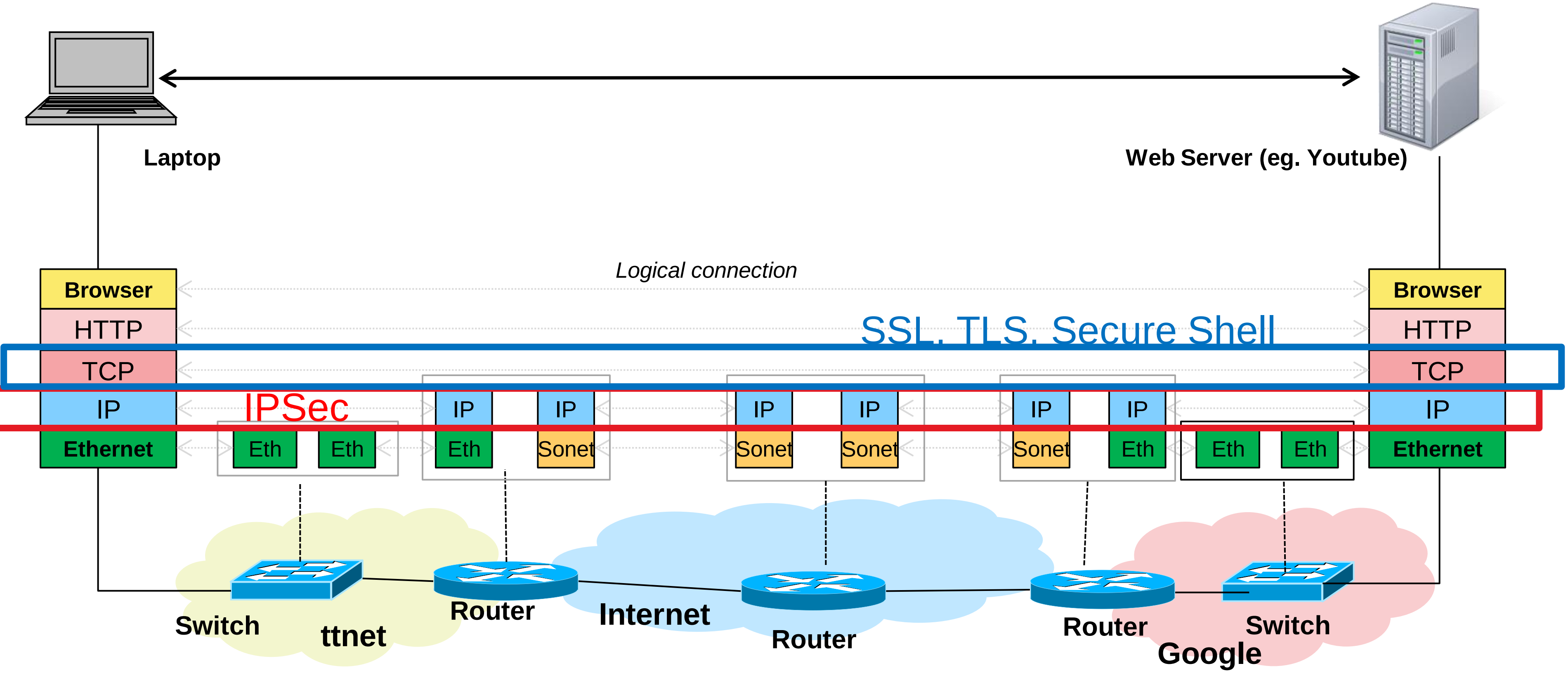
Securing the Internet

Irfan Ali

aliirfan04@yahoo.com

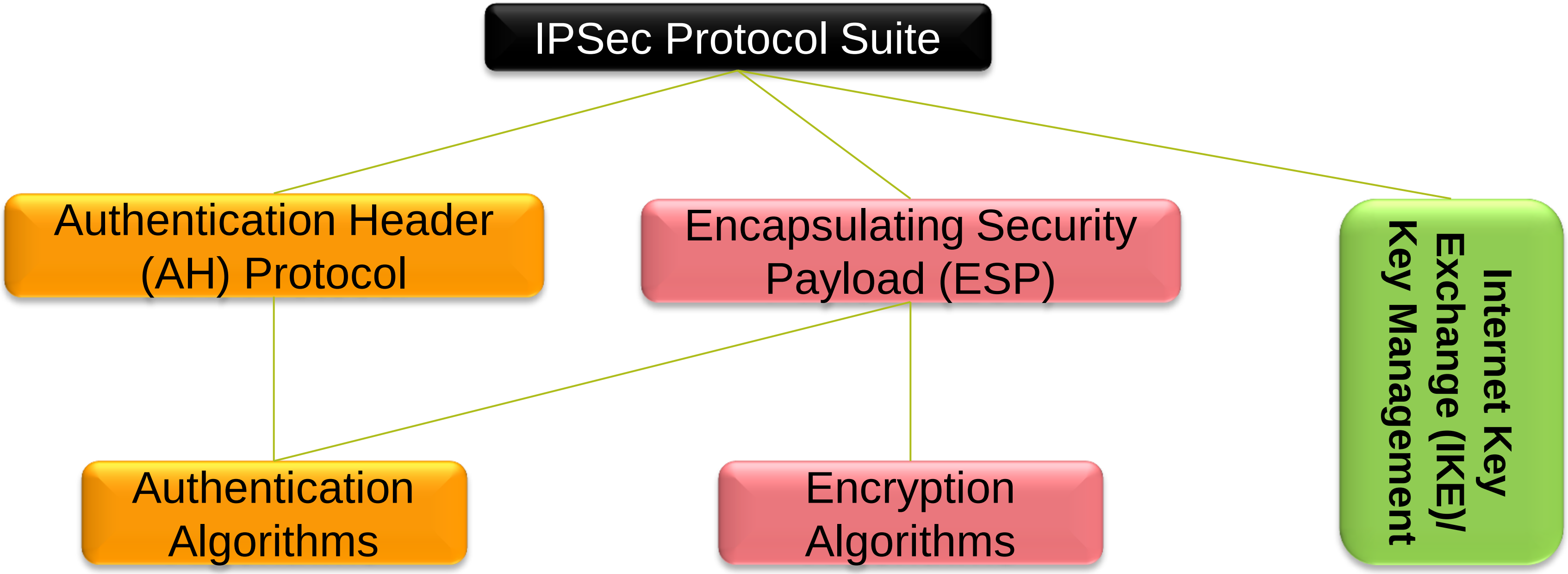
IPSec, SSL, TLS, Secure Shell, VPN



- IPv4 was created without paying attention to security, since it was created for open-collaboration between universities
- For providing security end-to-end Transport layer security mechanism needed to be created: SSL, TLS secure shell.
- IPSec is at the network layer and protects all traffic using IP.
- IPSec support is compulsory support for IPv6.

IPSec	Internet Protocol Security
SSL	Secure Socket Layer
TLS	Transport Layer Security

IPSec Components

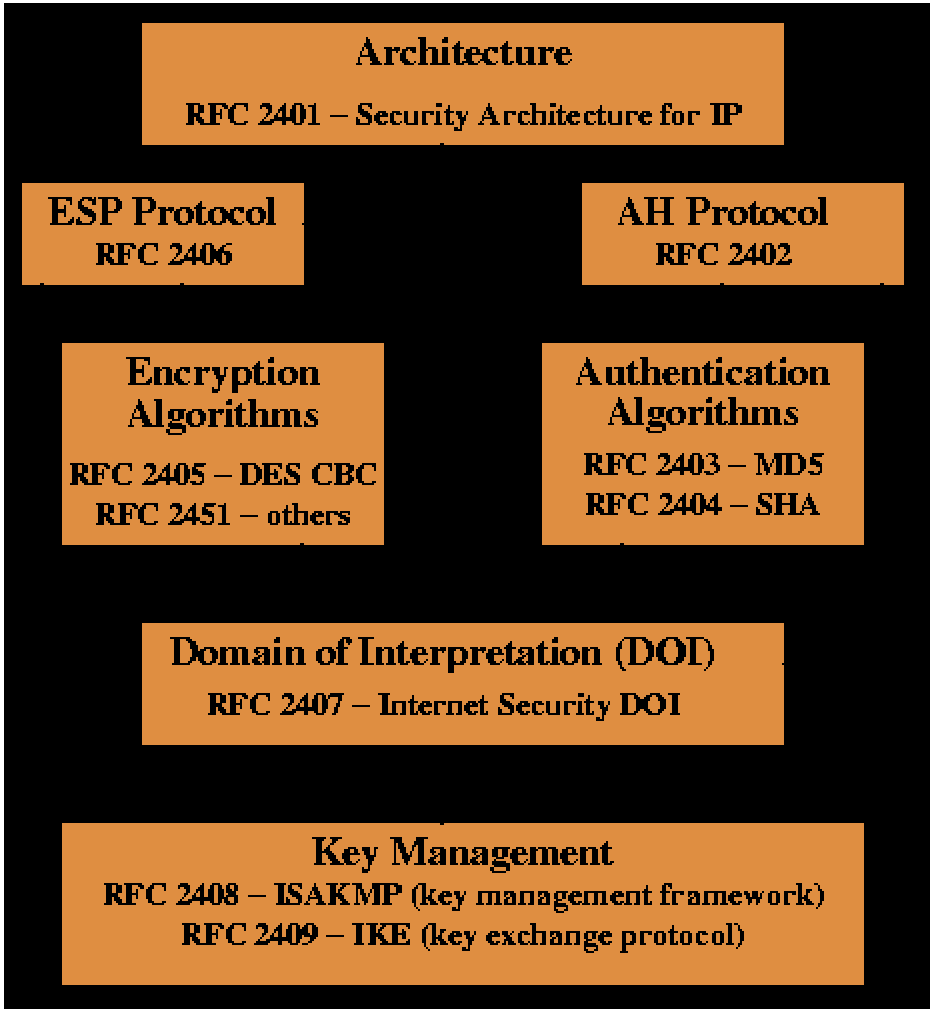

IPSec Protocol Suite
Authentication Header (AH) Protocol
Encapsulating Security Payload (ESP)
Internet Key Exchange (IKE)/ Key Management
Authentication Algorithms
Encryption Algorithms

- **IPSec Authentication Header (AH):** This protocol provides authentication services for IPSec. What this means is that it allows the recipient of a message to verify that the supposed originator of a message was in fact the one that sent it. It also allows the recipient to verify that none of the data in the datagram has been changed by any intermediate devices en route. It also provides protection against so-called "replay" attacks, where a message is captured by an unauthorized user and re-sent.

- **Encapsulating Security Payload (ESP):** The Authentication Header ensures integrity of the data in datagram, but not its privacy. When the information in a datagram is "for your eyes only", it can be further protected using the ESP protocol, which encrypts the payload of the IP datagram.

Source: http://www.tcpipguide.com/free/t_IPSecGeneralOperationComponentsandProtocols-2.htm

IPSec IETF Specifications



Architecture

RFC 2401 – Security Architecture for IP

ESP Protocol

RFC 2406

AH Protocol

RFC 2402

Encryption Algorithms

RFC 2405 – DES CBC

RFC 2451 – others

Authentication Algorithms

RFC 2403 – MD5

RFC 2404 – SHA

Domain of Interpretation (DOI)

RFC 2407 – Internet Security DOI

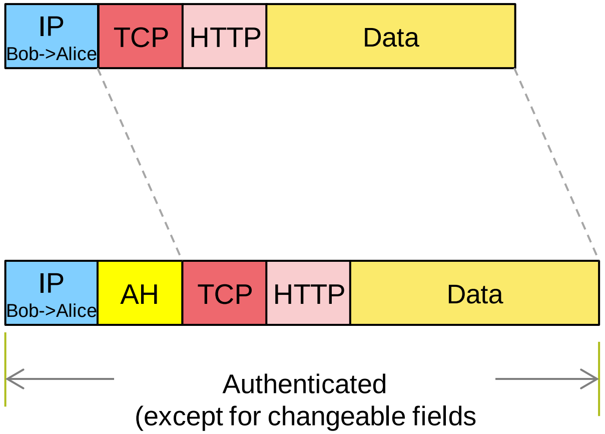
Key Management

RFC 2408 – ISAKMP (key management framework)

RFC 2409 – IKE (key exchange protocol)

IPv4 AH and ESP Packet headers

IPv4 before applying AH



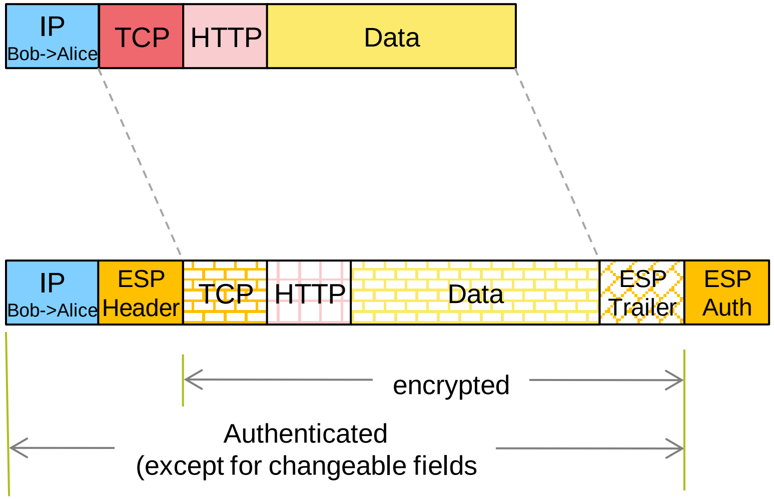
IP Bob->Alice | TCP | HTTP | Data

IP Bob->Alice | AH | TCP | HTTP | Data

Authenticated
(except for changeable fields

IPv4 after applying AH

IPv4 before applying ESP



IP Bob->Alice | TCP | HTTP | Data

IP Bob->Alice | ESP Header | TCP | HTTP | Data | ESP Trailer | ESP Auth

encrypted

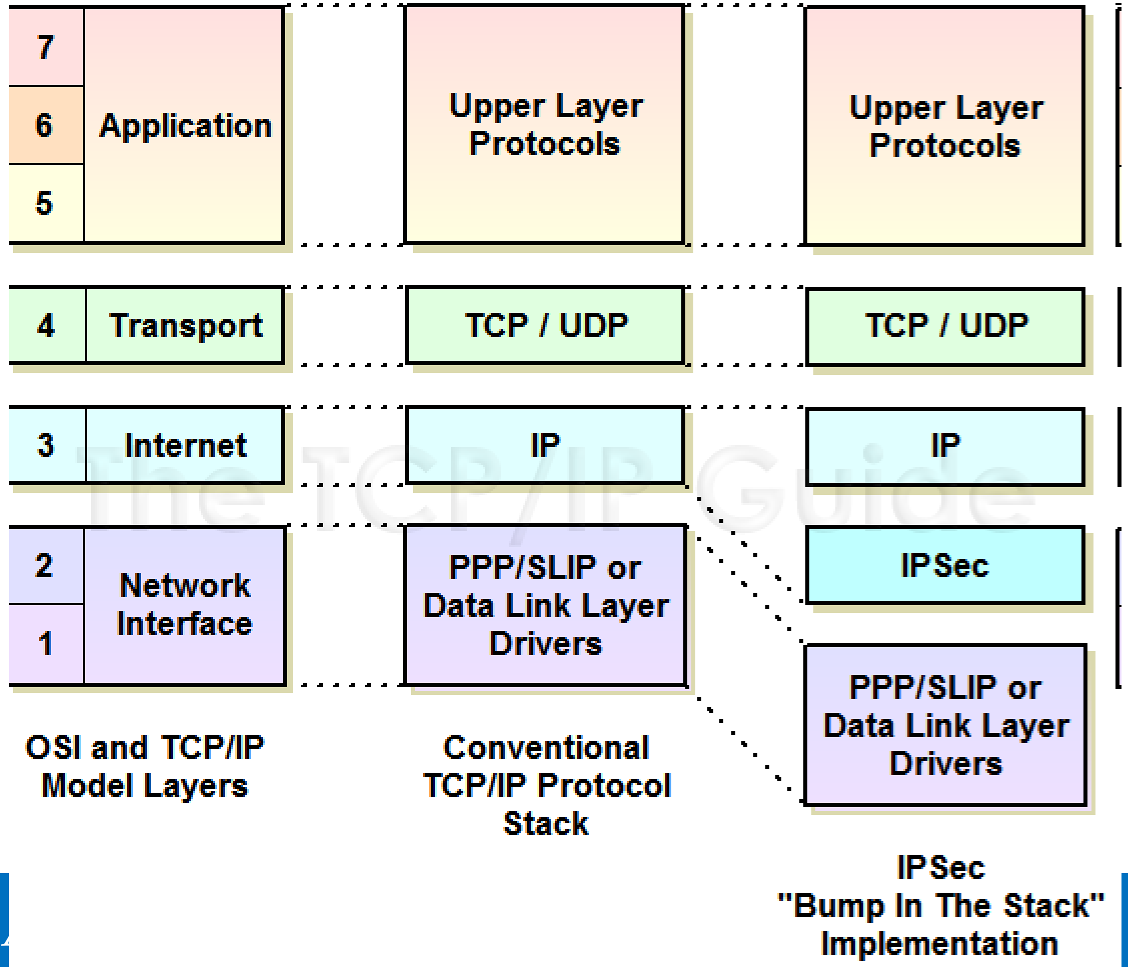
Authenticated
(except for changeable fields

IPv4 after applying ESP

- These are in “transport mode”. (Transport and Tunnel mode covered later in slides)

IPSec Implementation (1 of 2)

1. **IP Stack Integration:** Under ideal circumstances, we would integrate IPSec's protocols and capabilities directly into IP itself.With IPv4, integration would require making changes to the IP implementation on each device, which is often impractical (to say the least!). Possible for IPv6
2. **“Bump In The Stack” (BITS) Architecture:**

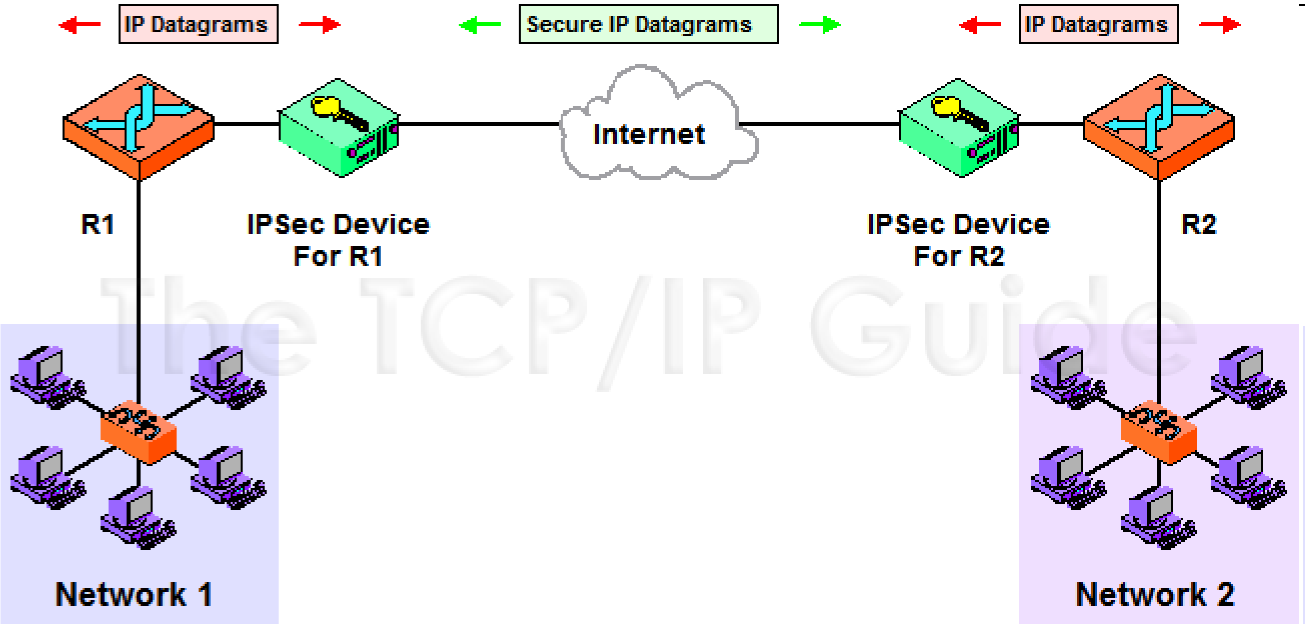


7
6
5
Application
Upper Layer Protocols
Upper Layer Protocols
4
Transport
TCP / UDP
TCP / UDP
3
Internet
IP
IP
IPSec
2
1
Network Interface
PPP/SLIP or Data Link Layer Drivers
PPP/SLIP or Data Link Layer Drivers
The TCP/IP Guide
OSI and TCP/IP Model Layers
Conventional TCP/IP Protocol Stack
IPSec "Bump In The Stack" Implementation

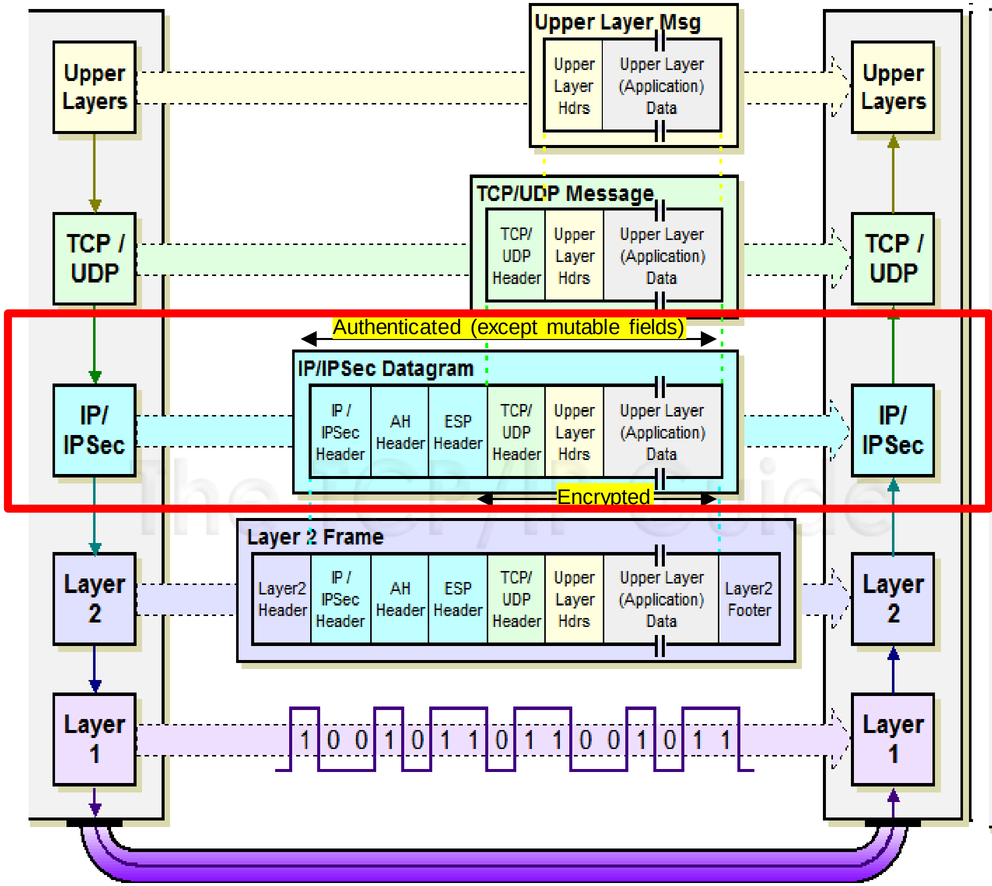
In this technique, IPSec is made a separate architectural layer between IP and the data link layer. IPSec intercepts IP datagrams as they are passed down the protocol stack, processes them to provide security, and then passes them through to the data link layer.

IPSec Implementation (2 of 2)

3. “Bump In The Wire” (BITW) Architecture

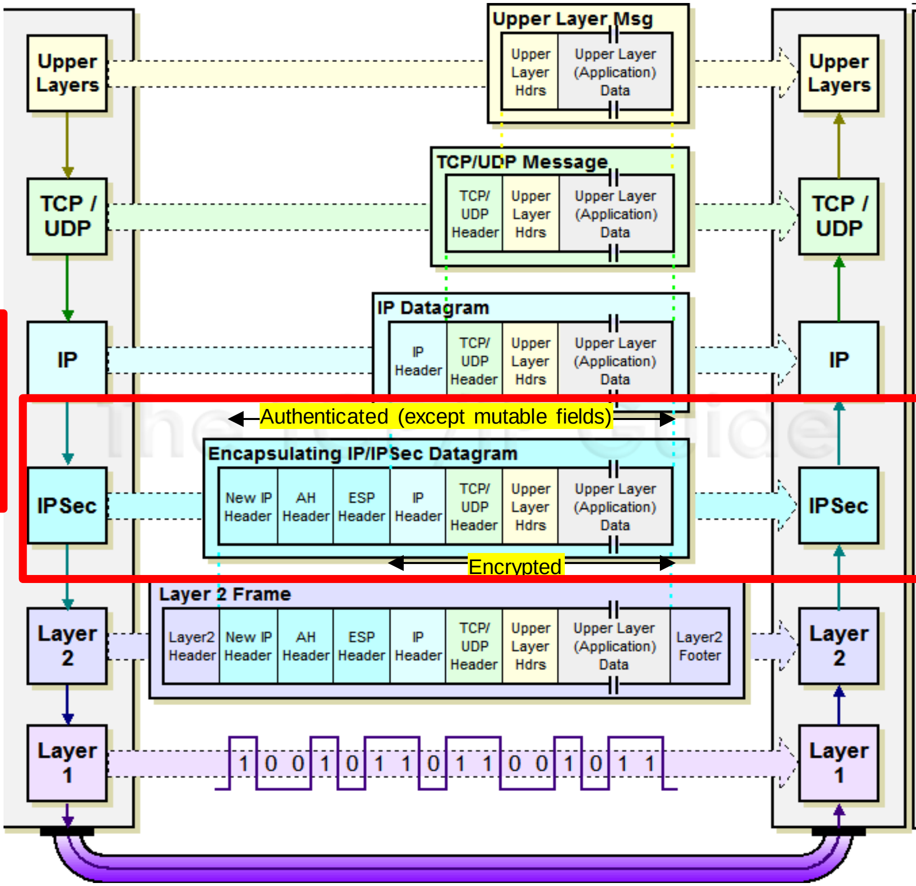


IPSec Modes (Tunnel & Transport)



Transport Mode

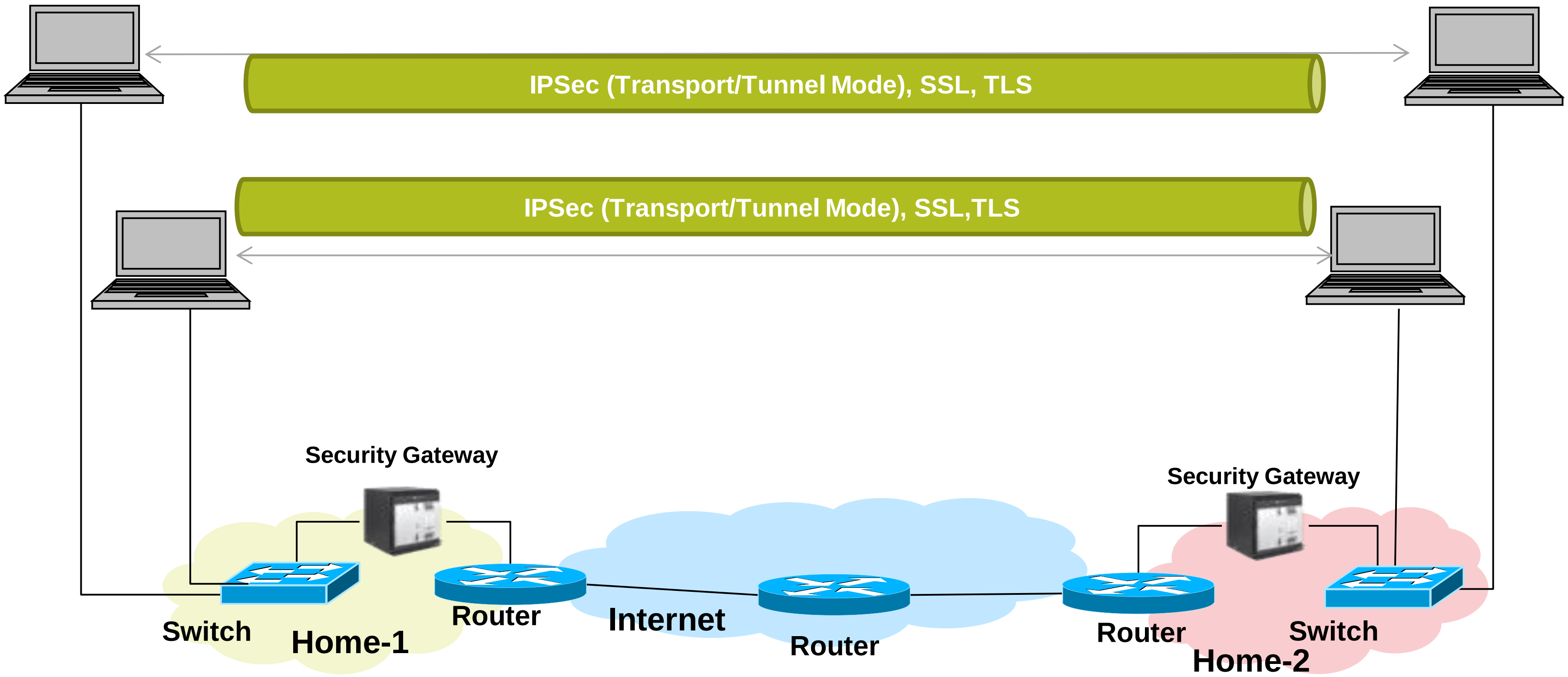
Used mostly in integrated architecture



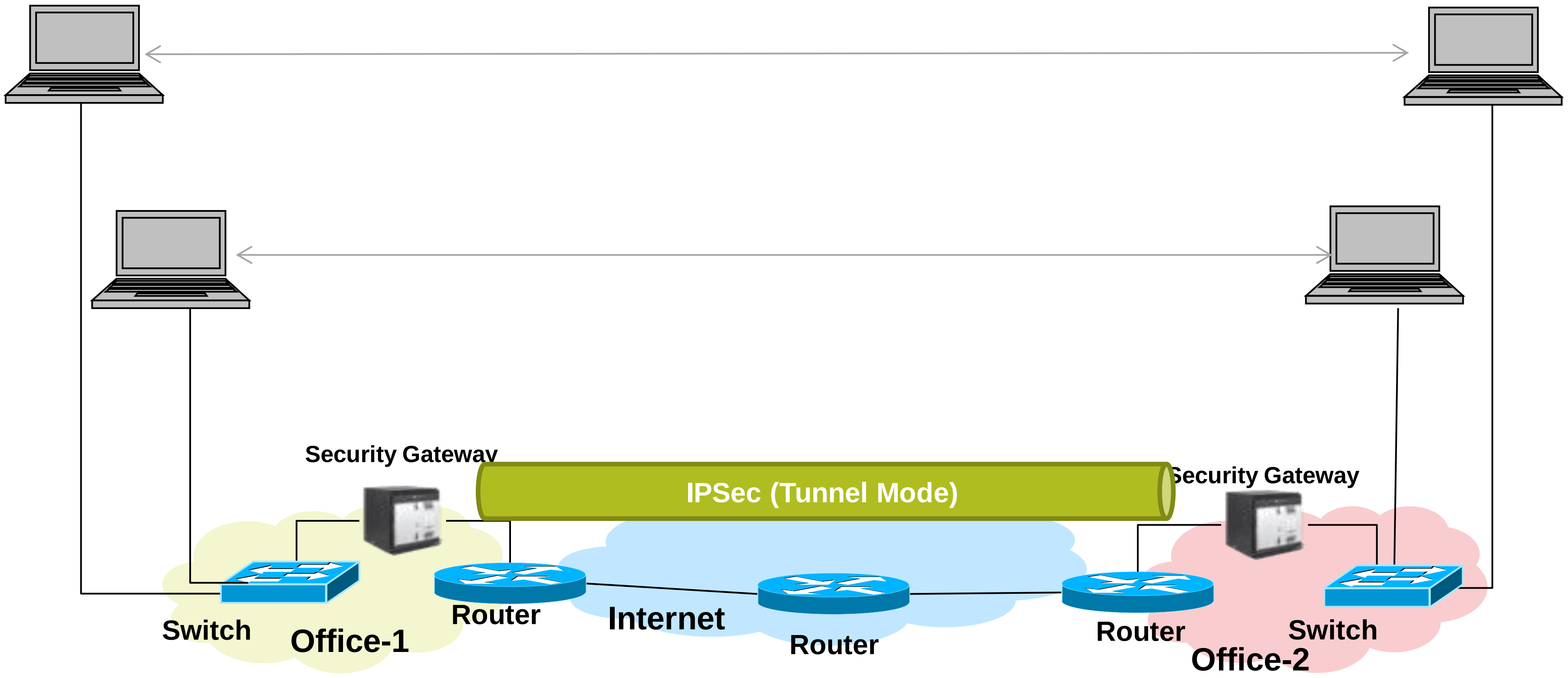
Tunnel Mode

Used mostly in the bump in the wire architecture (also bump in the stack). The outer IP header is between the gateways (SEGs)

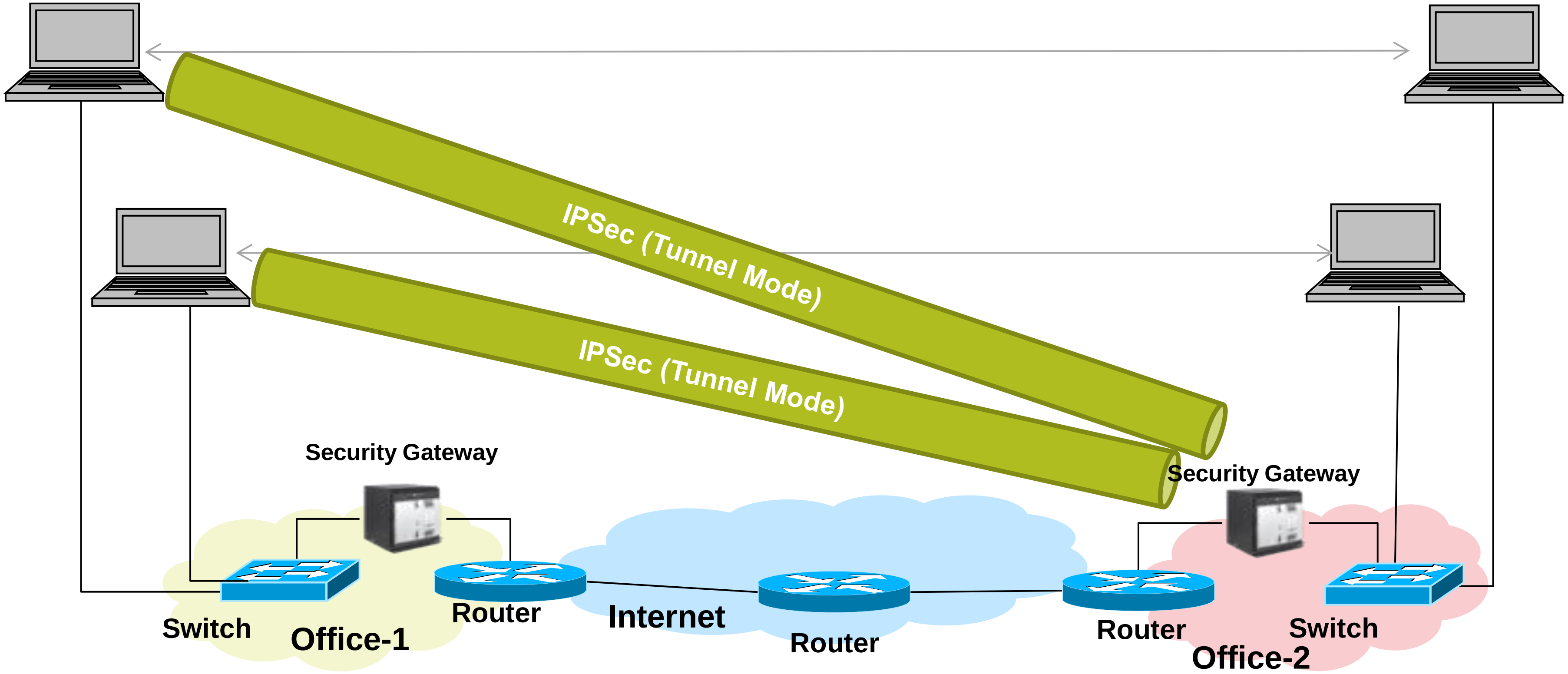
IPSec Deployment Scenarios: Host to Host



IPSec Deployment Scenarios: Gateway to Gateway

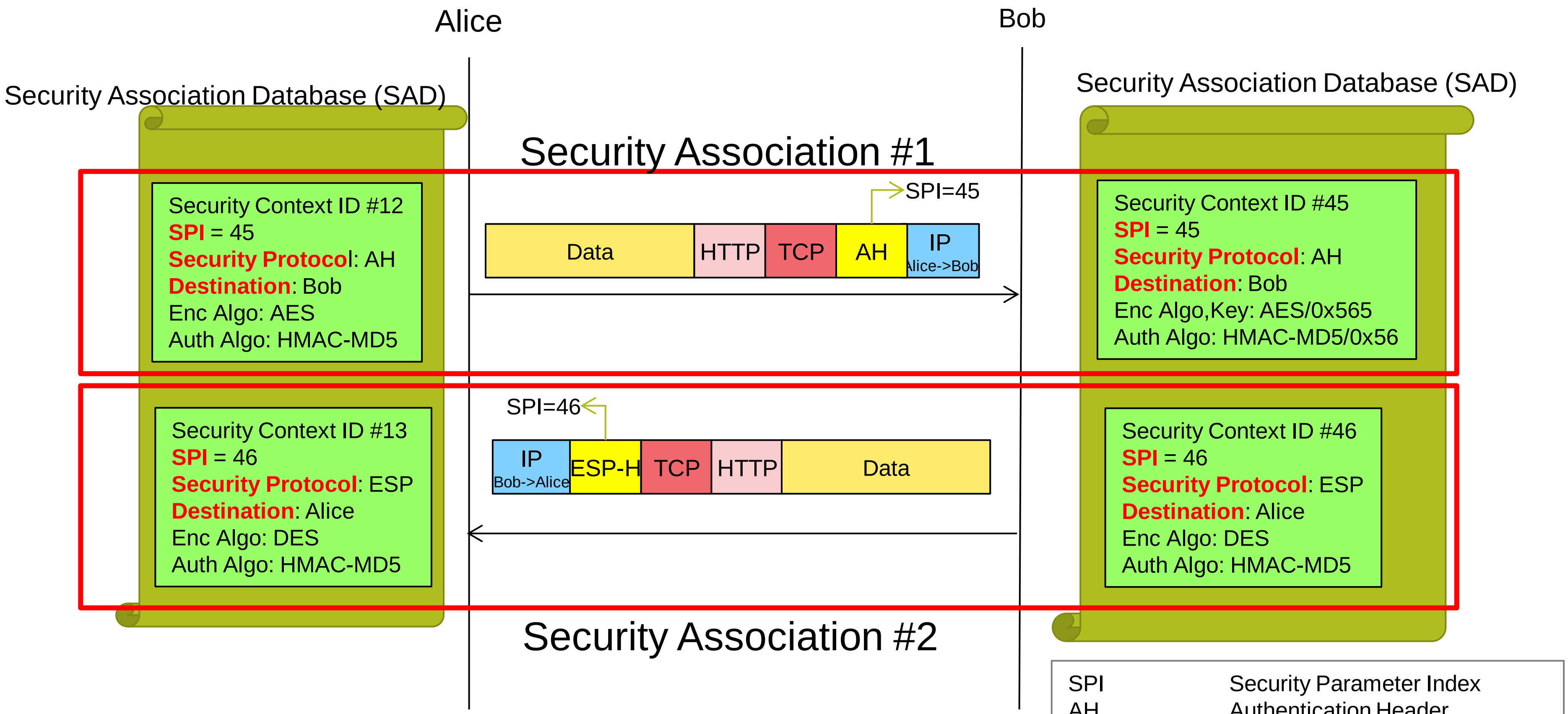


IPSec Deployment Scenarios: Host to Gateway



- Similar to VPN Tunnels used today for security.

Security Association



Alice

Bob

Security Association Database (SAD)

Security Association #1

Security Context ID #12
SPI = 45
Security Protocol: AH
Destination: Bob
Enc Algo: AES
Auth Algo: HMAC-MD5

Security Context ID #45
SPI = 45
Security Protocol: AH
Destination: Bob
Enc Algo,Key: AES/0x565
Auth Algo: HMAC-MD5/0x56

Security Context ID #13
SPI = 46
Security Protocol: ESP
Destination: Alice
Enc Algo: DES
Auth Algo: HMAC-MD5

Security Context ID #46
SPI = 46
Security Protocol: ESP
Destination: Alice
Enc Algo: DES
Auth Algo: HMAC-MD5

Security Association #2

SPI	Security Parameter Index
AH	Authentication Header
ESP-H	Encapsulating Security Protocol

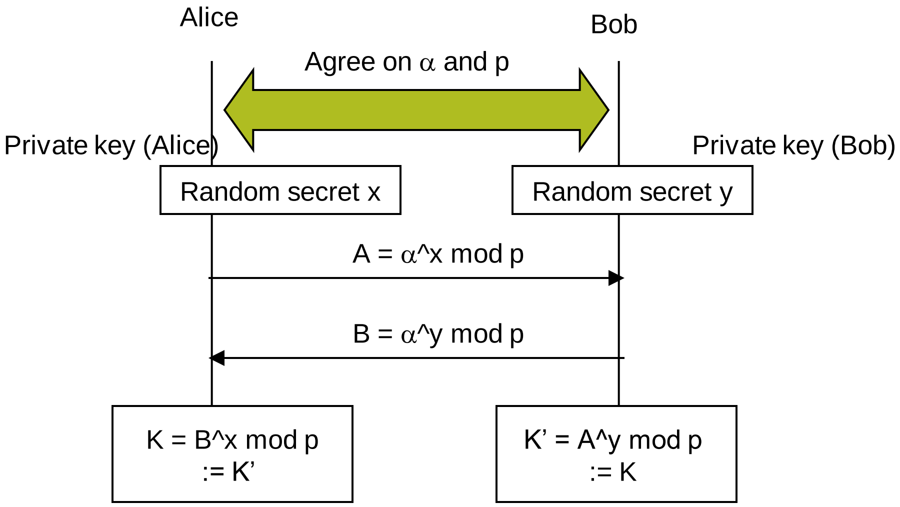
- Security Association is the security context that contains details (eg. Keys, algorithms, destination address,..) used for one direction of the communication
- There are two Security Associations for a bi-directional communication.

Setting up the security association

- Security association:
  - Keys
  - Algorithms and their parameters
- An SA is uniquely identified by a 3-tuple composed of:
  - Security Parameter Index (SPI), a 32-bit identifier of the connection
  - IP Destination Address
  - security protocol (AH or ESP) identifier
- Internet Security Association and Key Management Protocol (ISAKMP) defines framework that allows nodes to agree on a SA in a secure way.
- Internet Key Exchange is a hybrid of ISAKMP, Oakey and SKEME.
  - Don’t ask me what the last two are.

IPSec Key Exchange (IKE)

- IPSec, like many secure networking protocol sets, is based on the concept of a “shared secret”. Two devices that want to send information securely encode and decode it using a piece of information that only they know. Anyone who isn't “in” on the secret is able to intercept the information but is prevented either from reading it (if ESP is used to encrypt the payload) or from tampering with it undetected (if AH is used). Before either AH or ESP can be used, however, it is necessary for the two devices to exchange the “secret” that the security protocols themselves will use. The primary support protocol to exchange keys in IPSec is called Internet Key Exchange (IKE).
- IKE (builds on) uses Diffie-Hellman key-exchange algorithm in which two parties generate secure keys by exchanging messages over insecure network.



Alice

Bob

Agree on α and p

Private key (Alice)

Private key (Bob)

Random secret x

Random secret y

$A = \alpha$^x mod p

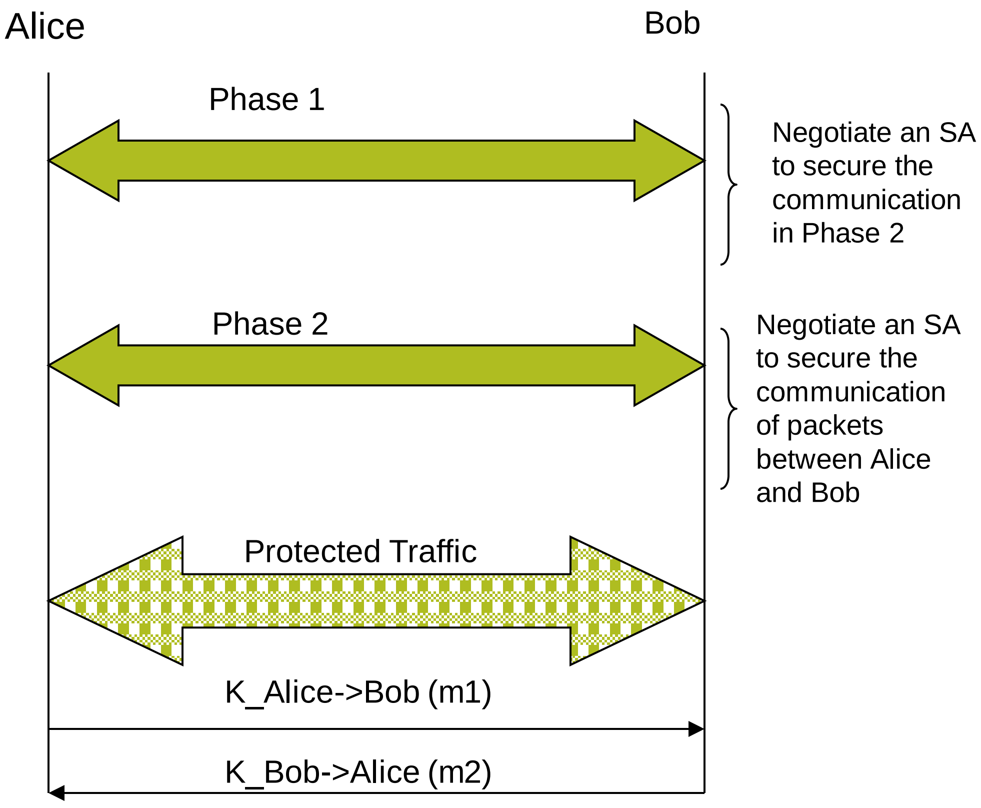
$B = \alpha$^y mod p

K = B^x mod p
:= K’

K’ = A^y mod p
:= K

α, p, A and B can be transmitted in the open, but difficult for attacker to calculate x and y
A,B are the public keys and x, y are the private keys
IKE defines six groups of (a, p). Only the index is exchanged.

IPSec Key Exchange Negotiation for setting up Security Association


Alice
Bob
Phase 1
Negotiate an SA to secure the communication in Phase 2
Phase 2
Negotiate an SA to secure the communication of packets between Alice and Bob
Protected Traffic
K_Alice->Bob (m1)
K_Bob->Alice (m2)

- Phase 1:
  - Public keys and certification, or pre-configured secret keys and mutual authentication.
  - Diffie-Hellman used in this tage.
- Phase 2 uses the secure channel created in Phase 1 to negotiate the SA:
  - Protocol used to protect comm (AH or ESP)
  - Type of data (TCP or UDP)
  - Lifetime of SA
  - Material used to generate keys.
- Following IKE exchange, the two nodes have enough material to generate two different keys. Each key is used to protect information in one direction.
  - Symmetric cryptography is used in unidirectional SAs.

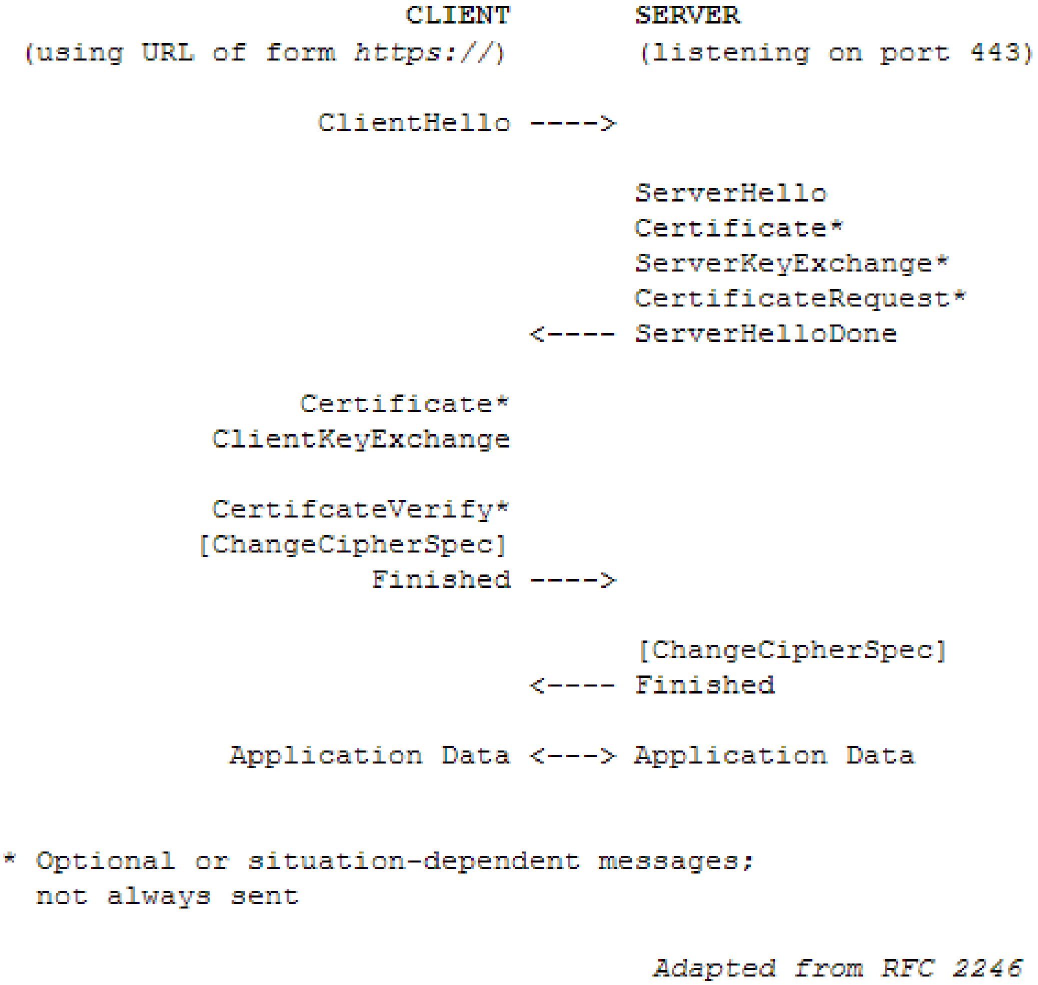
SSL and TLS

- The Secure Sockets Layer (SSL) protocol was developed by Netscape Communications to provide application-independent secure communication over the Internet for protocols such as the Hypertext Transfer Protocol (HTTP)
  - SSL employs RSA and X.509 certificates during an initial handshake used to authenticate the server (client authentication is optional)
  - In 1997, SSL v3 was found to be breakable
- TLS extends SSL and supports additional crypto schemes, such as Diffie-Hellman key exchange and DSS digital signatures; RFC 4279 describes the pre-shared key crypto schemes supported by TLS.
- SSL/TLS also provides security to the following protocols (using TCP):

Protocol	TCP Port Name/Number
Hyper Text Transfer Protocol (HTTP)	https/443
File Transfer Protocol (FTP)	ftps-data/989 & ftps/990
Internet Message Access Protocol v4 (IMAP4)	imaps/993
Lightweight Directory Access Protocol (LDAP)	ldaps/636
Network News Transport Protocol (NNTP)	nntps/563
Post Office Protocol v3 (POP3)	pop3s/995
Telnet	telnets/992

- SSL/TLS has also been applied for UDP with protocol called Datagram Transport Layer Security (DTLS)

SSL/TLS Negotiation for HTTPS



```
                       CLIENT       SERVER
 (using URL of form https://)       (listening on port 443)

                  ClientHello ---->

                                    ServerHello
                                    Certificate*
                                    ServerKeyExchange*
                                    CertificateRequest*
                              <---- ServerHelloDone

                 Certificate*
            ClientKeyExchange

            CertifcateVerify*
           [ChangeCipherSpec]
                     Finished ---->

                                    [ChangeCipherSpec]
                              <---- Finished

             Application Data <---> Application Data


* Optional or situation-dependent messages;
  not always sent

                                     Adapted from RFC 2246
```

Virtual Private Network

- VPN can use of the following protocols
  - IPSec Tunnel Mode
  - Transport Layer Security (SSL/TLS): OpenVPN
  - Datagram Transport Layer Security (DTLS): Cisco AnyConnect VPN
  - Secure Shell (SSH) VPN