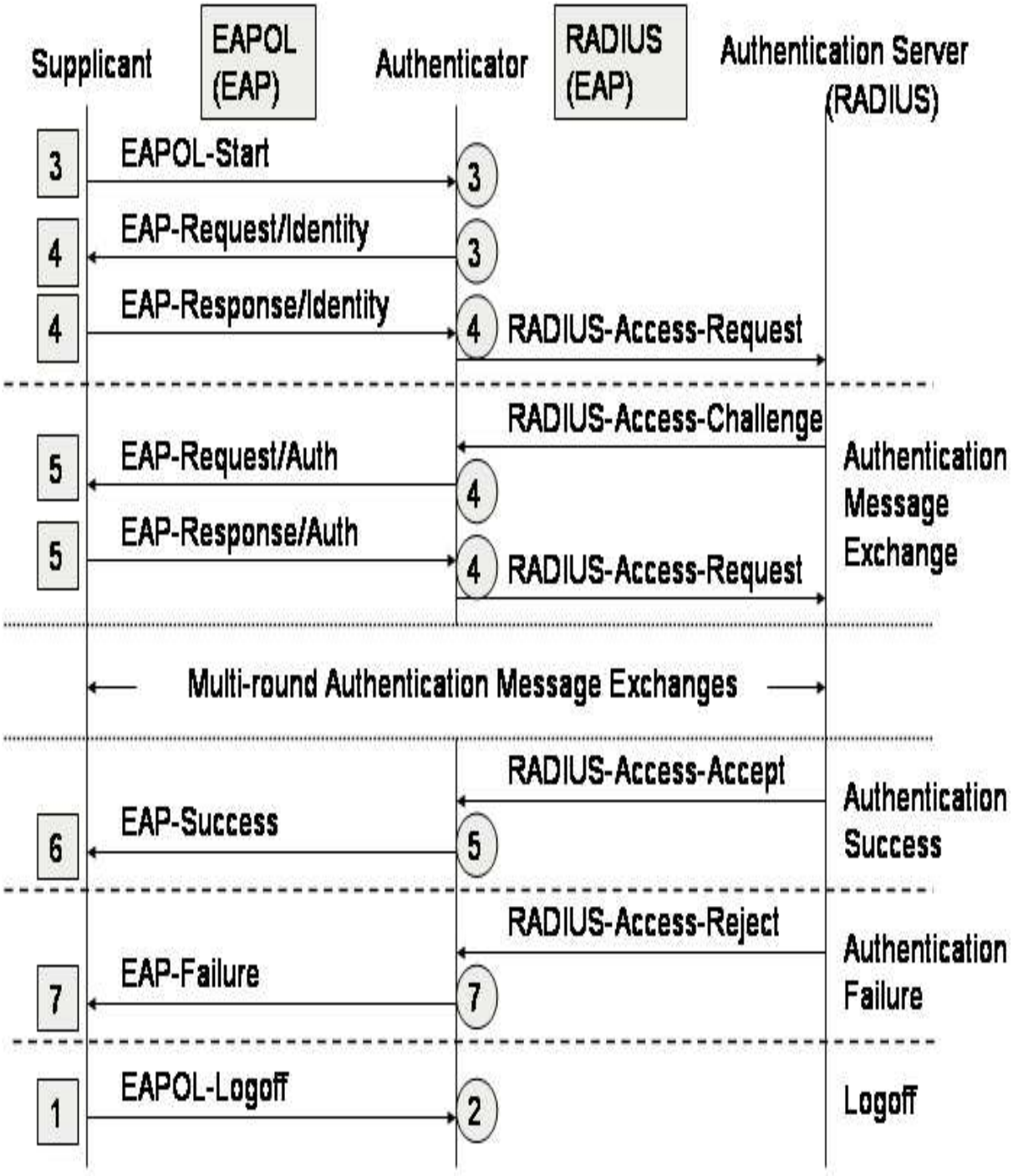


Figure 2: Example flows of IEEE 802.1X message exchange

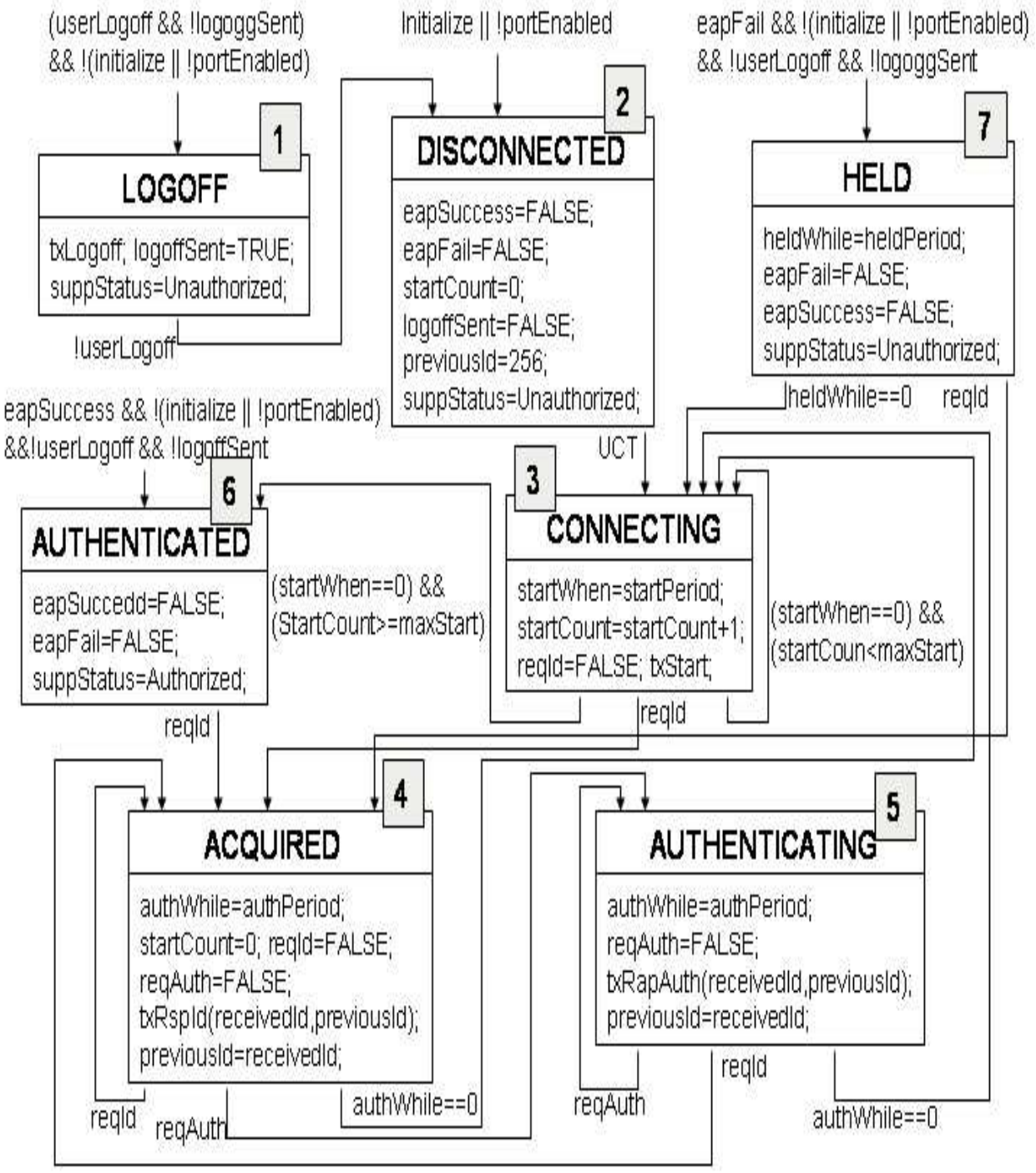


Figure 3: PAE (Port Access Entity) state machine in supplicant

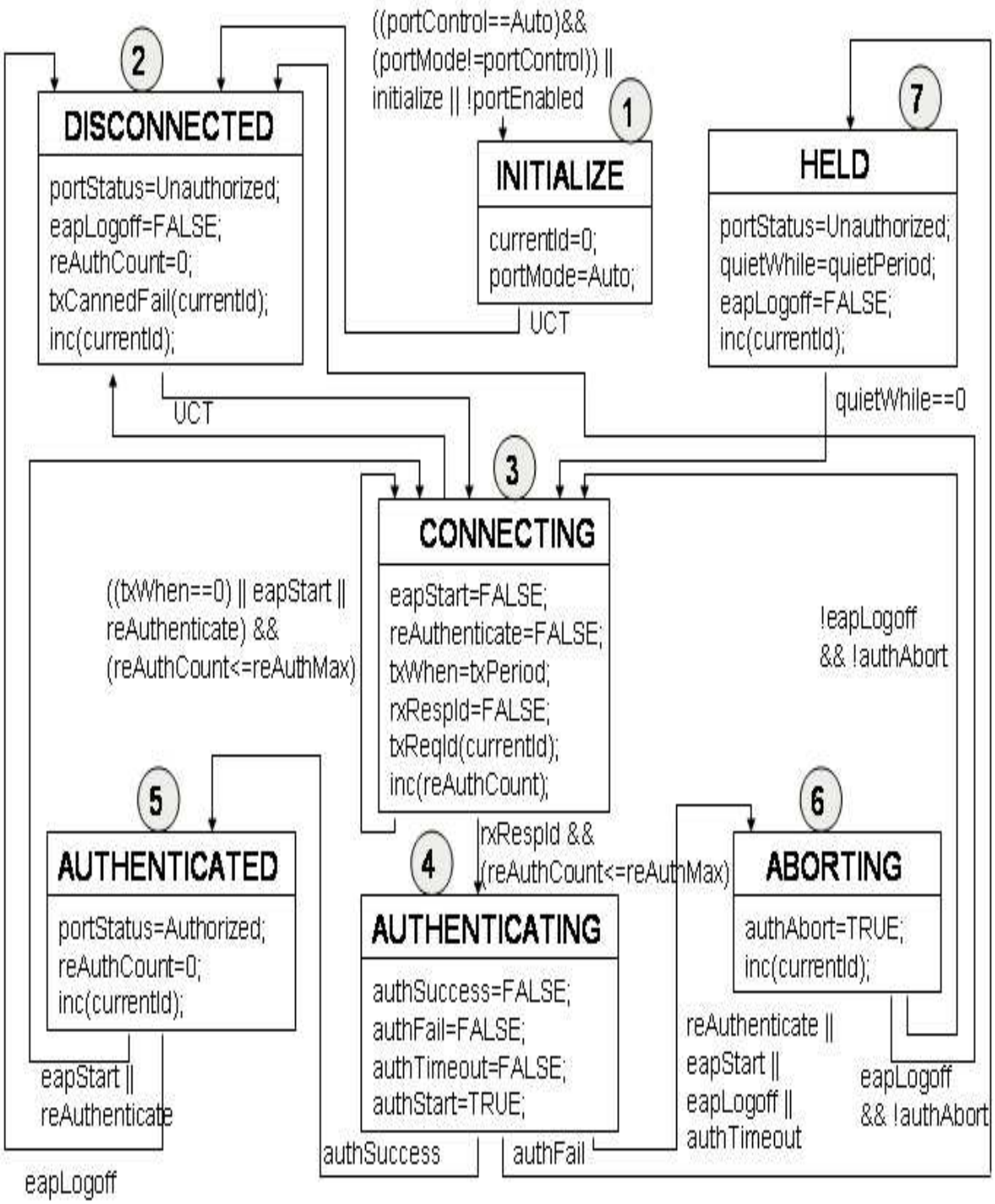


Figure 4: PAE (Port Access Entity) state machine in authenticator

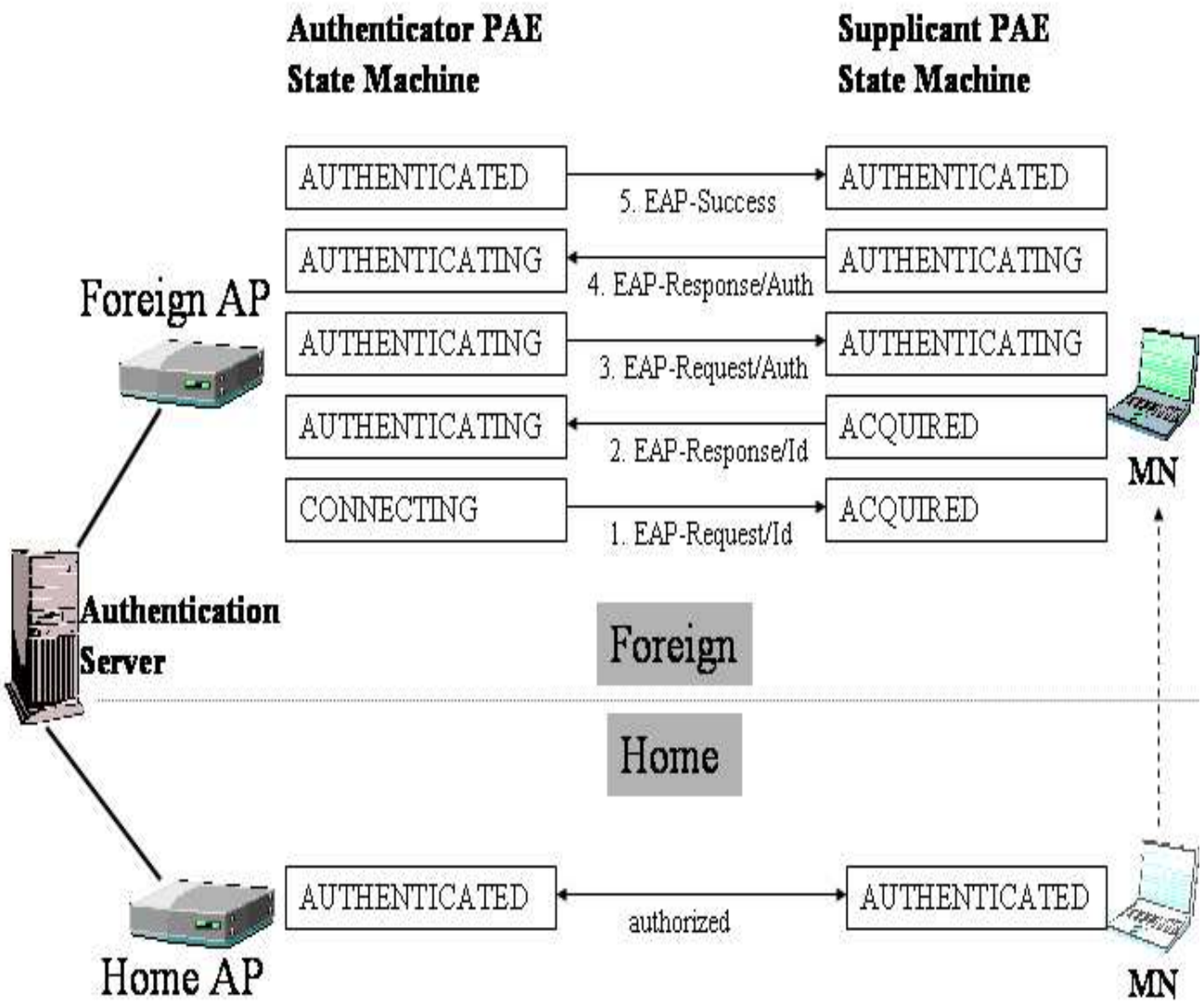


Figure 5: State transition in roaming

| Element ID | Length | Version | Group Key Cipher Suite | Pairwise Key Cipher Suite Count | Pairwise Key Cipher Suite List | Authentication and Key Management Suite Count | Authentication and Key Management Suite List | RSN Capabilities |
|---|---|---|---|---|---|---|---|---|

Figure 6: RSN IE format

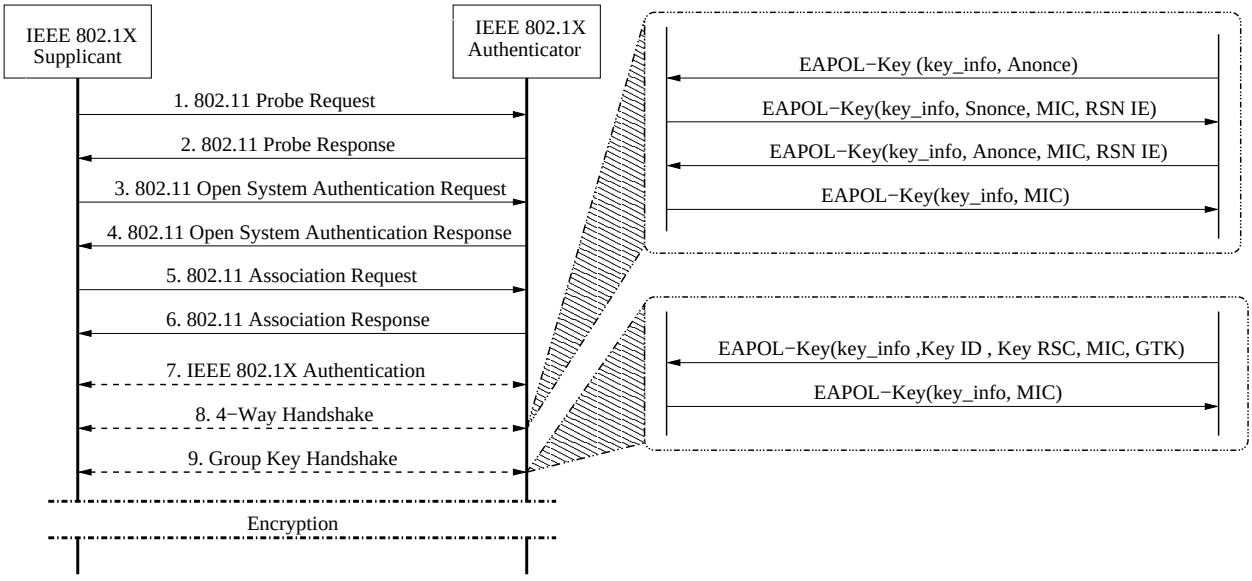


Figure 7: Example flows of RSNA establishment