Chair for Network Architectures and Services
Department of Informatics
TU München – Prof. Carle

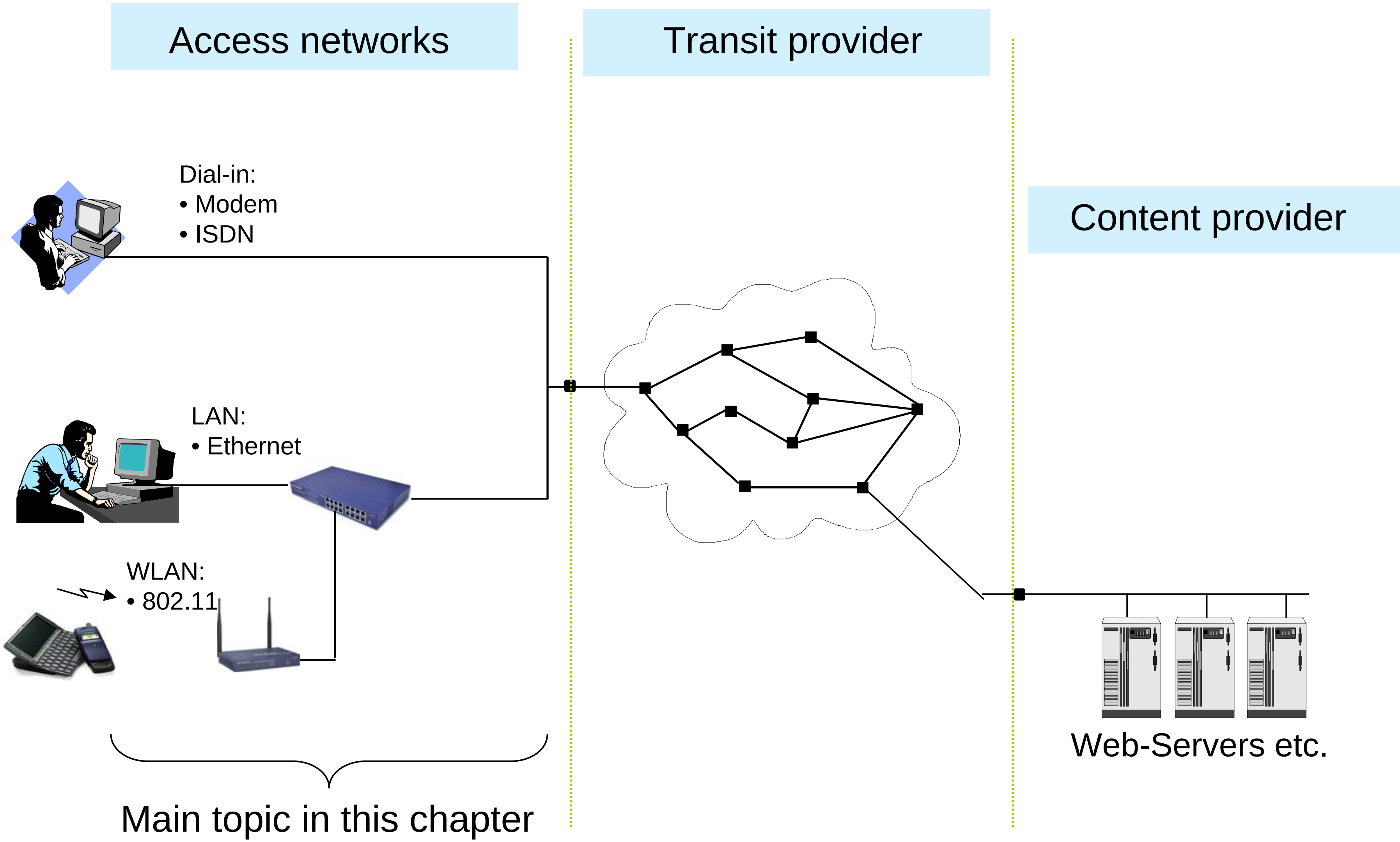
Network Security

Chapter 6
Security Protocols
of the Data Link Layer

Overview

- Introduction
- Point-to-Point Protocol (PPP)
- Extensible Authentication Protocol (EAP)
- IEEE 802.1x
- AAA Protocols
- Wireless LAN Security
  - WEP Security Flaws, WPA, WPA2
- Conclusions

Localization of Access Networks within the Internet-Based IT-Infrastructure

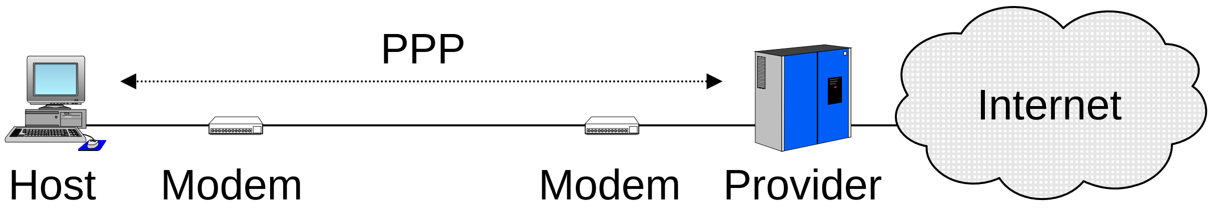


Overview

- Introduction
- **Point-to-Point Protocol (PPP)**
- Extensible Authentication Protocol (EAP)
- IEEE 802.1x
- AAA Protocols
- Wireless LAN Security
- Conclusions

Point-to-Point Protocol: Purpose and Tasks

- Large parts of the Internet rely on point-to-point connections:
  - Wide area network (WAN) connections between routers
  - Dial-up connections of hosts using (DSL) modems and telephone lines
- Protocols for this purpose:
  - Serial Line IP (SLIP): no error detection, supports only IP, no dynamic address assignment, no authentication [RFC 1055]
  - Point-to-Point Protocol (PPP): successor to SLIP, supports IP, IPX, ...



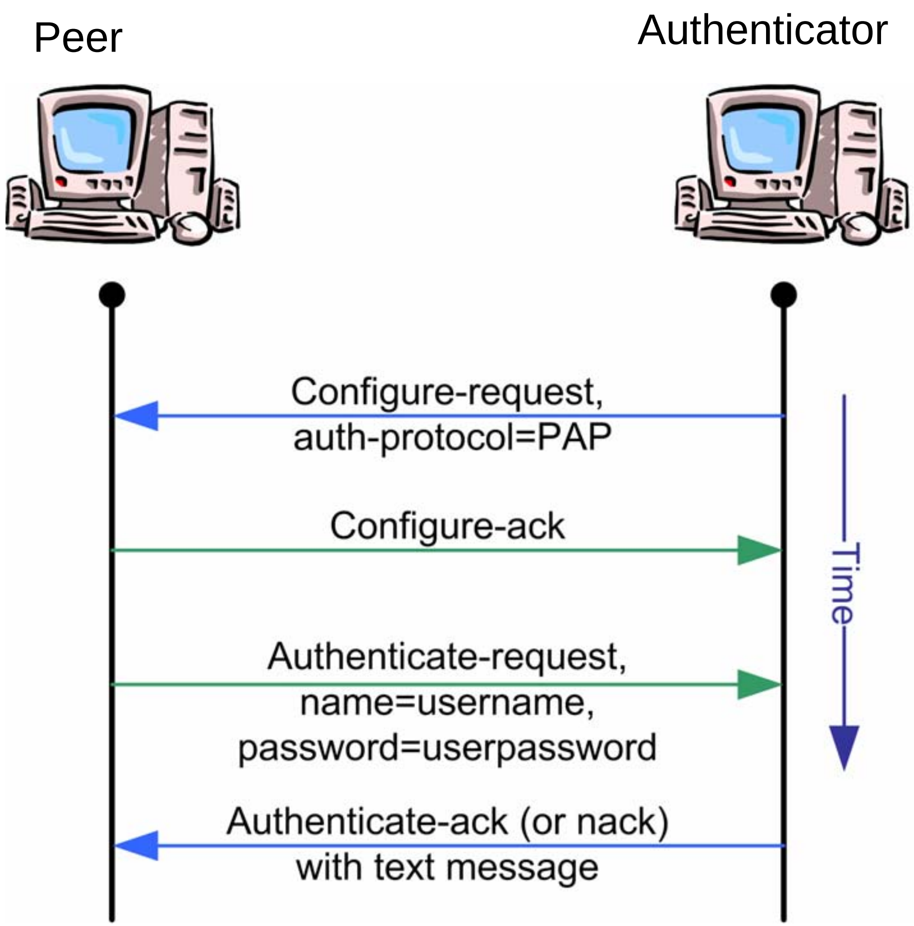
- PPP [RFC 1661/1662]:
  - Layer-2 frame format with frame delimitation and error detection
  - Control protocol *(Link Control Protocol, LCP)* for connection establishment, test, negotiation, and release

Point-to-Point Protocol: Security Services

- Entity authentication
  - The original version of PPP [RFC 1661] suggests the optional use of an authentication protocol after the link establishment phase:
    - If required, authentication is demanded by one peer entity via a LCP (Link Control Protocol) message at the end of the link establishment phase
    - Originally, two authentication protocols have been defined:
      - *Password Authentication Protocol (PAP)*
      - *Challenge Handshake Authentication Protocol (CHAP)*
    - Meanwhile, an extensible protocol has been defined:
      - *Extensible Authentication Protocol (EAP)*
- Encryption
  - PPP allows to negotiate data encryption after entity authentication with the *Encryption Control Protocol (ECP)*
  - However, ECP does not provide a mechanism for key management
  - Currently nobody uses ECP because there is no non-manual means of keying it.
- Message authentication
  - PPP does not provide message authentication

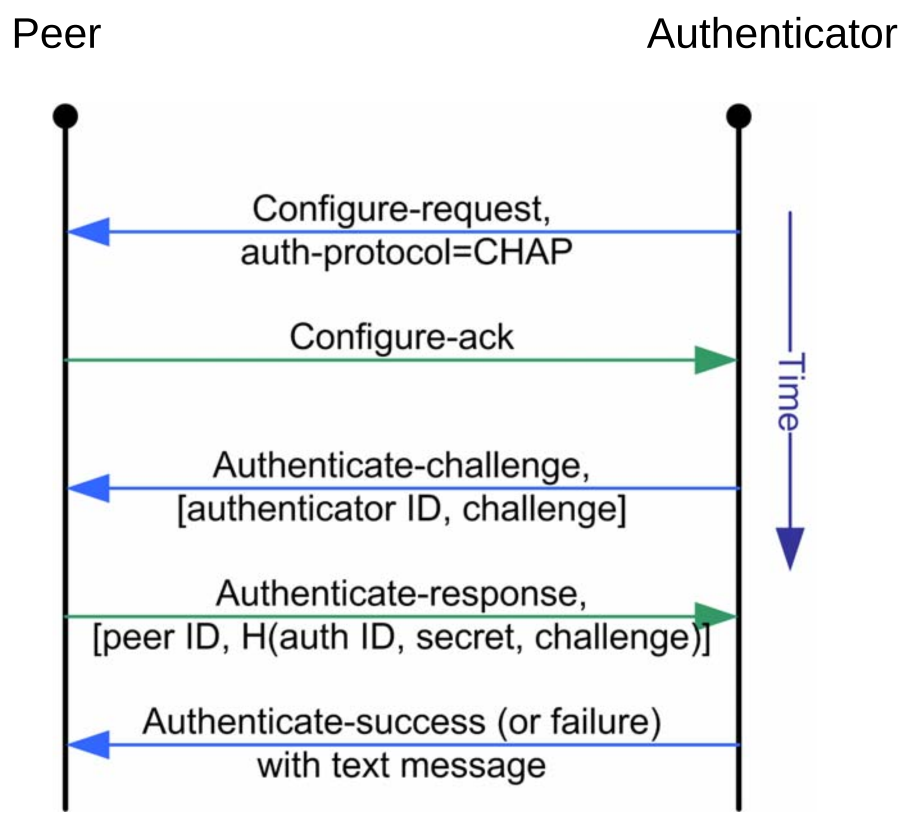
Point-to-Point Protocol: Authentication Protocols (1)

- Password Authentication Protocol (PAP):
  - PAP was defined 1992 [RFC 1334]



Point-to-Point Protocol: Authentication Protocols (2)

- Challenge Handshake Authentication Protocol (CHAP):



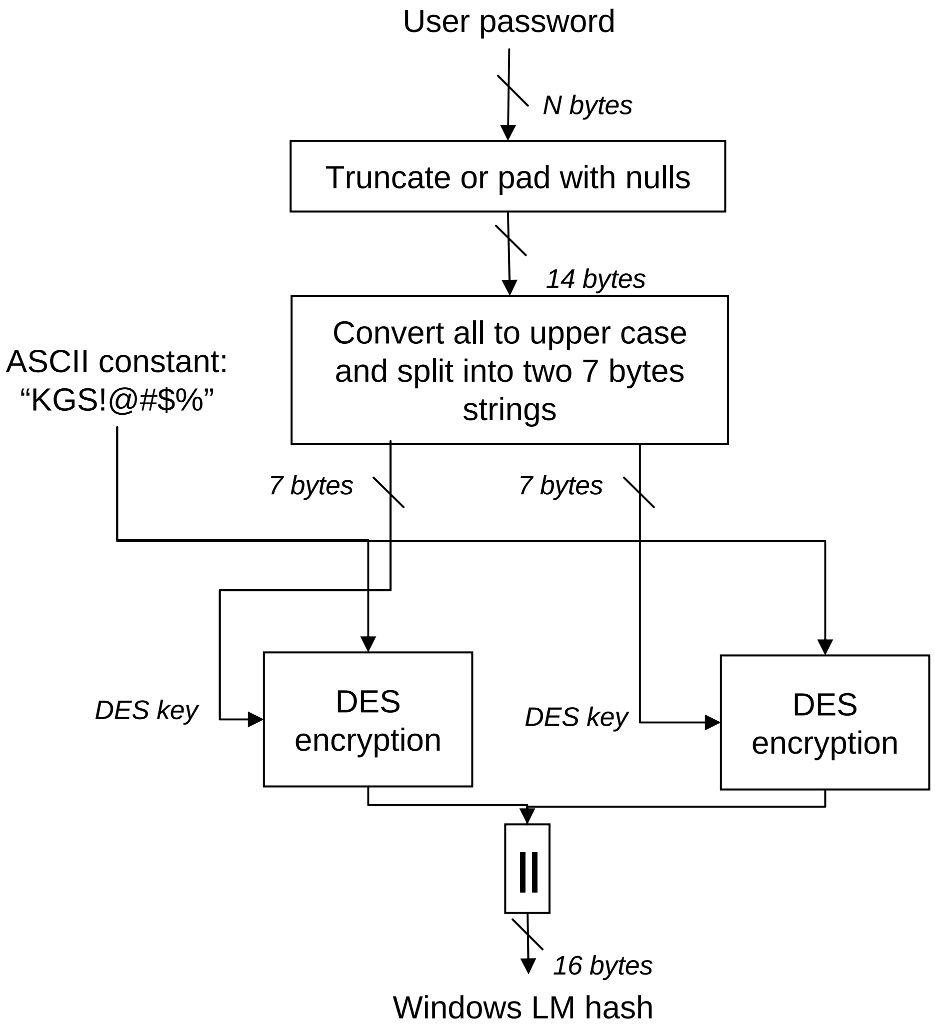
PPP Security – Reality Check (1)

- The lack of key management for PPP has lead to proprietary protocols with some security holes
  - Microsoft implemented CHAP with a home-made hash function
  - The Microsoft PPP authentication protocol was standardized as MSCHAP [RFC2433]
  - MSCHAP was accompanied with a proprietary key derivation mechanism.
    - The session key can be derived from the user's password.
    - The so-called Microsoft Point-to-Point Encryption (MPPE) was published in [RFC3078]
  - A security analysis of MSCHAP and MPPE was published by Schneier, *et al,* in 1998 [SMW99a] and show ed that MSCHAP and MPPE can be easily compromised
  - As a response to [SMW99a] Microsoft updated MSCHAP (➔ MSCHAP2) and MPPE

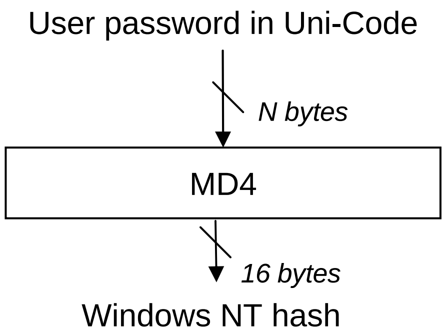
MSCHAP (1)

- MSCHAP uses
  - the Windows LAN Manager hash function
  - and the Windows NT hash function
- Windows LAN Manager Hash function:
  1. Turn the password into a 14-character string, either by truncating longer passwords or padding shorter passwords with nulls.
  2. Convert all lowercase characters to uppercase. Numbers and non-alphanumerics remain unaffected.
  3. Split the 14-byte string into two seven-byte halves.
  4. Using each seven-byte string as a DES key, encrypt a fixed constant with each key, yielding two 8-byte encrypted strings.
  5. Concatenate the two strings together to create a single 16-byte hash value.
- Windows NT Hash function:
  1. Convert the password case sensitive up to 14 bytes into Uni-Code
  2. The password is hashed using MD4, yielding a 16 byte hash value

MSCHAP (2)



Windows LAN Manager Hash Function



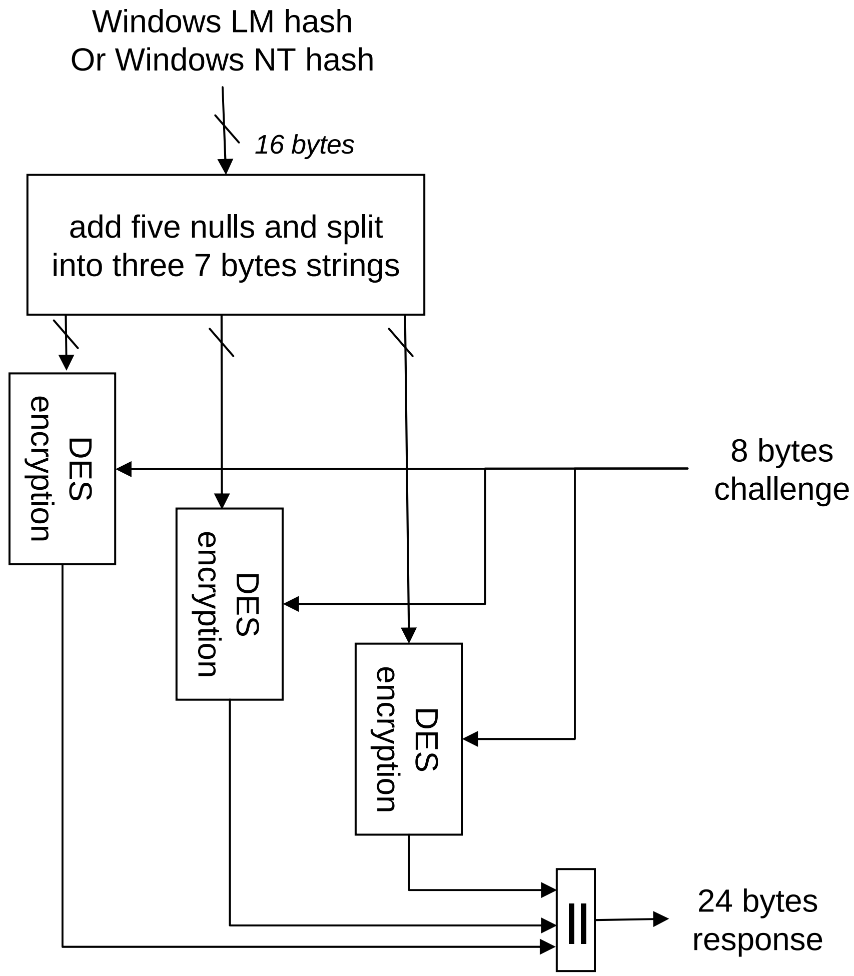
Windows NT Hash Function

MSCHAP (3)

- Weaknesses of the Windows LAN Manager hash function
  - Users typically choose poor passwords with small entropy
  - All characters are converted to upper case, making the number of possible passwords even smaller
  - The two seven-byte “halves" of the password are hashed independently
    - Thus, the two halves can be brute-forced independently, and the complexity of the attack is at most the complexity against a seven-byte password. Passwords longer than seven characters are no stronger than seven-character passwords.
  - Passwords of seven characters or less can be immediately recognized since the second half of the hash is always the same constant

MSCHAP (4)

- MSCHAP authentication dialogue
  1. Client requests a login challenge.
  2. Server sends back an 8-byte random challenge
  3. The client calculates the LAN Manager hash, and adds 5 nulls to create a 21-byte string, and partitions the string into three 7-byte keys. Each key is used to encrypt the challenge, resulting in a 24-byte encrypted value which is returned to the server
  - The client does the same with the Windows NT hash.
  - Given a challenge and the corresponding response that is computed with the Windows LM hash function, a dictionary attack can be performed within few minutes



MSCHAP Function Block

PPP Security – Reality Check (2)

- A security analysis of MSCHAP2 and the update of MPPE was published by Schneier in [SMW99a]
  - *„the fundamental weakness of the authentication and encryption protocol is that it is only as secure as the password chosen by the user"*
- MSCHAP2 and MPPE are still widely used
- However, in order to cope with the security weaknesses of legacy authentication methods, such as MSCHAP2, the authentication can be performed in 2 phases:
  - a TLS tunnel is established to the Authenticator first

    (Note: the client needs to verify the certificate of the Authenticator here)
  - then legacy (weak) authentication method is performed, e.g. PAP, CHAP, MSCHAP2
- Nevertheless, misconfigured Internet provider networks can lead to the hijacking of DSL connections
- A funny and interesting attack in practice can be found in [heise07]

Overview

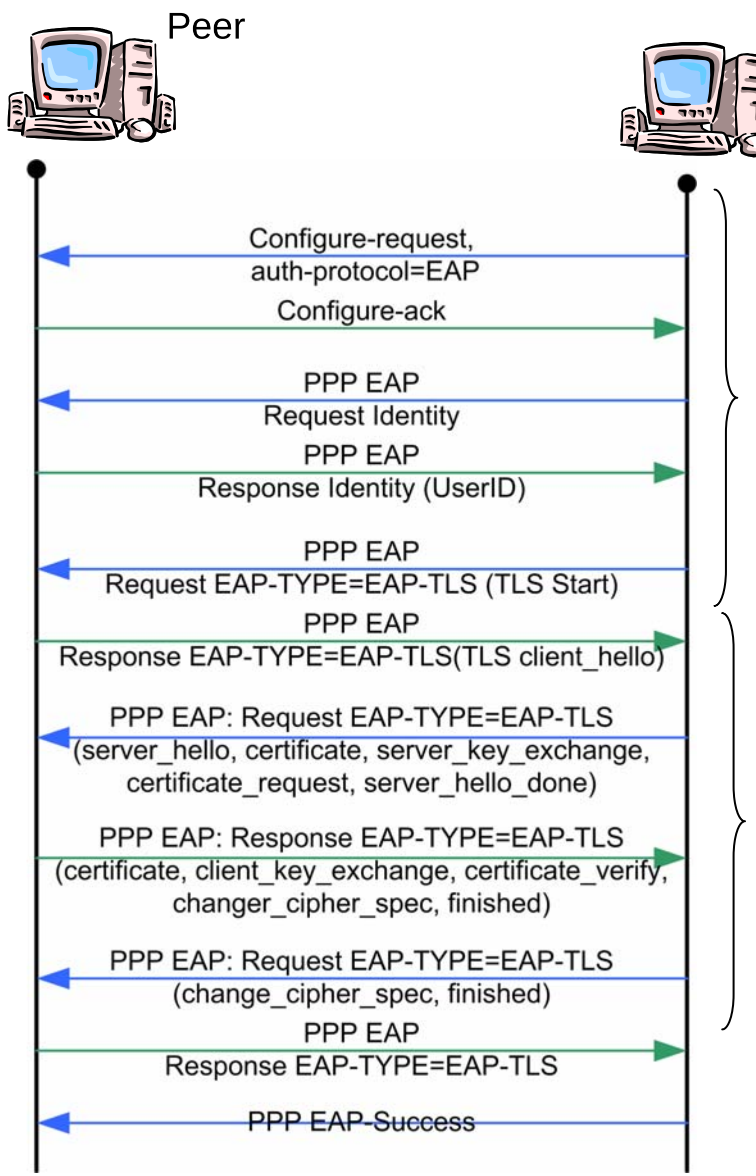
- Introduction
- Point-to-Point Protocol (PPP)
- **Extensible Authentication Protocol (EAP)**
- IEEE 802.1x
- AAA Protocols
- Wireless LAN Security
- Conclusions

Extensible Authentication Protocol (1)

- EAP is a general protocol for PPP authentication which supports multiple authentication methods [RFC2284]
- The main idea behind EAP is to provide a common protocol to run more elaborated authentication methods than "1 question + 1 answer"
- The protocol provides basic primitives:
  - Request, Response: further refined by *type field* + *type specific data*
  - Success, Failure: to indicate the result of an authentication exchange
- As EAP provides a generic framework for authentication, it supports several EAP methods, e.g.
  - EAP-MD5 Challenge (this is equivalent to CHAP)
  - EAP-TLS

Extensible Authentication Protocol (2)

- e.g. EAP-TLS:



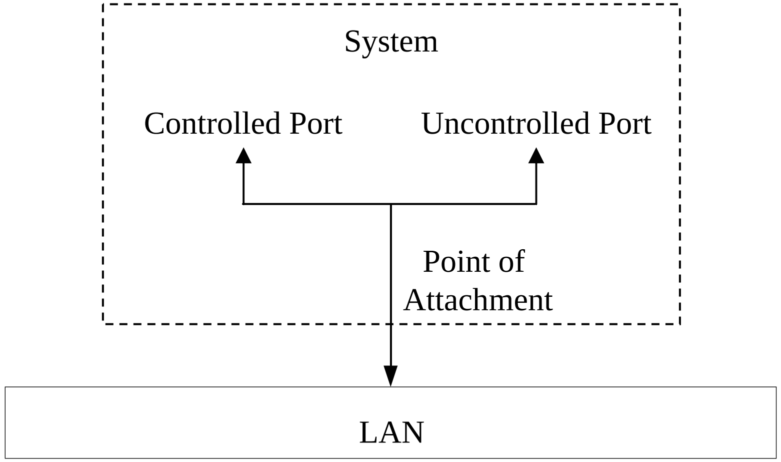
Negotiate EAP-TLS for authentication and exchange UserID

TLS handshake:
TLS messages are carried within an EAP message envelopes

Overview

- Introduction
- Point-to-Point Protocol (PPP)
- Extensible Authentication Protocol (EAP)
- **IEEE 802.1x**
- AAA Protocols
- Wireless LAN Security
- Conclusions

IEEE 802.1x: Controlled and Uncontrolled Ports



- IEEE 802.1x introduces the notion of two ***logical*** ports:
  - the uncontrolled port allows to authenticate a device
  - the controlled port allows an authenticated device to access LAN services
- Accessing a LAN with IEEE 802.1x security measures:
  - Prior to successful authentication the client can access the uncontrolled port:
    - The port is uncontrolled in the sense that it allows access prior to authentication
    - However, this port allows only restricted access
  - Authentication can be initiated by the client or the authenticator (e.g. LAN switch or WLAN access point)
  - After successful authentication the controlled port is opened

IEEE 802.1x: Roles

- Three principal roles are distinguished:
  - A device that wants to use the service offered by an IEEE 802.1x LAN acts as a ***supplicant*** requesting access to the controlled port
  - The point of attachment to the LAN infrastructure (e.g. a MAC bridge) acts as the ***authenticator*** demanding the supplicant to authenticate itself
  - The authenticator does not check the credentials presented by the supplicant itself, but passes them to his ***authentication server*** for verification
- *Authenticator* and *authentication server* communicate together using a so-called AAA protocol.

IEEE 802.1x Security Protocols & Message Exchange

- IEEE 802.1x does not define its own security protocols, but advocates the use of existing protocols:
  - The *Extensible Authentication Protocol (EAP)* may realize basic device authentication [RFC 2284]
  - If negotiation of a session key during authentication is required, the use of the *PPP EAP TLS Authentication Protocol* is recommended [RFC 2716]
  - Note however that newer methods might be appropriate, e.g. EAP-TTLS or PEAP
  - Furthermore, the *authentication server* is recommended to be realized with a AAA protocol such as RADIUS [RFC 2865] or DIAMETER [RFC 3588]

    (Diameter is the successor of the Radius protocol)
- Exchange of EAP messages between supplicant and authenticator is realized with the *EAP over LANs (EAPoL)* protocol:
  - EAPoL defines the encapsulation techniques that shall be used in order to carry EAP packets between the *supplicant* and the *Authenticator* in a LAN environment.

Overview

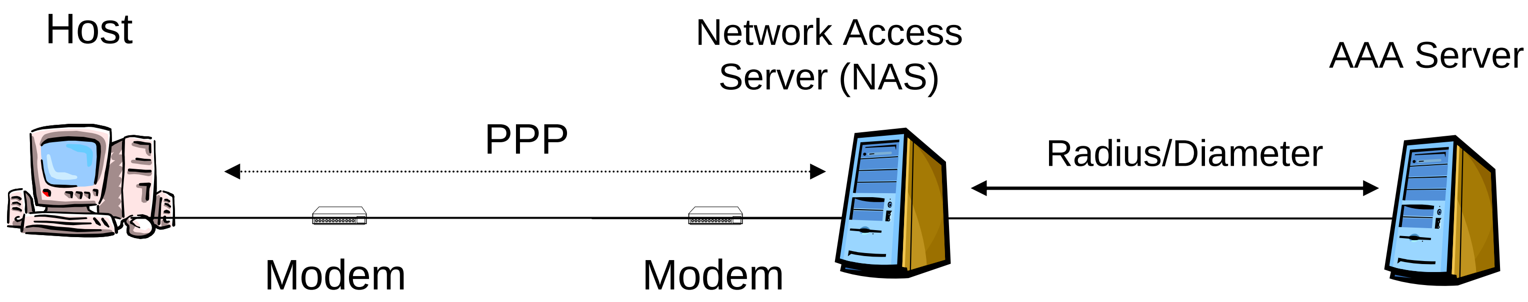
- Introduction
- Point-to-Point Protocol (PPP)
- Extensible Authentication Protocol (EAP)
- IEEE 802.1x
- **AAA Protocols**
- Wireless LAN Security
- Conclusions

Authentication, Authorization and Accounting (AAA) Protocols

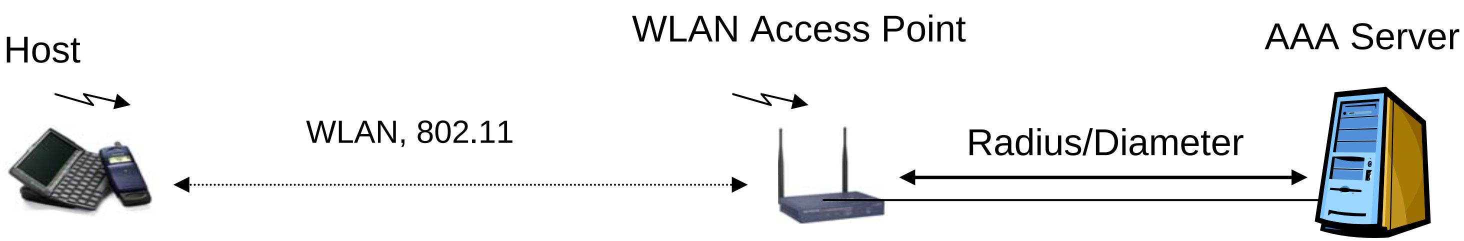
- <u>Motivation</u>
  - Provide a generic architecture for Authentication, Authorization and Accounting
  - Delegate AAA tasks (e.g. verification of user credentials such as passwords) to dedicated AAA servers.
  - AAA data (e.g. login/passwords) do not need to be stored at each *authenticator* device, e.g. Ethernet switch or wireless LAN access point.
  - The user database (e.g. login/passwords) can be re-used for several purposes and does not need to be duplicated (duplication can lead to inconsistency)

AAA Application Scenarios

- Authentication for dial-in services



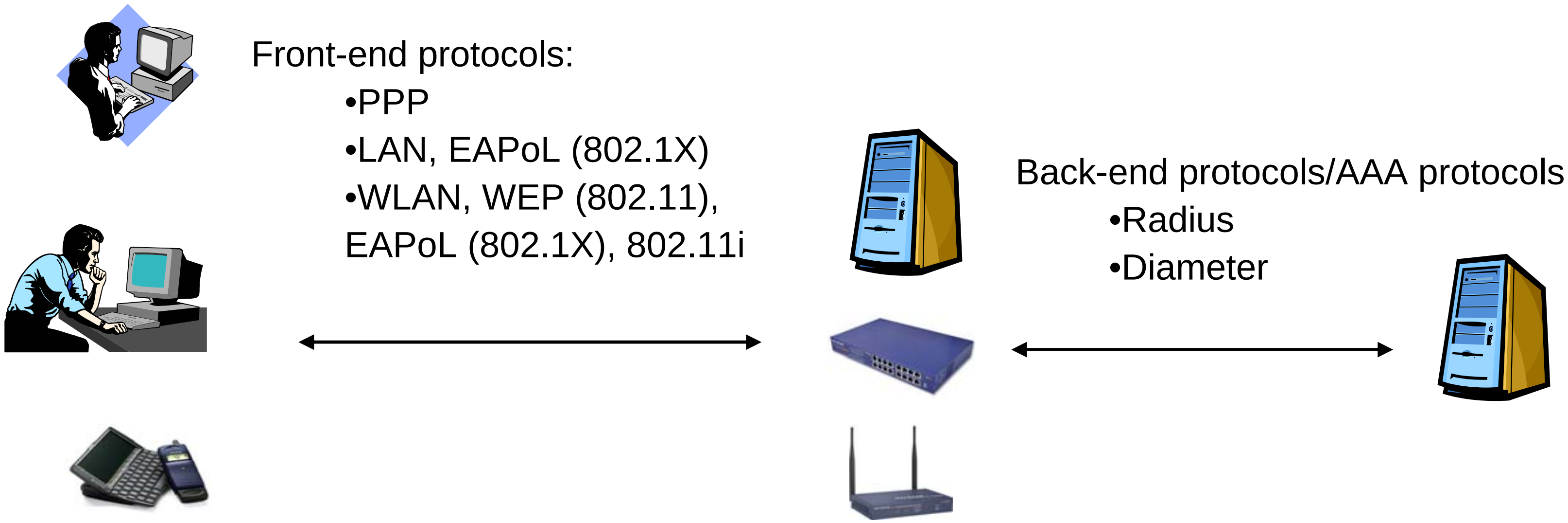
- Authentication for access to a wireless LAN network:



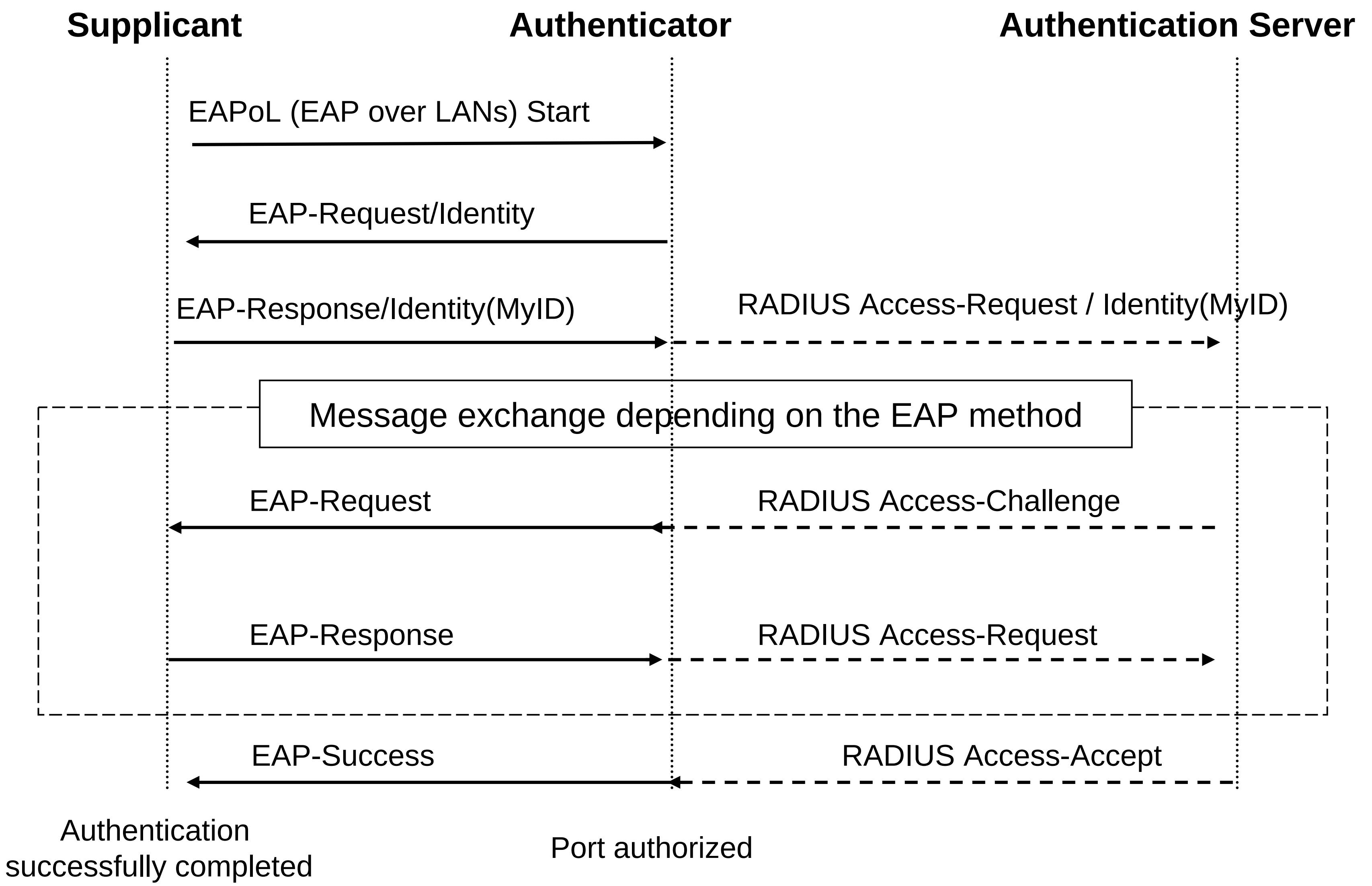
- AAA protocols can be also used between an Ethernet switch and a AAA server for access control with 802.1X
- Another application for AAA protocols (at the application layer) is the authenticating of users in Voice over IP (VoIP) networks

Back-End and Front-End Protocols

- Protocols between Supplicant and Authenticator are also called *Front-end protocols*
- Protocols between Authenticator and AS are also called *Back-end protocols*

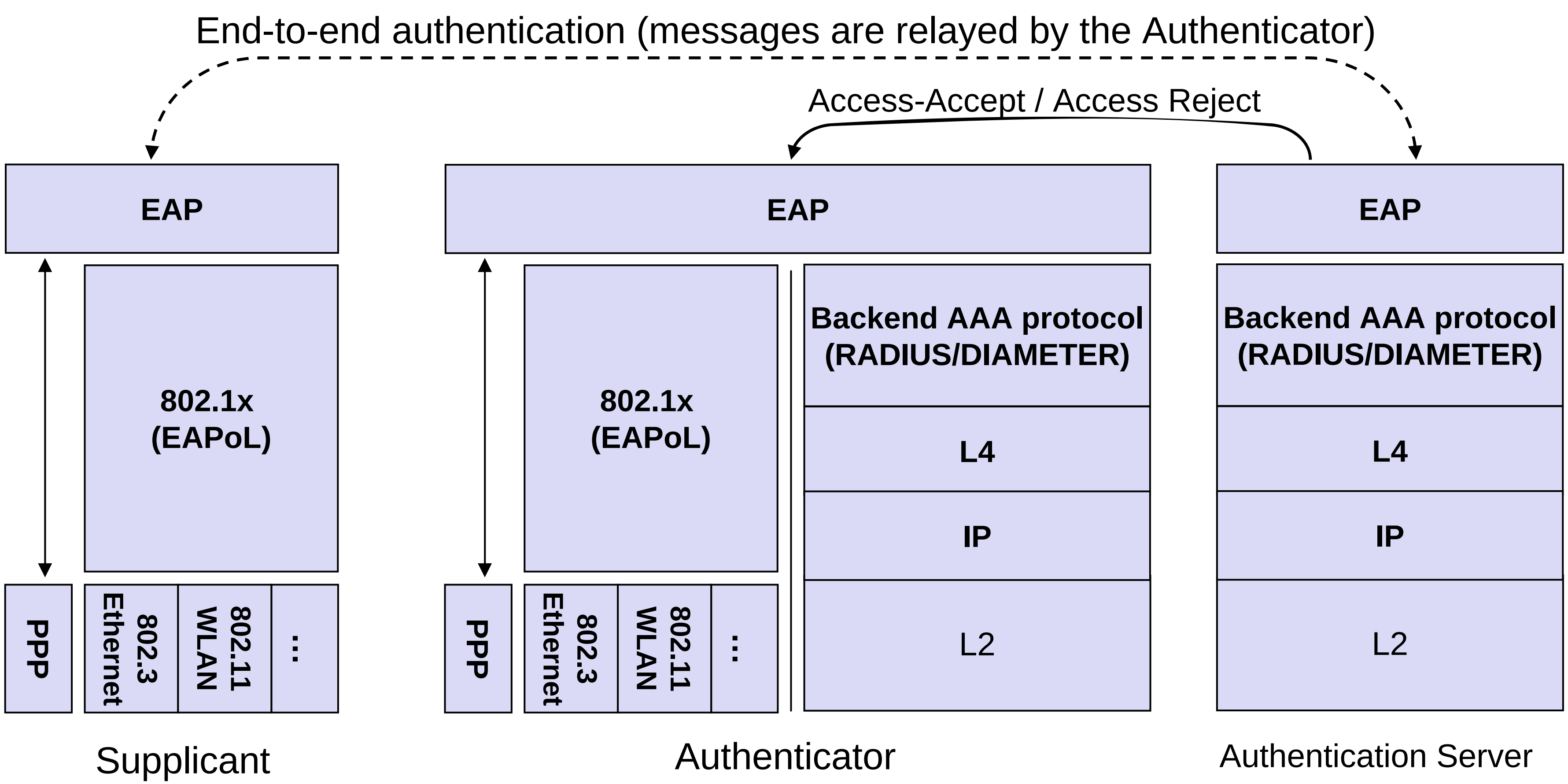


Putting the pieces together: Network Access Control with 802.1X, EAP and a AAA backend server



Putting the pieces together: EAP, 802.1X and AAA Protocols

- EAP was originally designed for PPP
- EAPoL encapsulates EAP messages within Ethernet or WLAN frames
- Between the authenticator and the authentication server, EAP messages are encapsulated within RADIUS/DIAMETER messages



Overview

- Introduction
- Point-to-Point Protocol (PPP)
- Extensible Authentication Protocol (EAP)
- IEEE 802.1x
- AAA Protocols
- **Wireless LAN Security**
- Conclusions

Wireless Security - Overview

- IEEE 802.11
- Wired Equivalent Privacy (WEP)
  - Security Flaws
- Access Control with 802.1X
- Wi-Fi Protected Access (WPA)
  - Temporal Key Integrity Protocol
- WPA2

IEEE 802.11

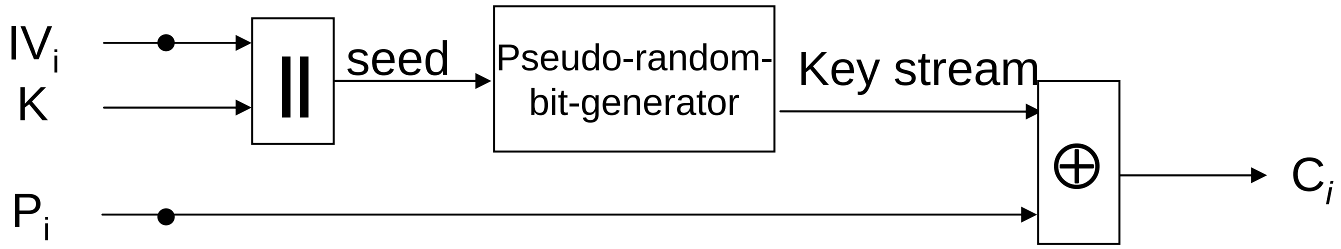
- IEEE 802.11 standardizes medium access control (MAC) and physical characteristics of a wireless *local area network (LAN)*
- Transmission occurs in the license-free 2.4 GHz band
- The medium access control (MAC) supports operation under control of an access point as well as between independent stations
- In this class we will mainly focus on the standard's security aspects:
  - Some equipment vendors claimed that IEEE 802.11 is as secure as a wired network (more on this below...)

Security Services of IEEE 802.11

- Security services of IEEE 802.11 are realized by:
  - Entity authentication service
  - *Wired Equivalent Privacy (WEP)* mechanism
- WEP is supposed to provide the following security services:
  - Confidentiality
  - Data origin authentication / data integrity
- WEP makes use of the following algorithms:
  - The RC4 stream cipher (please refer to chapter 3)
  - The Cyclic Redundancy Code (CRC) checksum for detecting errors

The Stream Cipher Algorithm RC4

- RC4 is a *stream cipher* that has been invented by Ron Rivest in 1987
- It was proprietary until 1994 when someone posted it anonymously to a mailing list
- RC4 works in Output Feedback (OFB) mode
  - The RC4 algorithm generates a pseudo-random sequence RC4(*IV*, *K*), that depends only on an initialization vector *IV* concatenated with the key *K*
  - The plaintext P_i is then XORed with the pseudo-random sequence to obtain the ciphertext and vice versa:
    - $C_i = P_i \oplus \text{RC4}(IV_i, K)$
    - $P_i = C_i \oplus \text{RC4}(IV_i, K)$



RC4 Encryption Block Diagram

Security of RC4 (1)

- RC4 uses a variable length key up to 2048 bit
  - The key serves as the seed for a pseudo-random-bit-generator
  - The variable key length of up to 2048 bit allows to make brute force attacks impractical (at least with the resources available in our universe)
  - However, by reducing the key length RC4 can also be made arbitrarily insecure!

- Known-Plain-Text Attacks on RC4:
  - It is crucial to the security of the RC4 that the initialization vector is never re-used!
    - If the plain text P_1 of a given ciphertext C_1 can be guessed and it happens that the initialization vector IV_1 is re-used later (i.e. $IV_1 = IV_2$ with the same K), then we have the same keystream $RC4(IV_1, K) = RC4(IV_2, K)$, then C_2 can be easily decrypted :

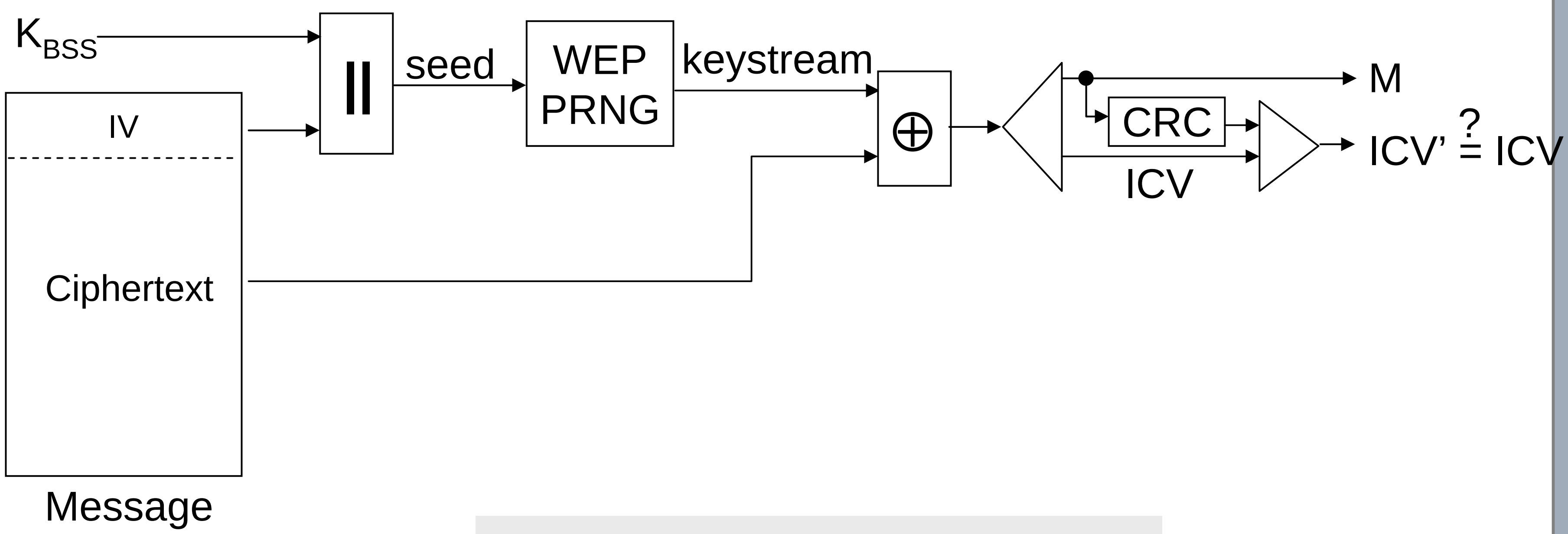
    $$P_2 = C_2 \oplus RC4(IV_2, K) = C_2 \oplus RC4(IV_1, K) = C_2 \oplus (C_1 \oplus P_1)$$
  - This means if all possible IVs has been used, key re-negotiation is necessary before proceeding.
  - However, if no key management is provided (*K* is constant) and the *IV* is short, a repetition of the same IV, and therefore a repetition of the keystream, can occur quickly.

Security of RC4 (3)

- In 2001 a new and surprising discovery was made by Fluhrer, Mantin and Shamir [FMS01a] :
  - Over all possible RC4 keys, the statistics for the first few bytes of output keystream are strongly non-random, leaking information about the key.
  - If the long-term key and nonce are simply concatenated to generate the RC4 key, this long-term key can be discovered by analyzing a large number of messages encrypted with this key.
  - This and related effects were then used to break the WEP ("wired equivalent privacy") encryption
- Applications using RC4 could defend against this attack by discarding the initial portion of the keystream (say the first 1024 bytes) before using it.

IEEE 802.11's Wired Equivalence Privacy (2)

- As *IV* is send in clear with every message, every receiver who knows K_{BSS} can produce the appropriate keystream to decrypt a message
  - This assures the important *self-synchronization property* of WEP
- The decryption process is basically the inverse of encryption:



WEP Decryption Block Diagram

Weakness #1: The Keys

- IEEE 802.11 does not specify any key management:
  - Manual management is error prone and insecure
  - Shared use of one key for all stations of a BSS introduces additional security problems
  - As a consequence of manual key management, keys are rarely changed

- Key Length:
  - The key length of 40 bit specified in the original standard provides only poor security
  - The reason for this was exportability
  - Note that
    - today’s wireless LAN cards often also allow keys of length 128 bit
    - However, WEP is still insecure even with 128 bits key length due to the reasons explained in the next slides.

Weakness #2: WEP Confidentiality is Insecure

- Even with well distributed and long keys WEP is insecure
- The reason for this is the reuse of keystream:
  - Recall that encryption is re-synchronized with every message by pre-pending an *IV* of length 24 bit to K_{BSS} and re-initializing the PRNG
  - Consider two plaintexts M_1 and M_2 encrypted using the same IV_1:
    - $C_1 = P_1 \oplus \text{RC4}(IV_1, K_{BSS})$
    - $C_2 = P_2 \oplus \text{RC4}(IV_1, K_{BSS})$
  - If an attacker knows, for example, P_1 and C_1 he can recover P_2 from C_2 without knowledge of the key K_{BSS}
    - $P_2 = C_1 \oplus C_2 \oplus P_1$
- How often does reuse of IV occur?
  - In practice quite often, as many implementations choose *IV* poorly
  - Even with optimum random choice, as IV's length is 24 bit, according the Birthday-Paradox it is expected that IV will be repeated after ~ 2^{12} WLAN frames

The Cyclic Redundancy Code (1)

- The cyclic redundancy code (CRC) is an error detection code
- Mathematical basis:
  - Treat bit strings as representations of polynomials with coefficients 0 and 1 $\Rightarrow$ a bit string representing message *M* is interpreted as *M(x)*
  - Polynomial arithmetic is performed modulo 2

    $\Rightarrow$ addition and subtraction are identical to XOR
- CRC computation for a message M(x):
  - A and B agree upon a polynomial *G(x);* usually *G(x)* is standardized
  - Let *n* be the degree of *G(x)*, i.e. the length of *G(x)* is *n* + 1
  - Then if $\frac{M(x)\times 2^n}{G(x)} = Q(x) + \frac{R(x)}{G(x)}$ it holds $\frac{M(x)\times 2^n + R(x)}{G(x)} = Q(x)$

    where *R(x)* is the remainder of *M(x)* divided by *G(x)*
  - Usually, R(x) is appended to M(x) before transmission and Q(x) is not of interest, as it is only checked if $\frac{M(x)\times 2^n + R(x)}{G(x)}$ divides with remainder 0

The Cyclic Redundancy Code (2)

- Consider now two Messages M_1 and M_2 with CRCs R_1 and R_2:
  - As $\frac{M_1(x)\times 2^n + R_1(x)}{G(x)}$ and $\frac{M_2(x)\times 2^n + R_2(x)}{G(x)}$ divide with remainder 0

    also $\frac{M_1(x)\times 2^n + R_1(x) + M_2(x)\times 2^n + R_2(x)}{G(x)} = \frac{(M_1(x)+M_2(x))\times 2^n + (R_1(x)+R_2(x))}{G(x)}$

    divides with remainder 0

  $\Rightarrow$ CRC is additive, that is $CRC(M_1 \oplus M_2) = CRC(M_1) \oplus CRC(M_2)$

- i.e. if a message *M* is modified to a message *M‘*

  where $M' = CRC(M \oplus \Delta)$

  then $CRC(M') = CRC(M + \Delta) = CRC(M) + CRC(\Delta)$

- Due to this property CRC is not appropriate for cryptographic purposes! (more on this below...)

Weakness #3: WEP Data Integrity is Insecure

- Recall that CRC is an additive function and RC4 is additive as well
- Consider *A* sending an encrypted message to *B* which is intercepted by an attacker *E*:
  - $A \rightarrow B$: (IV, C) with $C = RC4(IV, K_{BSS}) \oplus (M, CRC(M))$
- The attacker *E* can construct a new ciphertext *C'* that will decrypt to a message *M'* with a valid checksum CRC(M'):
  - *E* chooses an arbitrary message Δ of the same length as M
  - $C' = C \oplus (\Delta, CRC(\Delta)) = RC4(IV, K_{BSS}) \oplus (M, CRC(M)) \oplus (\Delta, CRC(\Delta))$
    $= RC4(IV, K_{BSS}) \oplus (M \oplus \Delta, CRC(M) \oplus CRC(\Delta))$
    $= RC4(IV, K_{BSS}) \oplus (M \oplus \Delta, CRC(M \oplus \Delta))$
    $= RC4(IV, K_{BSS}) \oplus (M', CRC(M'))$
  - Note, that *E* does not know *M'* as it does not know *M*
  - Nevertheless, a "1" at position *n* in Δ results in a flipped bit at position *n* in *M'*, so *E* can make controlled changes to *M*

  $\Rightarrow$ Data origin authentication / data integrity of WEP is insecure!
- Recall that CRC is used for WEP as integrity function and it is computed without any key!

Weakness #5: Weakness in RC4 Key Scheduling

- In early August 2001 a new attack to WEP was discovered:
  - The shared key can be retrieved in less than 15 minutes provided that about 4 to 6 million packets have been recovered
  - The attack is basically a known-plaintext attack, that makes use of the following properties of RC4 and WEP's usage of RC4:
    - RC4 is vulnerable to deducing bits of a key if:
      - many messages are encrypted with keystream generated from a variable initialization vector and a fixed key, and
      - the initialization vectors and the plaintext of the first two octets are known for the encrypted messages
    - The IV for the keystream is transmitted in clear with every packet
    - The first two octets of an encrypted data packet can be guessed
  - The attack is described in [SMF01a] and [SIR01a]
  - R. Rivest comments on this [Riv01a]:

  "Those who are using the RC4-based WEP or WEP2 protocols to provide confidentiality of their 802.11 communications should consider these protocols to be broken [...]"

Summary of WEP weaknesses

- Missing key management makes use of the security mechanisms tedious and leads to rarely changed keys or even security switched off
- Entity authentication as well as encryption rely on a key shared by all stations of a basic service set
- 40 bit keys are too short to provide any security
- Re-use of keystream makes known-plaintext attacks possible
- Additive integrity function allows to forge ICVs
- Unkeyed integrity function allows to circumvent access control by creating valid messages from a known plaintext-ciphertext pair
- Weakness in RC4 key scheduling allows to crypto-analyze keys
- Even with IEEE 802.1x and individual keys the protocol remains weak

Evolution of WLAN Security (1)

- 802.11, which dates from 1997, helped to kick off the present adoption of WLANs, but was primarily concerned with connectivity and not with security.
- In June 2001 802.1X was ratified.
  - 802.1X provides Access Control, recommends the use of EAP with AAA servers for authentication.
  - However, 802.1X does not solve the confidentiality and integrity problems of WEP
- An IEEE Task Group had been working on a secure standard for WLANs: 802.11i. This was published in June 2004.
- In the mean time, (in October 2002), the Wi-Fi Alliance (a consortium of about 170 WLAN vendors) announced a security solution that counters the known weaknesses of WEP, called

  Wi-Fi Protected Access (WPA).

Evolution of WLAN Security (2)

- WPA was a snapshot of 802.11i.
- It was announced earlier than 802.11i due to the urgent need for a security solution for WLANs on the market and due to the slow process of standardization.
- However, WPA was only a short-term solution to patch WEP and re-uses the same hardware
- The long-term solution, also called ***WPA2***, uses
  - AES CTR mode for encryption instead of RC4
  - AES-CBC-MAC for data integrity

Wi-Fi Protected Access (WPA)

- WPA Authentication:
  - WPA incorporates the 802.1X standard with stations (Supplicant), access points (Authenticators) and authentication servers.
- Data Privacy (Encryption)
  - The Temporal Key Integrity Protocol (TKIP) for encryption is a rapid re-keying solution to patch WEP
  - TKIP provides a key management system with a *per-packet key* for WEP encryption to fix the WEP flaws
  - TKIP is a “work-around” to use the same WEP hardware while achieving a stronger encryption
- Data integrity:
  - TKIP includes also Message Integrity Code called MIC or „Michael“ at the end of each plaintext message to ensure messages are not being spoofed or altered.
  - Note: the IEEE uses the acronym MIC instead of MAC (Message Authentication Code) for the simple reason that MAC is reserved for „Medium Access Control“.
- TKIP is a work around WEP to correct its weaknesses while still using the same hardware

TKIP Rekeying (1)

- TKIP uses a key hierarchy to generate temporal keys that have a short lifetime and are frequently refreshed.
- The key hierarchy has three layers:

1. **Master key**:
   - The master key is the highest key in the hierarchy.
   - The master key is generated by the 802.1X authentication server during the authentication and is provided to the station (via the AP).
   - The master key is used to secure the distribution of the key-encryption keys.
   - A session structure can be formed based on this key, spanning from authentication until the key is revoked, expires, or the station looses contact with the infrastructure.
   - Note: if an attacker compromise the master key then he can trivially compromise the key-encryption keys and temporal keys, thus voiding any TKIP privacy claims.
2. **Key-encryption keys**:
   - The key-encryption keys are used to protect the transport of the temporal keys.
   - There are 2 key-encryption keys: one to encrypt the distributed keying material, and a second to protect the "*rekey* key" messages from forgery.

TKIP Rekeying (2)

3. Temporal keys:

- TKIP employs a pair of temporary key types:
  - a 128-bit encryption key
  - a 64-bit key for data integrity
- TKIP uses a separate key pair for each direction of an association.
- Hence, each association has a total of 4 temporal keys.
- **Temporal keys** are refreshed with a „*rekey key*“ message.
- The „*rekey key*“ message distributes keying material from which both the station and the Access Point derive the next set of temporal keys. This exchange is secured by the **key encryption keys**.

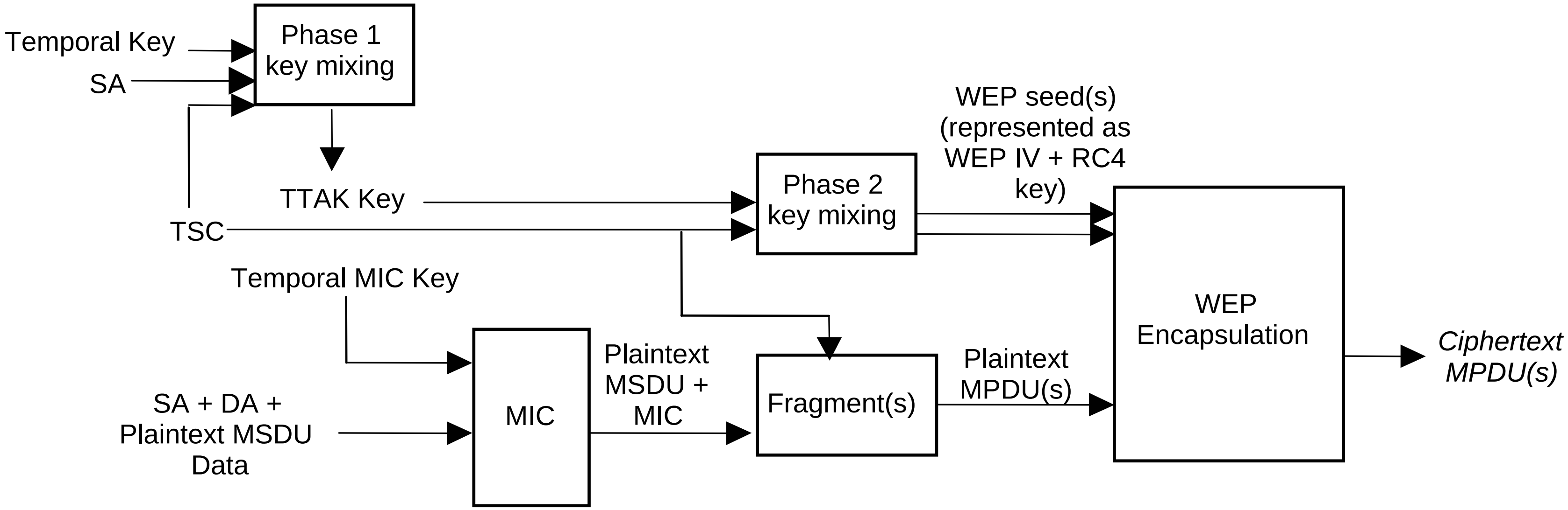
TKIP Key Mixing

- The temporal encryption keys are used to generate a *per-packet key* for WEP encryption.
- Note: this is not a new sophisticated method for encryption. It is designed just to correct the WEP‘s misuse of RC4.
- TKIP uses a function called the TKIP mixing function to transform the temporal key and a packet sequence number into a per-packet key and IV.
- The mixing function operates in 2 phases:
  - Phase 1 generates an intermediate key where:
    - intermediate key := *S* (MAC address, temporal key)
    - *S* is a non linear function which is a combination of table-look-ups and XOR.

    Note here that involving the MAC address avoids that 2 different stations could use the same key.
  - Phase 2 uses a cipher function to “encrypt” the packet sequence number under the intermediate key, producing a 128-bit per-packet WEP key (24 bits IV and 104 bits RC4 key).
  - The cipher function used here has a Feistel structure and is a combination of XOR, shift, rotate and table look-ups (all cheap CPU operations common on 802.11 devices).

TKIP Function Block on Sender Side

- Putting everything together:



- DA – Destination Address
- ICV– Integrity Check Value
- MPDU – Message Protocol Data Unit
- MSDU – MAC Service Data Unit
- SA – Source Address

- TSC – TKIP Sequence Counter
- TTAK– result of phase 1 key mixing of Temporal Key
- and Transmitter Address (Intermediate Key)

The improved Wireless LAN Security Standard: 802.11i

- The long term solution – also called ***WPA2***
  - Counter-Mode/CBC-MAC Protocol (CCMP):
    - Provides confidentiality, data integrity and replay protection
    - Uses AES in CTR mode for confidentiality
    - Uses AES-CBC-MAC (with a different key!) for data integrity

- Both WPA and WPA2 utilize
  - 802.1X for access control
  - EAP for authentication
- In both WPA and WPA2 the Authenticator can operate in
  - Stand-alone mode:
    - The Authenticator plays the role of the Authentication Server
  - Pass-through mode
    - The Authenticator relays authentication messages between the Supplicant and the Authentication Server.
    - When the authentication exchange is completed, the Authentication Server informs the Authenticator whether the Authentication was successful

Wireless LAN Security - Conclusions

- IEEE 802.11 does not provide sufficient security
- WPA uses TKIP for data encryption and integrity and 801.1X for access control
- 801.1X enables the use of different authentication methods by using EAP
- WPA2 uses CCMP which uses AES in CTR mode for encryption and AES-CBC-MAC for data integrity

Overview

- Introduction
- Point-to-Point Protocol (PPP)
- Extensible Authentication Protocol (EAP)
- IEEE 802.1x
- AAA Protocols
- Wireless LAN Security

- Conclusions

Link Layer Security – Summary and Conclusions (1)

- Mechanisms and protocols for *link layer security* aim at providing
  - Authentication of end hosts
  - Access control at the link layer
  - Data origin authentication at the link layer
  - Message integrity at the link layer
  - Confidentiality at the link layer
- Bad design and abuse of cryptography showed that these goals have been missed several times, e.g. MSCHAP, MSCHAP2, WEP
- Even though the introduction of EAP provided a basis for integrating stronger methods for authentication, initial EAP methods (e.g. EAP-MD5) do not provide keying material for a secure channel between the Supplicant and the Authenticator

Link Layer Security – Summary and Conclusions (2)

- IEEE/IETF standardization committees have learned lessons from other security protocols, e.g. IPSec and TLS
- However, requirements for link layer security are different
  - e.g. security have often to be implemented at the hardware interface with limited resources
  - Layer 2 frame properties and message overhead have to be considered
- Link layer security is still work-in-progress and it is expected to have many advancements and updates in the near future, e.g.
  - IEEE 802.1AE which is a standard for integrating security services, such as data integrity and confidentiality in Ethernet switches
  - Improvement of EAP methods, also with respect to latency in handover scenarios

Additional References (1)

[ATM99a] ATM Forum. *ATM Security Specification Version 1.0.* AF-SEC- 0100.000, February, 1999.

[BGW01a] N. Borisov, I. Goldberg, D. Wagner. *Intercepting Mobile Communications: The Insecurity of 802.11.* 7th ACM SIGMOBILE Annual International Conference on Mobile Computing and Networking (MOBICOM), Rome, Italy, July 2001.

[FH98a] P. Ferguson, G. Huston. What is a VPN? The Internet Protocol Journal, volume 1, no. 1&2, Cisco Systems. 1998.

[Gast07] M. Gast, „TTLS and PEAP Comparison“, 2007 http://www.opus1.com/www/whitepapers/ttlsandpeap.pdf

[heise07] Johannes Endres, “In Nachbars Netz - Fremde Daten per ADSL-Kurzschluss“, heise article, 2007,

http://www.heise.de/netze/In-Nachbars-Netz--/artikel/93149

[IEEE97a] IEEE. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* IEEE Std 802.11-1997, The Institute of Electrical and Electronics Engineers (IEEE), 1997.

[IEEE01a] IEEE. *Standards for Local and Metropolitan Area Networks: Standard for Port Based Network Access Control.* IEEE Draft P802.1X/D11, 2001.

[IEEEAE] IEEE. *802.1AE - Media Access Control (MAC) Security.* 2006.

[RFC1661] W. Simpson. *The Point-to-Point Protocol (PPP).* RFC 1661, 1994.

[RFC1968] G. Meyer. *The PPP Encryption Control Protocol (ECP).* RFC 1968, 1996.

[RFC1994] W. Simpson. *PPP Challenge Handshake Authentication Protocol (CHAP).* RFC 1994 (obsoletes RFC 1334), 1996.

Additional References (2)

[RFC2284] L. Blunk, J. Vollbrecht. *PPP Extensible Authentication Protocol (EAP).* RFC 2284, 1998.

[RFC2289] N. Haller, C. Metz, P. Nesser, M. Straw. *A One-Time Password System.* RFC 2289, 1998.

[RFC2341] A. Valencia, M. Littlewood, T. Kolar. Cisco Layer Two Forwarding Protocol (L2F). RFC 2341, 1998.

[RFC2419] K. Sklower, G. Meyer. The PPP DES Encryption Protocol, Version 2 (DESE-bis). RFC 2419 (obsoletes RFC 1969), 1998.

[RFC2420] H. Kummert. The PPP Triple-DES Encryption Protocol (3DESE). RFC 2420, 1998.

[RFC2433] G. Zorn, S. Cobb. *Microsoft PPP CHAP Extensions.* RFC 2433, 1998.

[RFC2637] K. Hamzeh, G. Pall , W. Verthein, J. Taarud, W. Little, G. Zorn. *Point-to-Point Tunneling Protocol (PPTP).* RFC 2637, 1999.

[RFC2661] W. Townsley, A. Valencia, A. Rubens, G. Pall, G. Zorn, B. Palter. *Layer Two Tunneling Protocol (L2TP).* RFC 2661, 1999.

[RFC2828] R. Shirey. *Internet Security Glossary.* RFC 2828, 2000.

Additional References (3)

[RFC3078] G. Pall, G. Zorn. *Microsoft Point to Point Encryption Protocol (MPPE).* RFC 3078, 2001.

[RFC3588] P. Calhoun, et al, Diameter Base Protocol, RFC 3588, 2003

[Riv01a] R. Rivest. *RSA Security Response to Weaknesses in Key Scheduling Algorithm of RC4.* http://www.rsa.com/rsalabs/technotes/wep.html, 2001.

[SM98a] B. Schneier, Mudge. *Cryptanalysis of Microsoft's Point-to-Point Tunneling Protocol (PPTP).* Proceedings of the 5th ACM Conference on Communications and Computer Security, ACM Press, pp. 132-141, 1998.

[SMW99a] B. Schneier, Mudge, D. Wagner. Cryptanalysis of Microsoft's PPTP Authentication Extensions (MSCHAPv2). Counterpane Systems, 1999.

[SIR01a] A. Stubblefield, J. Ioannidis, A. D. Rubin. *Using the Fluhrer, Mantin, and Shamir Attack to Break WEP.* AT&T Labs Technical Report TD-4ZCPZZ, August 2001.

[SMF01a] Adi Shamir, Itsik Mantin and Scott Fluhrer. *Weaknesses in the Key Scheduling Algorithm for RC4.* http://eyetap.org/~rguerra/toronto2001/rc4_ksaproc.pdf, August 2001.

[Walk02] J. Walker. 802.11 Security Series: The Temporal Key Integrity Protocol (TKIP). http://cache-www.intel.com/cd/00/00/01/77/17769_80211_part2.pdf