THE UNIVERSITY OF BRITISH COLUMBIA

Network Security

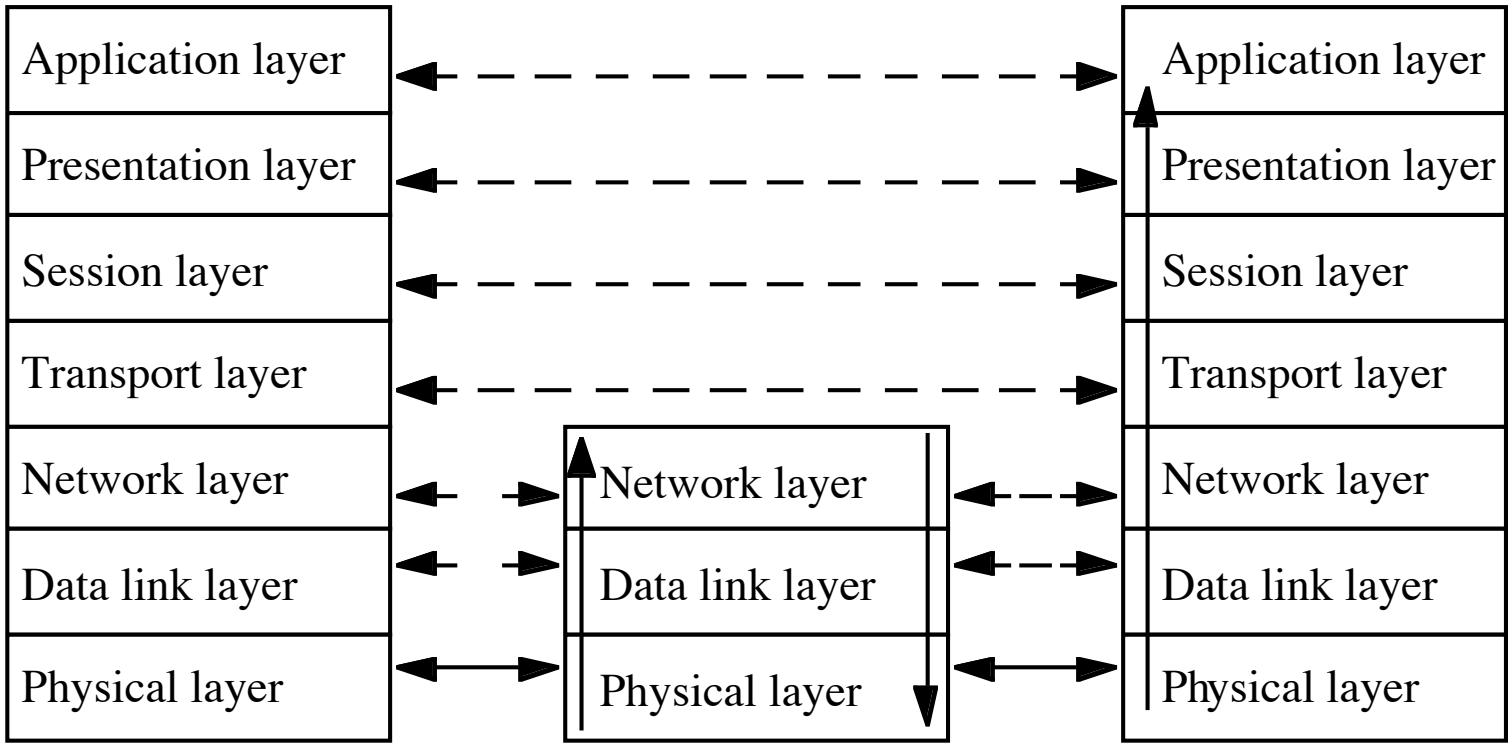
EECE 412

Session 8

Outline

- Link & end-to-end protocols
- SSL/TLS
- WPA

Networks


Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer
Network layer
Data link layer
Physical layer
Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer

Link and End-to-End Protocols

Link Protocol

End-to-End (or E2E) Protocol

Examples

- Telnet protocol
  - Messages between client, server enciphered, and
    - encipherment/decipherment occur only at these hosts
  - End-to-end protocol
- PPP Encryption Control Protocol
  - Host gets message, deciphers it
    - Figures out where to forward it
    - Enciphers it in appropriate key and forwards it
  - Link protocol

Link vs. End-to-end protection

Link encryption

- Can protect headers of packets
- Possible to hide source and destination
  - Note: may be able to deduce this from traffic flows

End-to-end encryption

- Cannot hide packet headers
- Attacker can read source, destination

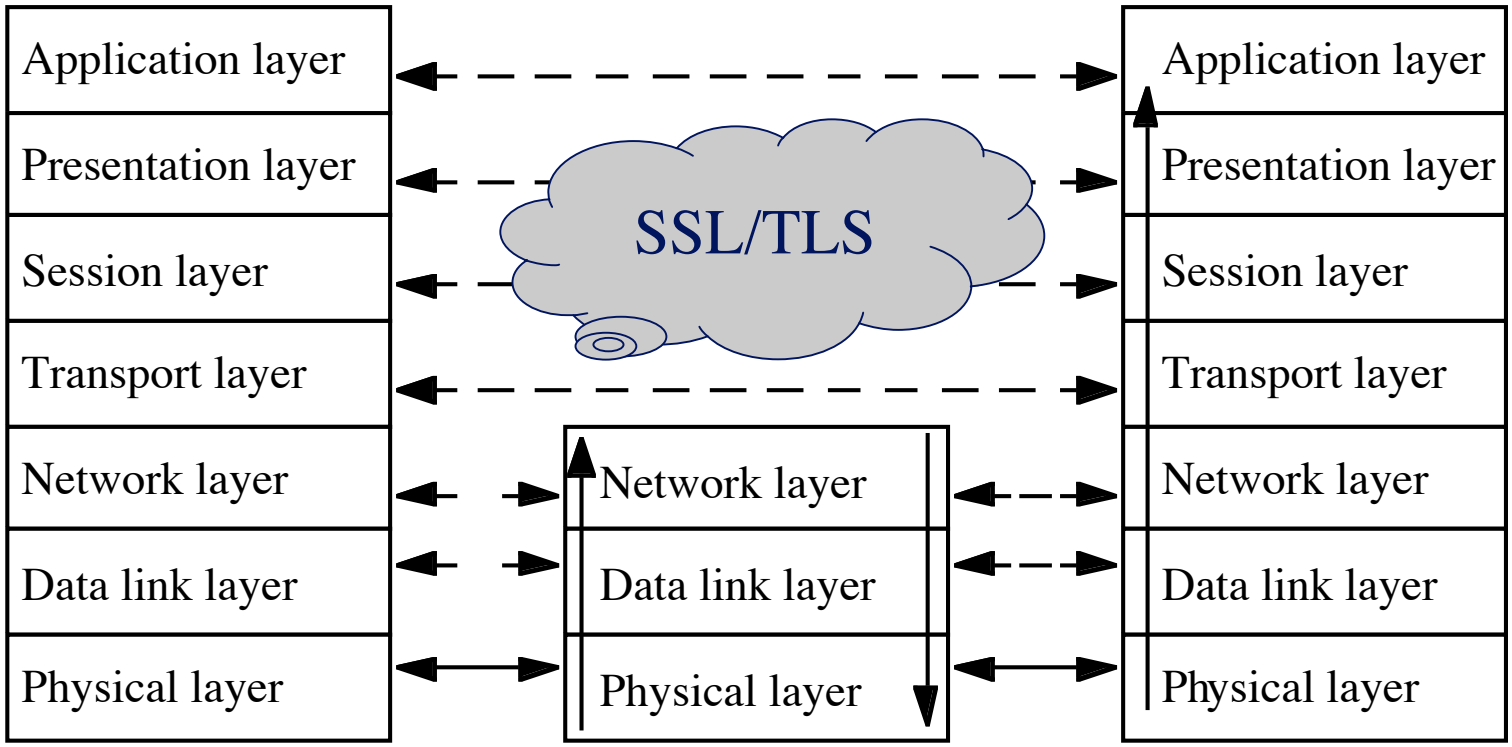
Example Protocols

- Privacy-Enhanced Electronic Mail (PEM)
  - Applications layer protocol
  - Bishop
- Secure Socket Layer (SSL)/Transport Layer Security (TLS)
  - Transport layer protocol
- IP Security (IPSec)
  - Network layer protocol
  - Bishop
- Wi-Fi Protected Access (WPA)
  - Data layer protocol
  - Today session

THE UNIVERSITY OF BRITISH COLUMBIA

Secure Socket Layer (SSL)
a.k.a.
Transport Layer Security (TLS)

Networks


Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer
SSL/TLS
Network layer
Data link layer
Physical layer
Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer

SSL Session

Association between two peers

- Two peers may have several sessions
- Information for each association:
  - Unique session identifier
  - Peer's X.509v3 certificate, if needed
  - Compression method
  - Cipher spec for cipher and MAC
  - "Master secret" shared with peer
    - 48 bits

SSL Connection

Describes how data exchanged with peer in a session

- Several connections per session
- Information for each connection
  - Random data
  - Write keys (used to encipher data)
  - Write MAC key (used to compute MAC)
  - Initialization vectors (IVs) for ciphers, if needed
  - Sequence numbers

Supporting Crypto

- All parts of SSL use them
- Initial phase: public key system exchanges keys
  - Messages enciphered using classical ciphers, and MACed
  - Only certain combinations allowed
    - Depends on algorithm for key exchange cipher
  - Key exchange (a.k.a., interchange) algorithms:
    - RSA
    - Diffie-Hellman
    - Fortezza

RSA: Cipher, MAC Algorithms

Interchange cipher	***Classical cipher***	***MAC Algorithm***
RSA, key ≤ 512 bits	*none*	MD5, SHA
	RC4, 40-bit key	MD5
	RC2, 40-bit key, CBC mode	MD5
	DES, 40-bit key, CBC mode	SHA
RSA	*None*	MD5, SHA
	RC4, 128-bit key	MD5, SHA
	IDEA, CBC mode	SHA
	DES, CBC mode	SHA
	DES, EDE mode, CBC mode	SHA

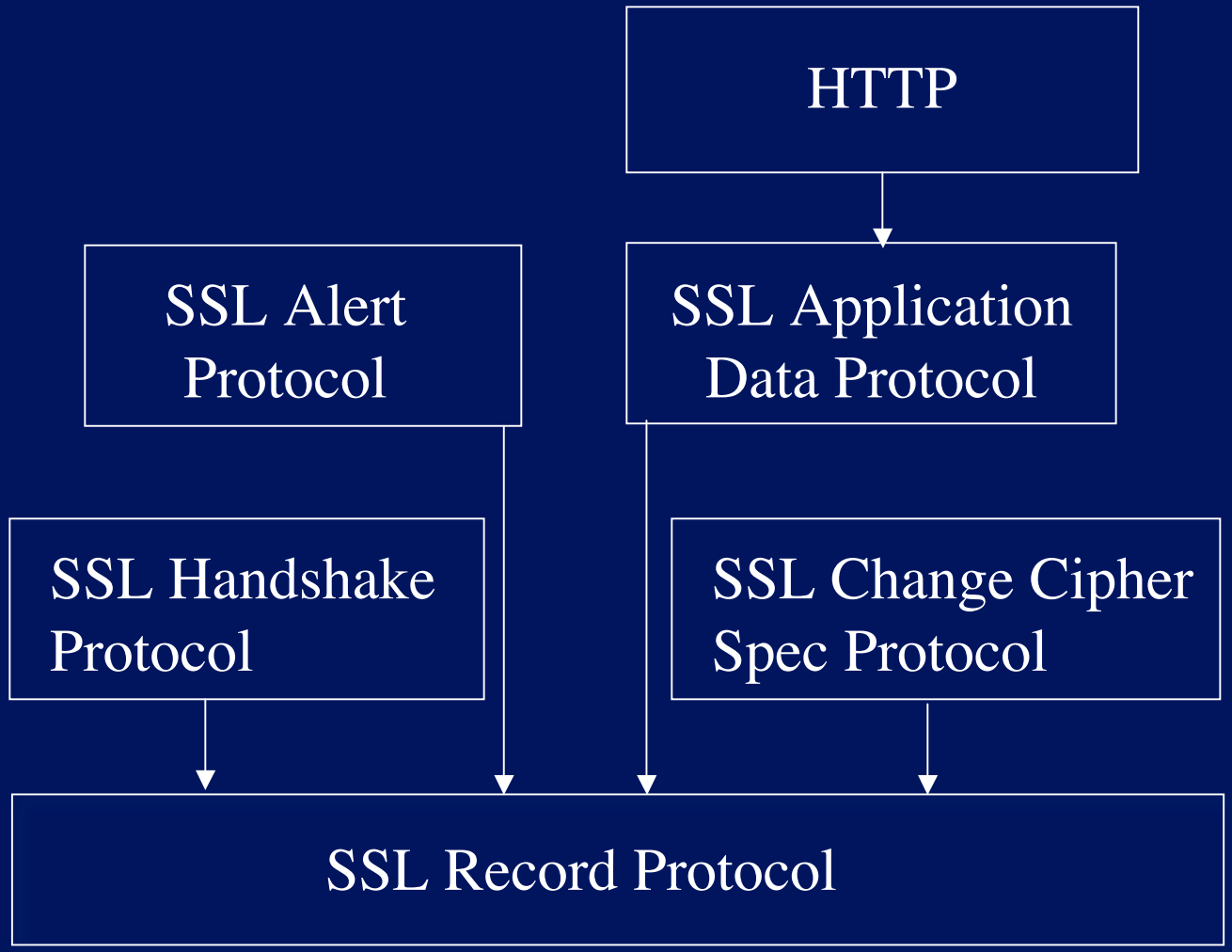
D-H: Cipher, MAC Algorithms

Interchange cipher	*Classical cipher*	*MAC Algorithm*
Diffie-Hellman, DSS Certificate	DES, 40-bit key, CBC mode	SHA
	DES, CBC mode	SHA
	DES, EDE mode, CBC mode	SHA
Diffie-Hellman, key ≤ 512 bits RSA Certificate	DES, 40-bit key, CBC mode	SHA
	DES, CBC mode	SHA
	DES, EDE mode, CBC mode	SHA

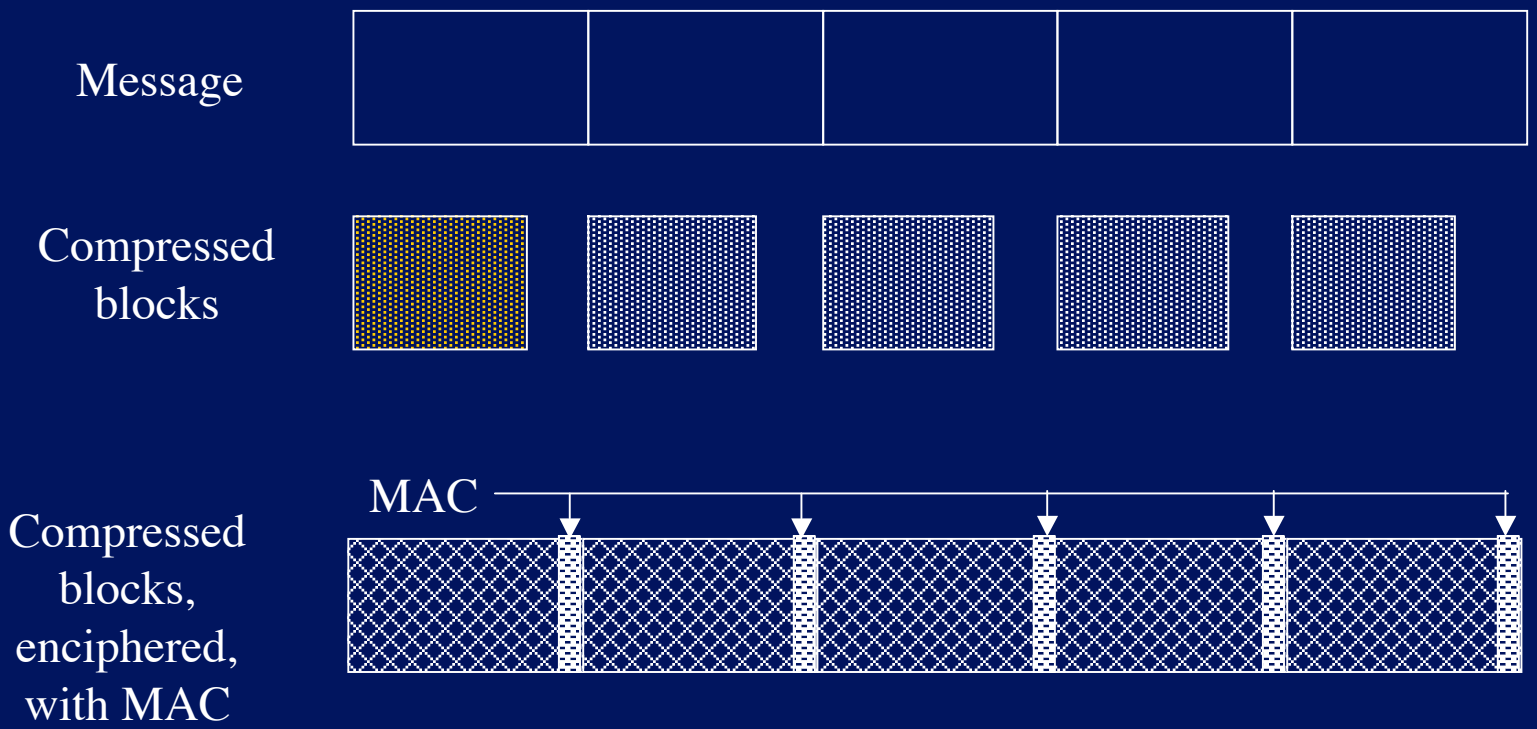
Fortezza: Cipher, MAC Algorithms

Interchange cipher	***Classical cipher***	***MAC Algorithm***
Fortezza key exchange	*none*	SHA
	RC4, 128-bit key	MD5
	Fortezza, CBC mode	SHA

SSL Protocols



SSL Record Layer

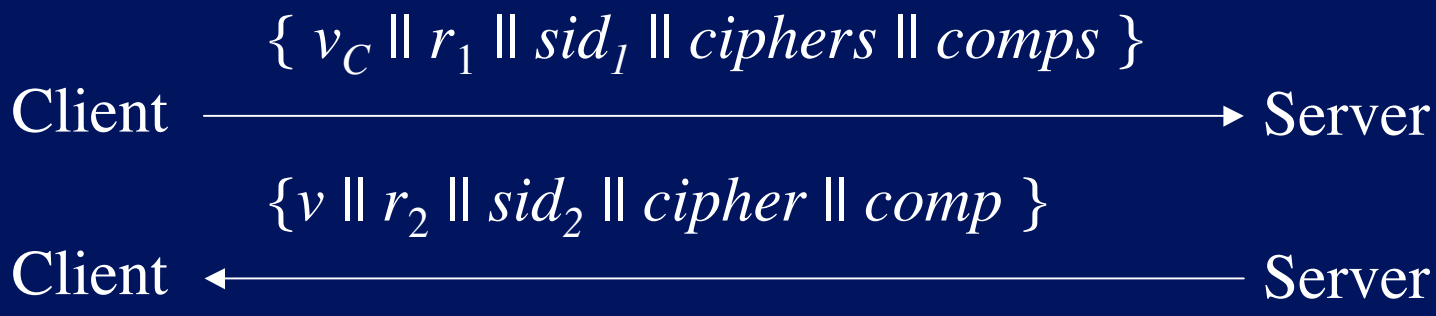


Overview of Handshake Rounds

1. Create SSL connection between client, server
2. Server authenticates itself
3. Client validates server, begins key exchange
4. Acknowledgments all around

Handshake Round 1

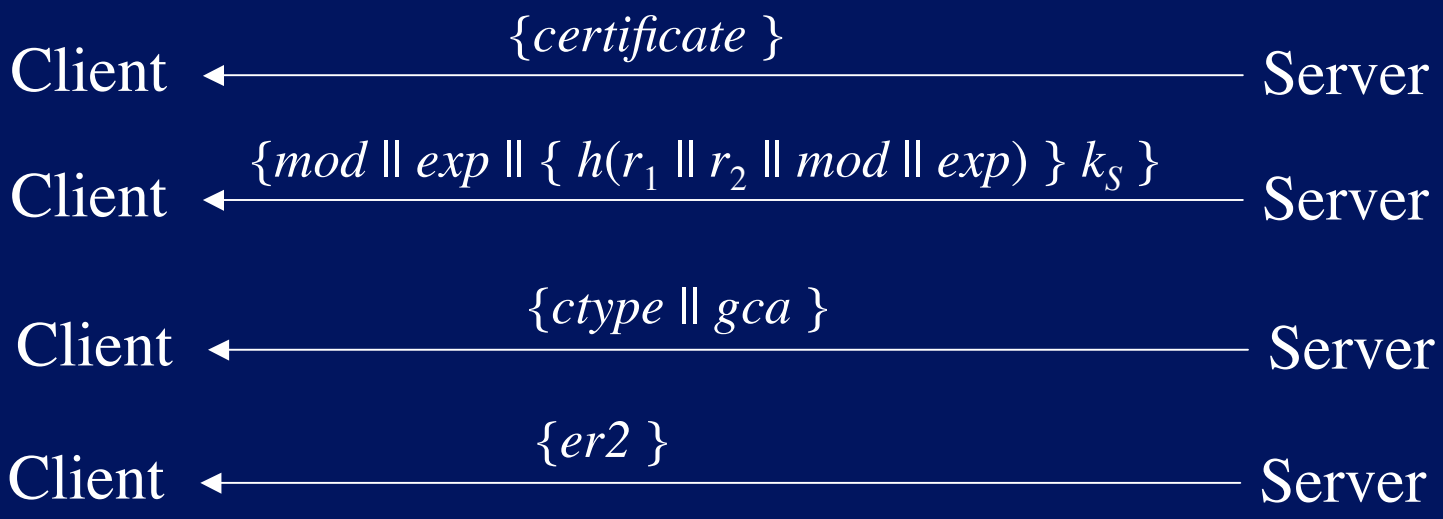
Purpose: Create SSL connection between client, server



v_C	Client's version of SSL
v	Highest version of SSL that Client, Server both understand
r_1, r_2	nonces (timestamp and 28 random bytes)
sid_1	Current session id (0 if new session)
sid_2	Current session id (if s1 = 0, new session id)
ciphers	Ciphers that client understands
comps	Compression algorithms that client understand
cipher	Cipher to be used
comp	Compression algorithm to be used

Handshake Round 2

Purpose: Server authenticates itself



Note: if Server not to authenticate itself, only last message sent; third step omitted if Server does not need Client certificate

mod	*public key modulus*
exp	*public key exponent*
k_S	Server's private key
ctype	Certificate type requested (by cryptosystem)
gca	"Good" certification authorities
er2	End round 2 message

Handshake Round 3

Purpose: Client validates server, begins key exchange

Both Client, Server compute master secret *master*:

master = MD5(*pre* || SHA('A' || *pre* || r_1 || r_2) ||
MD5(*pre* || SHA('BB' || *pre* || r_1 || r_2) ||
MD5(*pre* || SHA('CCC' || *pre* || r_1 || r_2)

msgs	Concatenation of previous messages sent/received this handshake
opad, *ipad*	As above

Handshake Round 4

Client sends "change cipher spec" message using that protocol

Client ⟶ Server

{ *h*(*master* || *opad* || *h*(*msgs* || 0x434C4E54 || *master* || *ipad*)) }

Client ⟶ Server

Server sends "change cipher spec" message using that protocol

Client ⟵ Server

{ *h*(*master* || *opad* || *h*(*msgs* || *master* | *ipad*)) }

Client ⟵ Server

msgs Concatenation of messages sent/received this handshake in *previous* rounds (does notinclude these messages)

opad, *ipad*, *master* As above

SSL Change Cipher Spec Protocol

- Send single byte
- In handshake, new parameters considered "pending" until this byte received

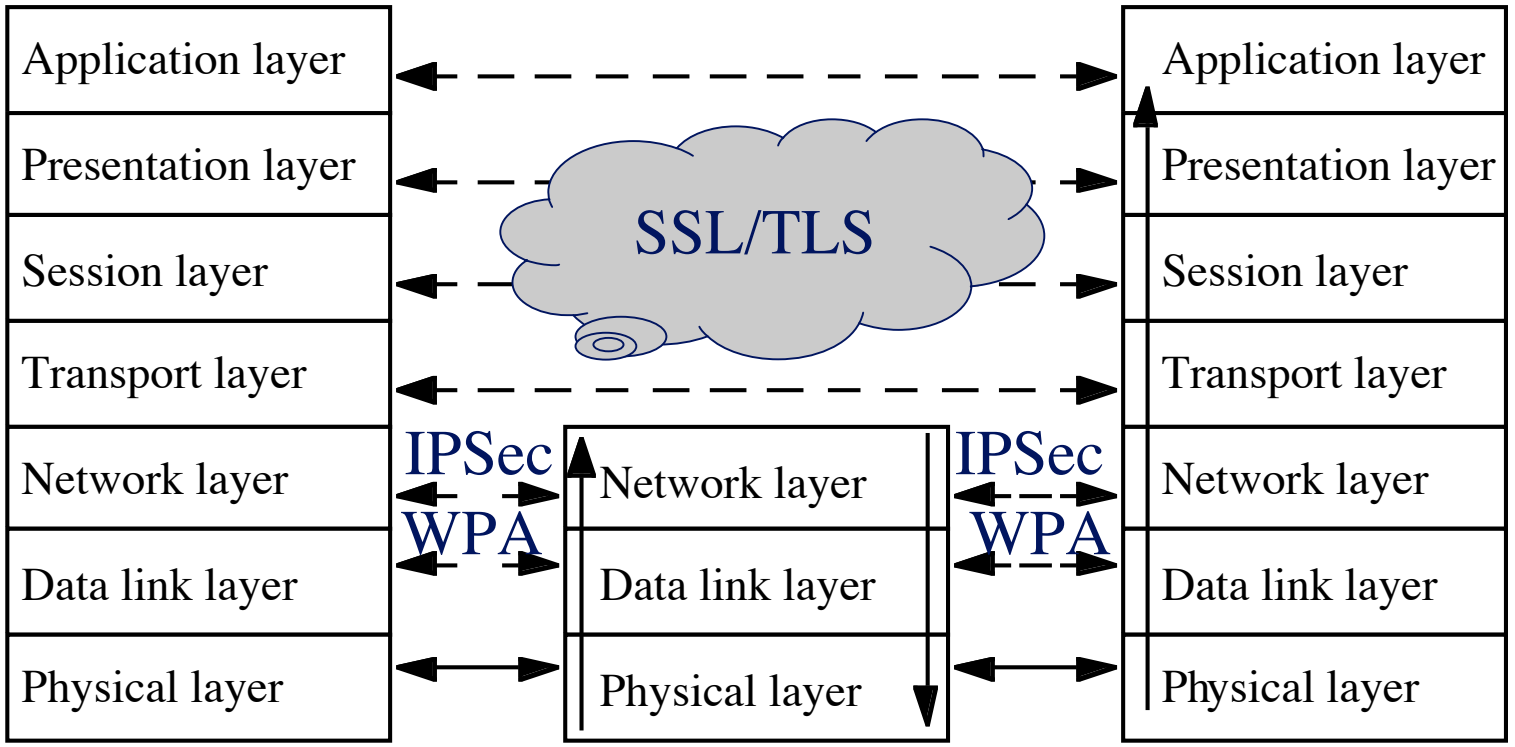
SSL Alert Protocol

- Closure alert
  - Sender will send no more messages
- Error alerts
  - Warning
    - connection remains open
  - Fatal error
    - connection torn down as soon as sent or received

THE UNIVERSITY OF BRITISH COLUMBIA

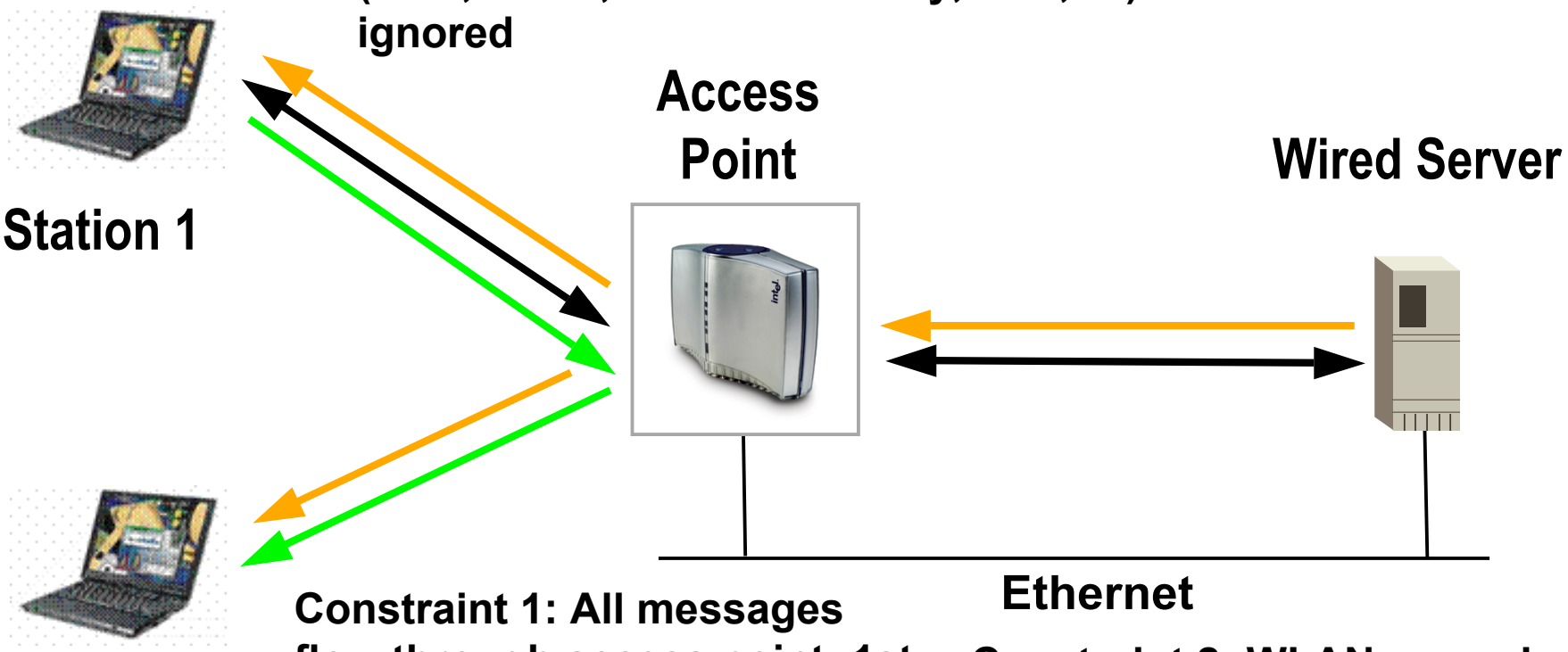
Wi-Fi Protected Access (WPA)

Where is WPA?


Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer
SSL/TLS
IPSec
WPA
Network layer
Data link layer
Physical layer
IPSec
WPA
Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer

Design Constraints

Constraint 3: Multicast integral to modern networking (ARP, UPnP, Active Directory, SLP, …) and cannot be ignored



Constraint 1: All messages flow through access point; 1st generation AP MIP budget = 4 Million instructions/sec

Constraint 2: WLAN uses short range radios, so APs must be ubiquitous, so lowest cost

Wireless Security Overview

Paul Cychosz

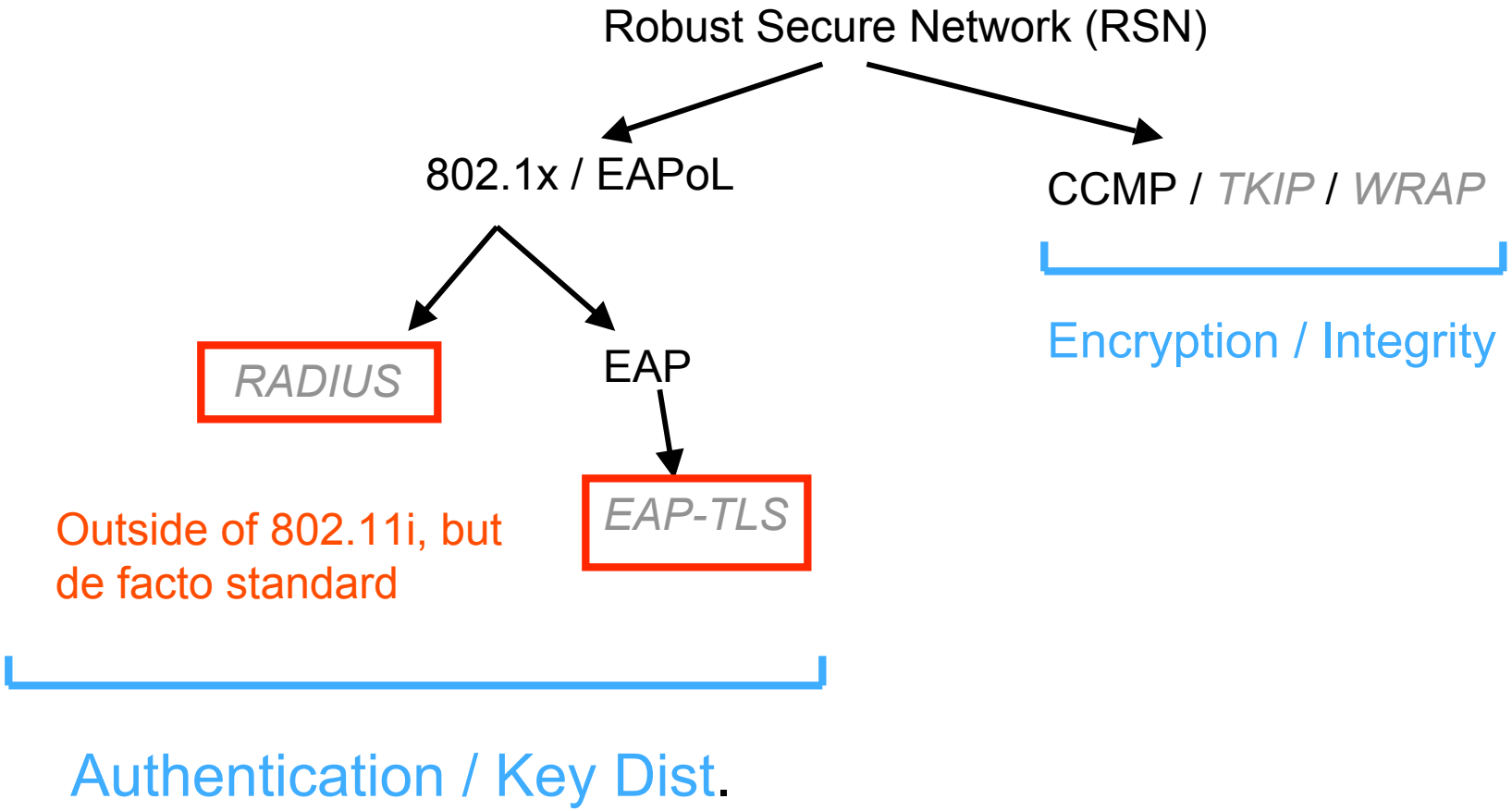
March 2005

802.11i

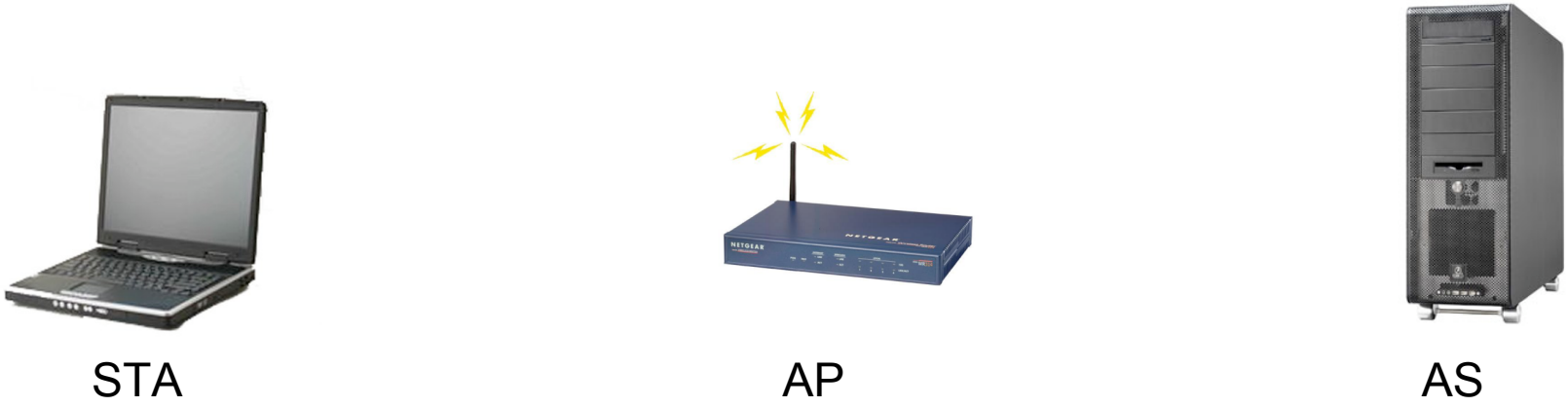
Terms:

- 802.1x: Authentication standard
- RADIUS: Authentication Server
- EAP: Extensible Authentication Protocol
- CCMP: Encryption based on AES counter mode with CBC-MAC

802.11i Parts



802.11i – First half



Capability Discovery

802.1x Authentication

802.1x Key Management

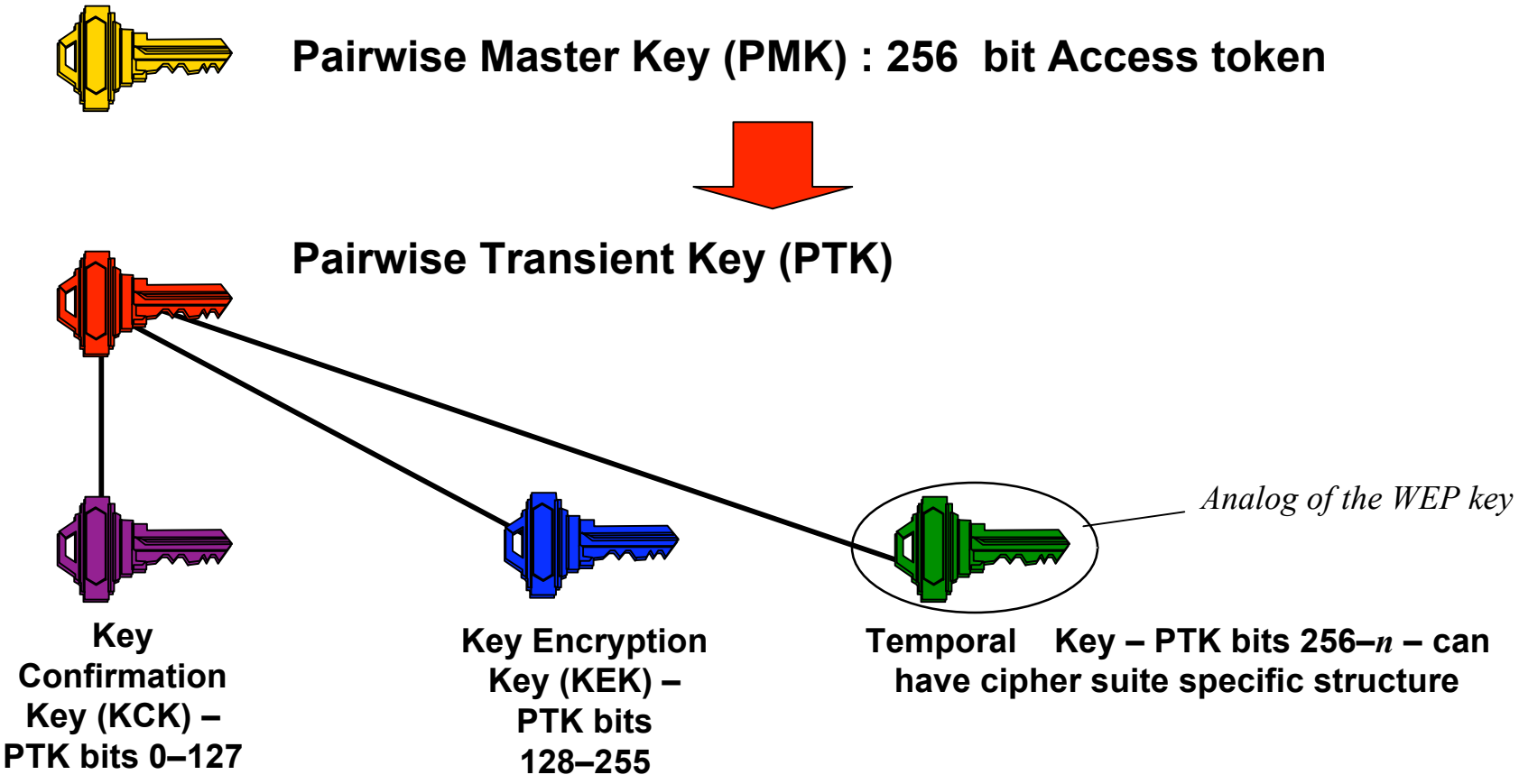
Keygen & Distribution

Encryption + Additional handshaking

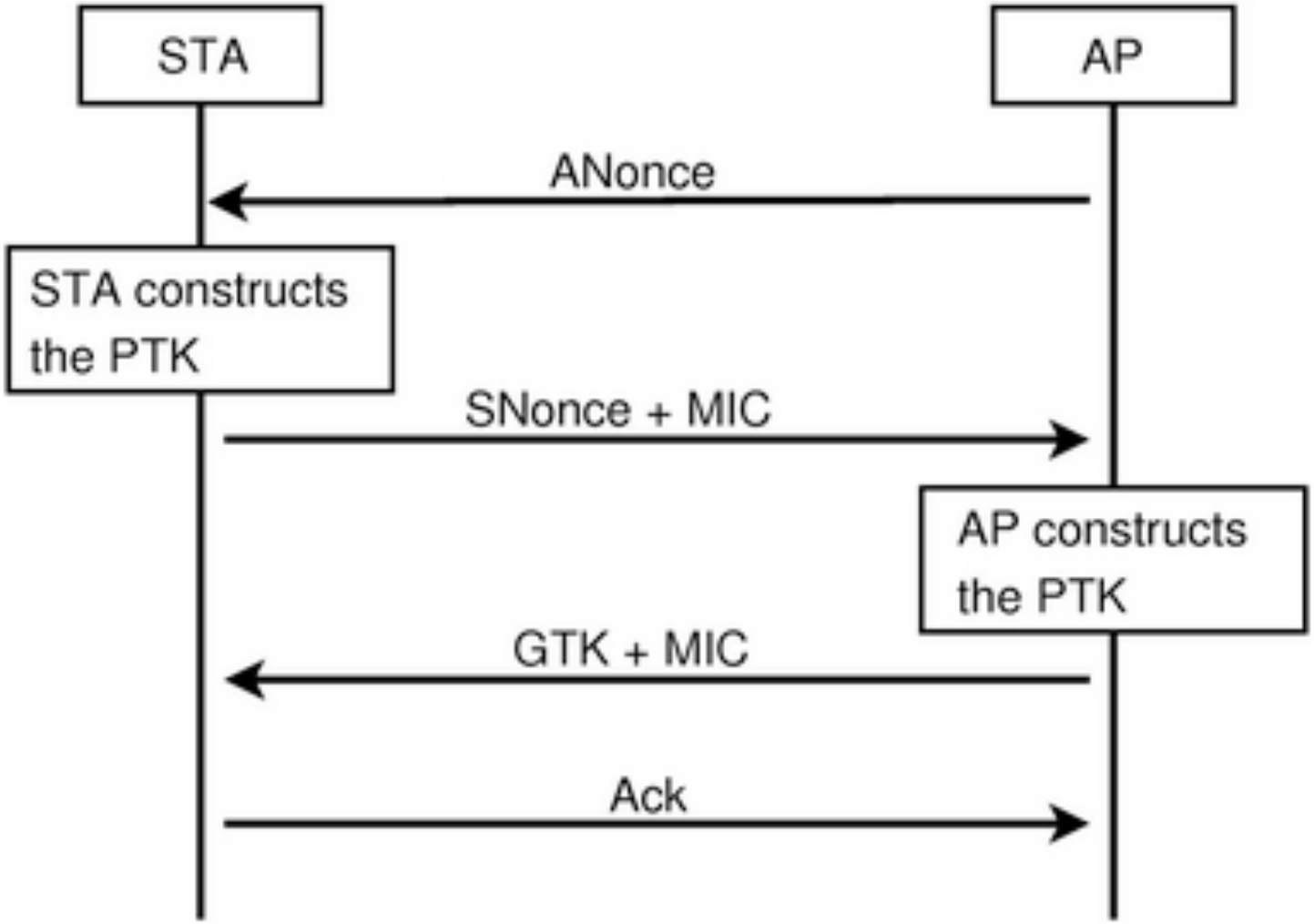
THE UNIVERSITY OF BRITISH COLUMBIA

WPA Key Managment

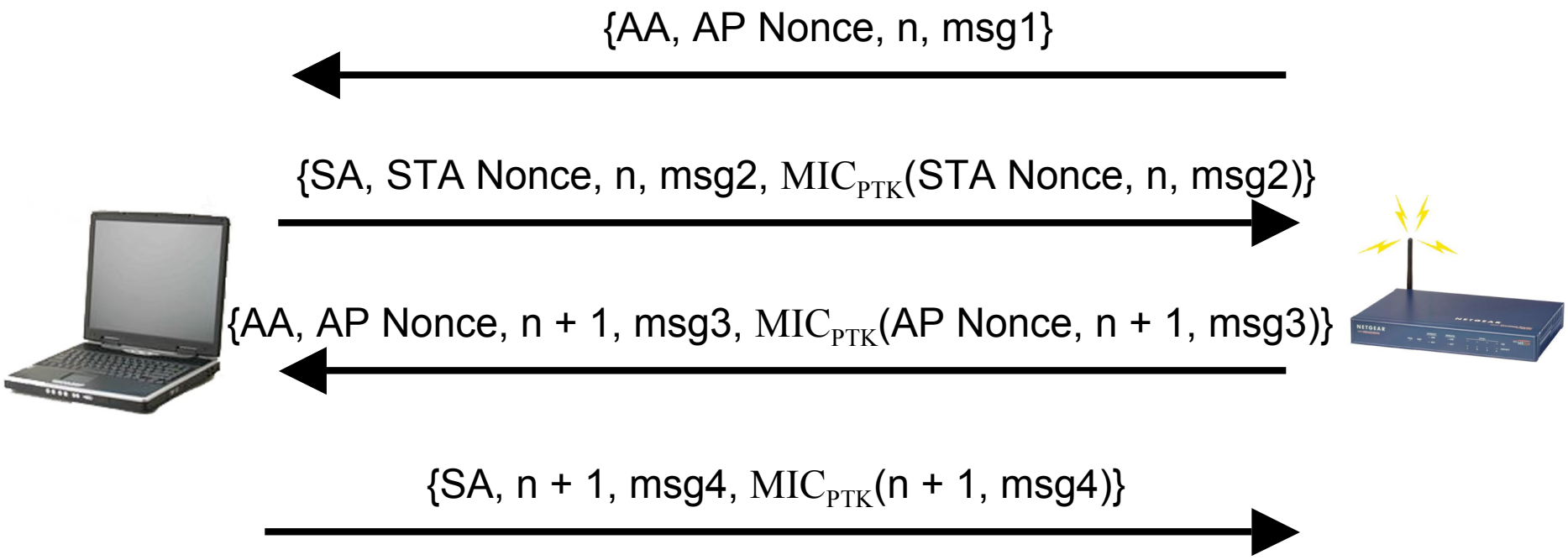
802.11i Pairwise Key Hierarchy



Session Key Establishment



Handshake Details



Message 1

- not protected, doesn't matter though

AP → STA: {AA, AP Nonce, n, msg1}

AA: MAC Address of AP

AP Nonce: random value

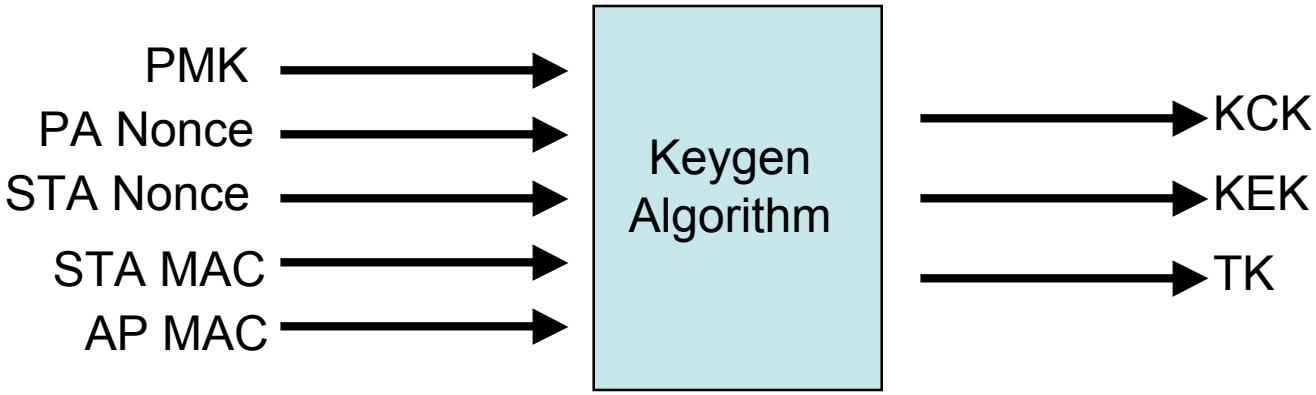
n: sequence identifier

msg1: PMKID = HMAC-SHA1-128(PMK, "PMK Name" || AA || SPA).

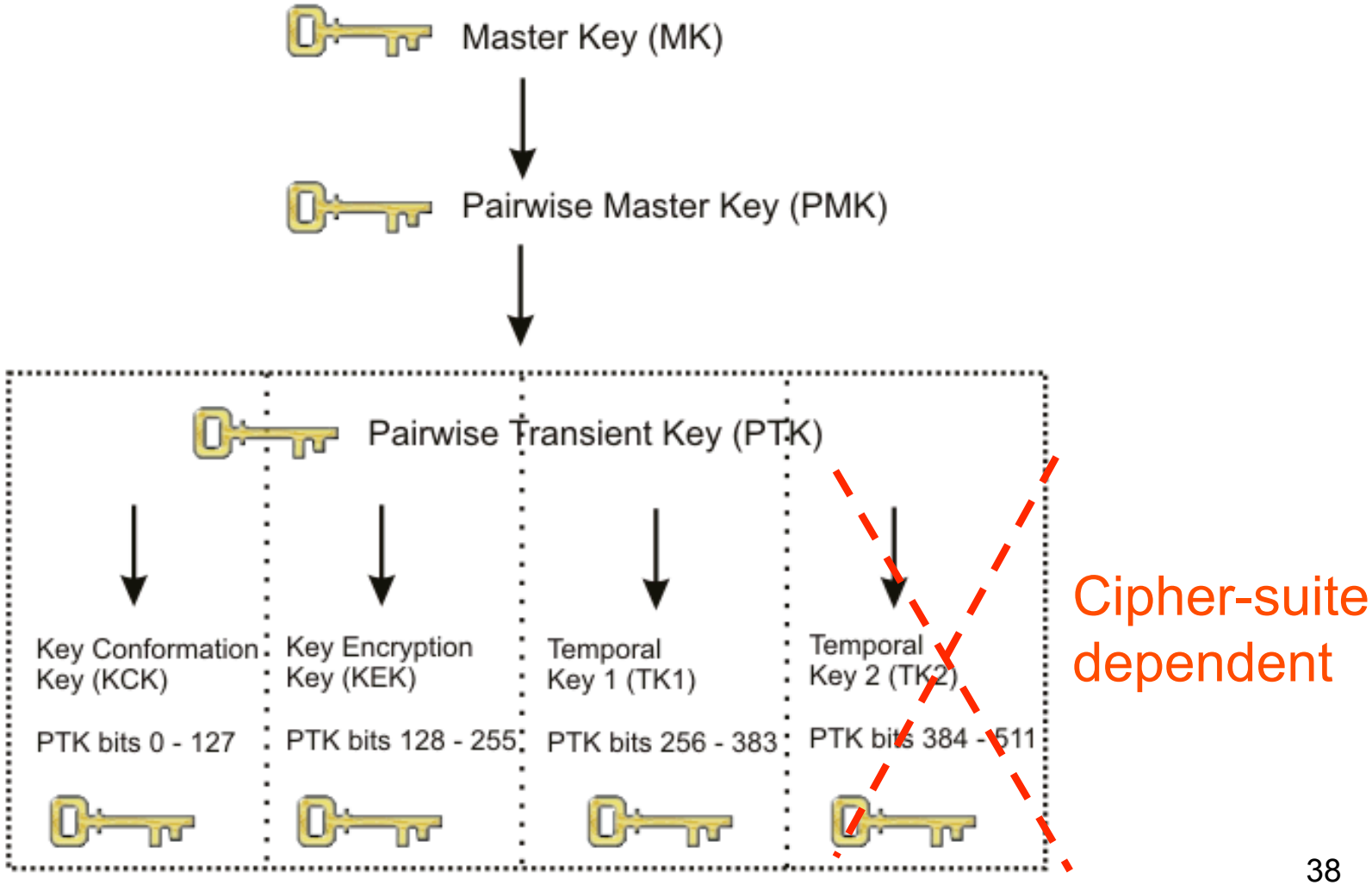
•Client uses AP Nonce and PMK to compute PTK

PTK = 802.11i-PRF(
PMK,
min(AP Nonce, STA Nonce) || max(AP nonce, STA Nonce) ||
min(AP MAC Addr, STA MC Addr) || max(AP MAC Addr, STA MAC Addr))

802.11i – What's PTK?



802.11i – Key Heirarchy



Message 2

STA → AP: {SA, STA Nonce, n, msg2, $\mathrm{MIC}_{\mathrm{PTK}}$(STA Nonce, n, msg2)}

SPA: MAC Address of STA

SNonce: random value

n: sequence identifier, matches msg1

msg2: RSN IE of STA

- AP uses STA Nonce and PMK to compute PTK

Message 3

AP → STA: {AA, AP Nonce, n + 1, msg3, MIC_{PTK}(AP Nonce, n + 1, msg3)}

AA: MAC Address of AP

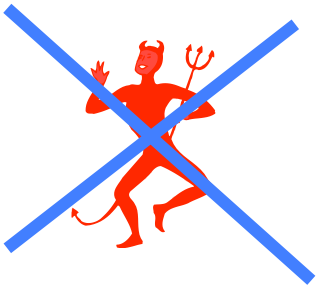
AP Nonce: random value again

n: sequence identifier, to match msg4

msg3: Informs STA that TK ready to use, RSN IE of AP.

MIC: to verify the above. Silently discarded if MIC fails.

Verifies no MITM attack happening



Message 4

STA → AP: {SPA, n + 1, msg4, MIC_{PTK}(n + 1, msg4)}

SPA: MAC Address of STA

n: sequence identifier, to match msg3

MIC: to verify the above. Silently discarded if MIC fails.

- This message dropped in some implementations.
- Only kept for convention

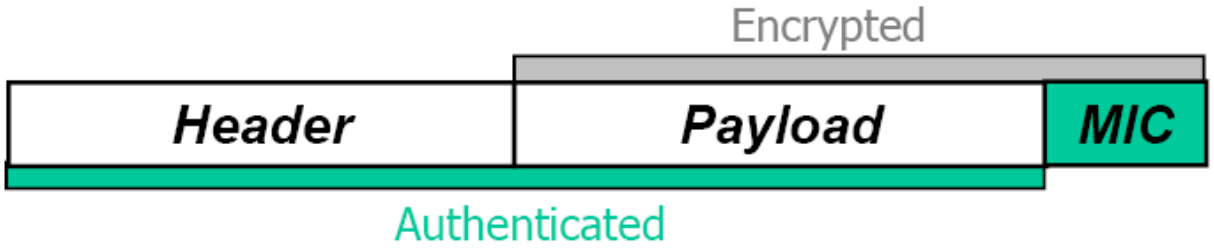
WPA Data Protection

AES-CCMP

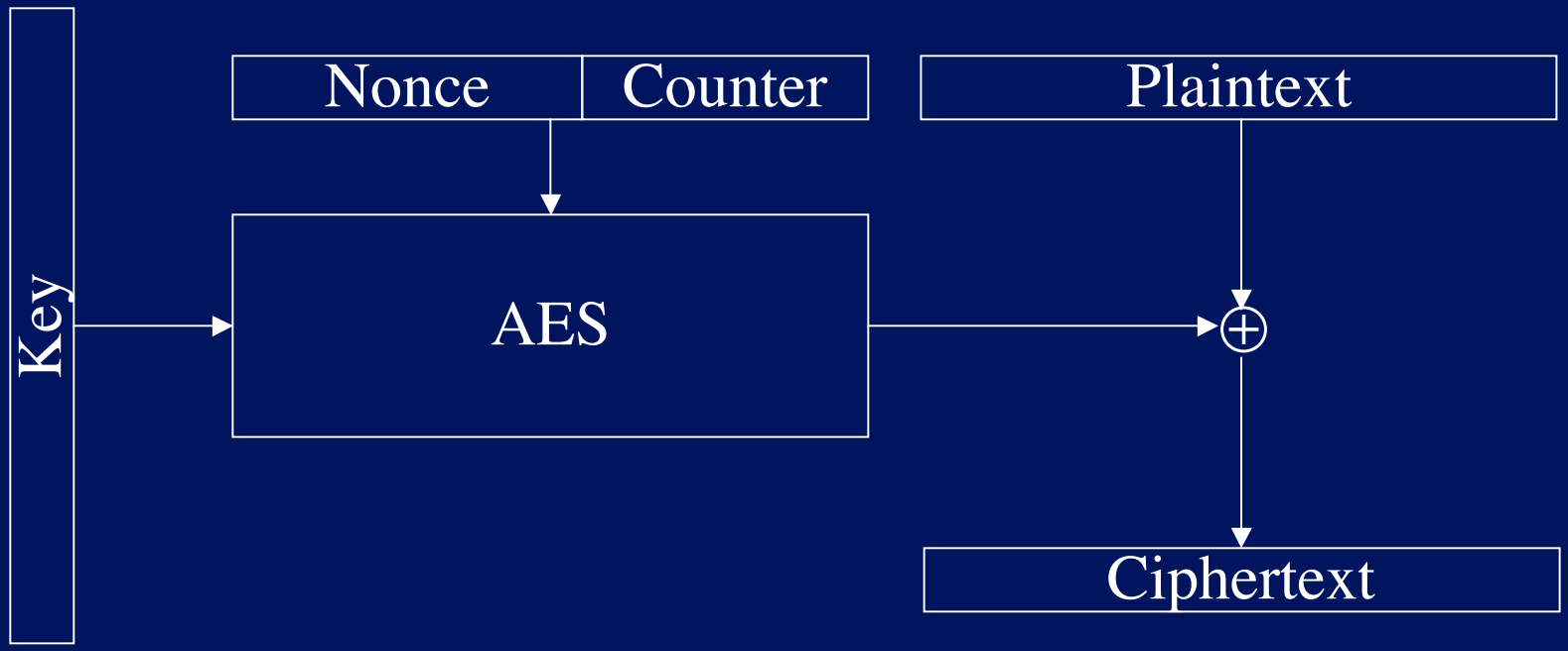
- New encryption based on AES

" NIST estimates that a machine that can break 56-bit DES key in 1 second would take about 149 trillion years to crack a 128-bit AES key (unless someone is very lucky)"

- CCMP: Counter Mode with Cipher Block Chaining Message Authentication Code Protocol
  - Confidentiality protection: counter mode
  - Authenticity and integrity protection: CBC-MAC

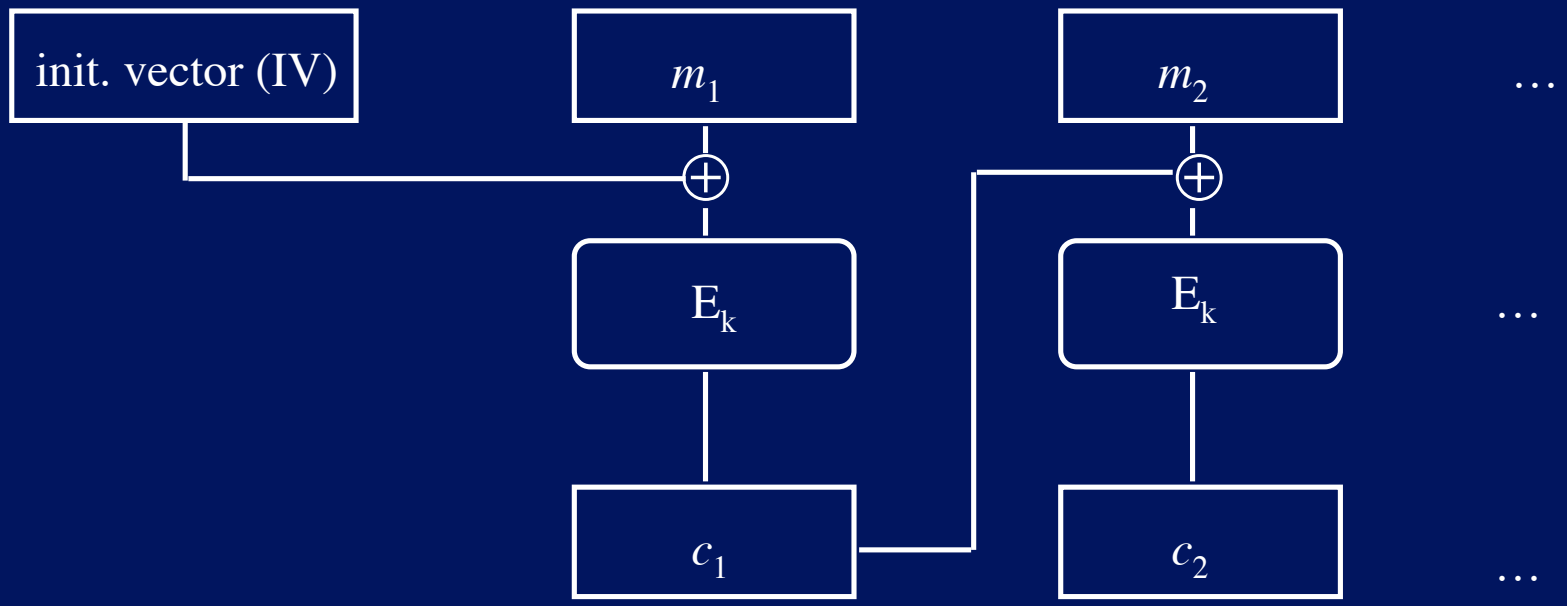


AES-CCMP: Counter Mode Encryption



Cipher Block Chaining (CBC)

$M = m_1 \mid m_2 \mid \ldots \mid m_n$



$C = IV \mid c_1 \mid c_2 \mid \ldots \mid c_n$

Integrity and authenticity Protection

MIC: CBC-MAC / per packet algorithm

- 128-bit generation, but only take first 64-bits
- XOR blocks, hence “block-chaining”
- MIC computed on packet header
- MIC then encrypted (using IV = 0, CTR mode) and appended to payload