DI-FCT-UNL
Segurança de Redes e Sistemas de Computadores
Network and Computer Systems Security

Mestrado Integrado em Engenharia Informática
MSc Course: Informatics Engineering
2º Semestre, 2020/2021

IPSec (IP Security)

"If a secret piece of news is divulged by a spy before the time is ripe, he must be put to death, together with the man to whom the secret was told."

—*The Art of War*, Sun Tzu

Before … We analysed TLS

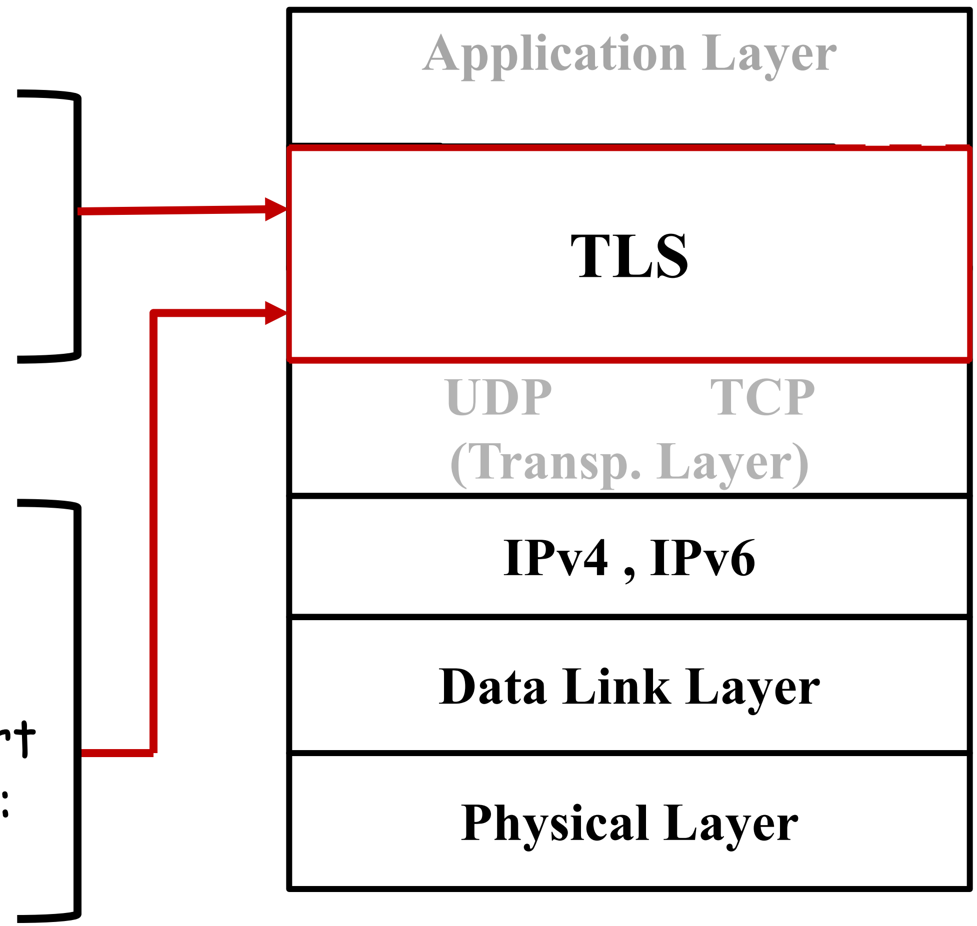
TLS Stack: set of protocols enabling security on top of Transport Layer (in TCP/IP Stack) providing:

Session-Layer-Security Services :

- HP: Handshake Protocol
- CCSP: Change Cipher Spec
- AP: Alert Protocol
- HBP: Heartbeat Protocol

Transport-Layer Security Services:

- RLP: Record Layer Protocol:
    - Transport-Level Security Format on top of Transport protocols in TCP/IP Stack: TCP, UDP



Today: IPSec - IP Security
Goal: Network-Layer Security (IP Level Security)

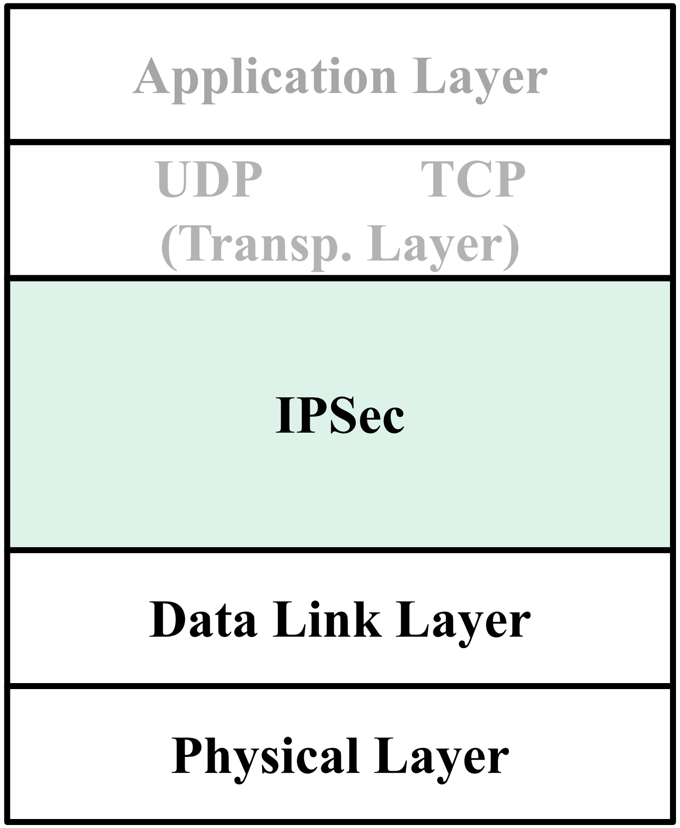
Security at the Network Level (IP Traffic Protection)

- Initially addressed as answer to requirements and challenges in IAB (RFC 1636, Fev/1994)
- IPSec Architecture, AH and ESP Protocols (1st Approach):
  IETF RFCs 1825, 1826, 1827 (Aug/1995)
  ……. > RFCs 7296, 7670, 8247 (… 2017)
  ……. > other RFC work-drafts (on going)

IPSec Stack Standardization has been an evolving effort

Extensive standardization & documentation

See, ex: https://en.wikipedia.org/wiki/IPsec



Learning topics (study check list)

- Know about what is IPSec / Stack of IPSec protocols and their roles
- Know about the security properties provided by IPSec
- Know about IPSec modes: transport and tunnel mode
- Know about the IPSec general operation and know about IPSec Security Associations and Security Policies, their differences and how they are managed and maintained
- Know how IPSec packets (IKE/ISAKMP, ESP-A, ESP-AE or AH) are processed as outbound/inbound packets
- Know how protocols are used and how they are encapsulated in IPV4 or IPV6 packets (IPSec/IP overlaying)
- Know to interpret IPSec encapsulation (ex., looking to a wireshark trace)
- Know the security guarantees specifically provided by ESP (ESP-A, ESP-AE) and AH
- Know about the handshake supported by IKE
- Know about other flexible forms of using IPSec: encapsulation variants in TCP/IP stack and combination of Security Associations
- Know the cryptographic mechanisms used by IPSec protocols

Roadmap / Outline

- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

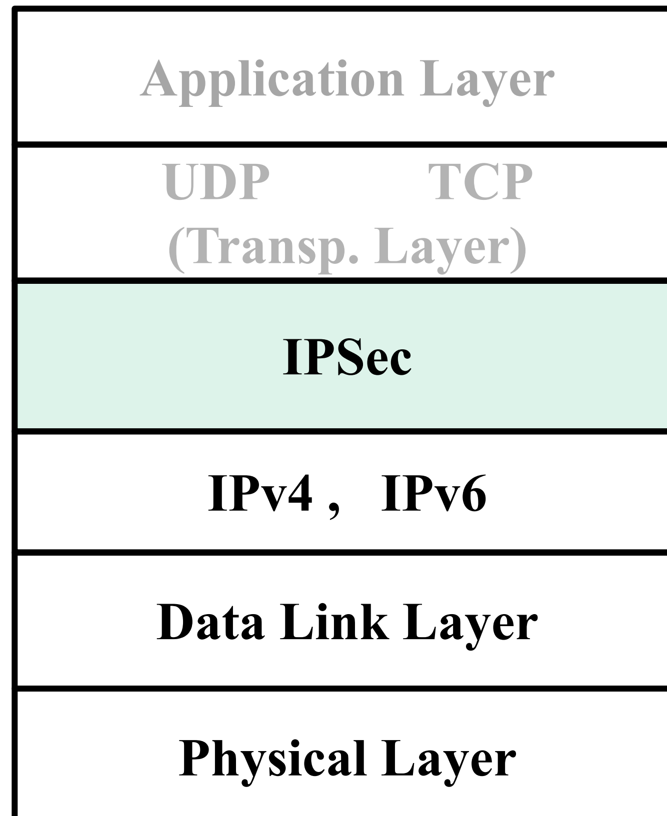
What is IPSec ?

Security Approach at the Network Level (IP Traffic Protection)

- In the base idea: IP/IP (IPSec/IP) encapsulation approach

IPSec: A Security Stack of sub-protocols, used for IP Traffic Protection

Supported (encapsulated) by IPV4 and IPV6 Protocols



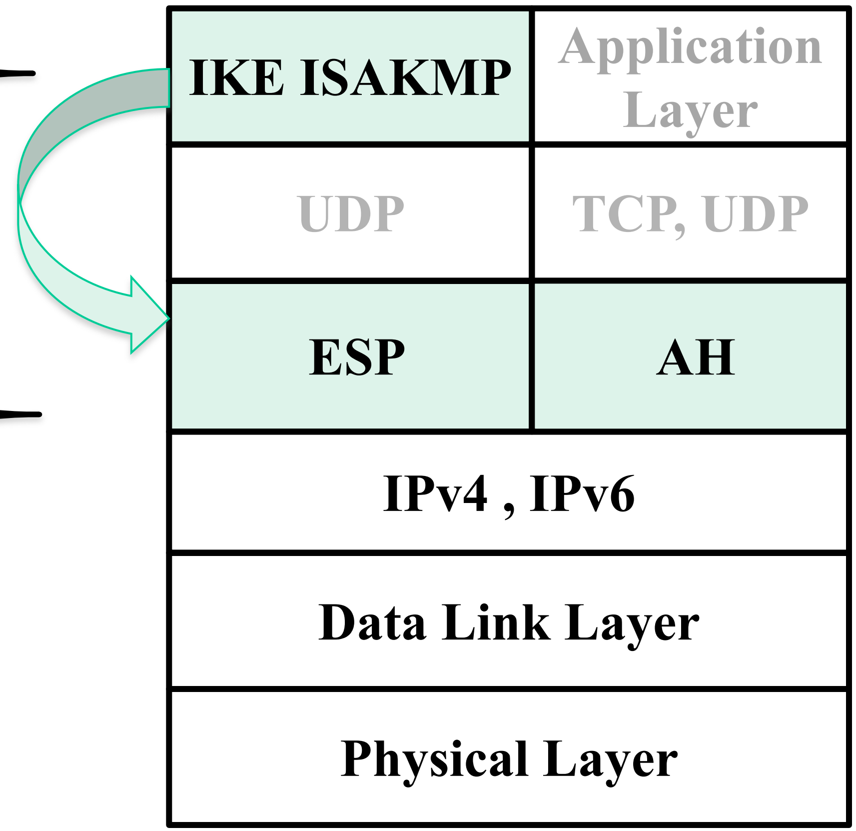
IPSec Protocols Stack: IKE, ISAKMP, ESP and AH Protocols

IKE
Internet Key Exchange

ISAKMP
Internet Security Association and Key Management Protocol (Message format for IKE)

ESP
Encapsulating Security Payload

AH
Authentication Header



IP Security Sub-Protocols: Security Guarantees

IPsec Stack: IKE, ISAKMP, ESP and AH Protocols

IKE
Internet Key Exchange

ISAKMP
Internet Security Association and Key Management Protocol (Message format for IKE)

- Peer-authentication of IPSec Endpoints (IP Addresses)
- Secure Establishment of Keys and other SA (Security Assotiation) Parameters between IPSEc endpoints
- Access-Control (or Packet Admission Control) Mechanism

ESP: ESP-A, ESP-AE
Encapsulating Security Payload
ESP-A (ESP w/ Authentication Only)
ESP-AE (ESP w Authentication and Encryption)

- Payload Data Origin Authentication
- Connectionless-Integrity
- Anti-Replaying
- Connectionless-Confidentiality
- Limited Traffic Flow Confidentiality

AH
Authentication Header

- Payload Data Origin Authentication
- Connectionless-Integrity
- Anti-Replaying

IP Security Stack: Base security mechanisms

IPsec Stack: IKE, ISAKMP, ESP and AH Protocols

IKE
Internet Key Exchange

- X509 Certification +
- Authenticated Diffie-Hellman
- Agreement: EH,ECDH, ECDSA
  Digital Signatures, HMACs-SHA2
  and other techniques
- Access-Control (or Packet Admission Control) List

ISAKMP
Internet Security Association and Key Management Protocol (Message format for IKE)

ESP: ESP-A, ESP-AE
Encapsulating Security Payload

- ESP-Authentication Only:
  - use of HMACs-SHA2)
- ESP-Authentication and Encryption:
  Use of HMACs + Symmetric Encryption (in different encryption modes, ex: GCM, GMAC)

AH
Authentication Header

- Authentication Header
- (Use of HMACs-SHA2)

Summary of IPSec Services (Ref ESP, RFC 4301)

- Access control for IPsec packets
- Connectionless integrity
- Data origin authentication (IP Authentication) of delivered/received IP packets (*)
- Anti-Replaying Protection: Rejection of replayed packets
  - a form of partial sequence integrity
- Confidentiality: Connectionless Confidentiality (encryption)
- Limited traffic flow confidentiality protection, w/ possible enforcement using tunnelling encapsulation strategies

---

- Helps in securing routing, but no routing control: different routing attacks require other contra-measures complementarily to IPSec
  - Problem/Focus: Security in Routing Protocols (Ex., Secure BGP)

Protection in the IPSec protocol suite

Protection against communication attacks against IP Traffic (remember ref. X.800 or RFC 2828)

	AH	ESP (E-Only)	ESP (A+E)
Access control – IPSec Packet admission	X	X	X
Connectionless integrity	X		X
Authentication (IP origin) (authentication of the IP peers and packet origin)	X		X
Anti-*replay* (IP packet replay) *(Form of Sequential integrity)*	X	X	X
Connectionless Confidentiality + limited traffic-flow confidentiality		X X	X X
• Availability (DoS, DDoS)	?	?	?
• Routing control (IP routing control)	?	?	?

Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

Use of IPSec

- Secure branch office connectivity over the Internet
  - Branch-to-Branch, ex., LAN-to-Lan
- Secure remote access over the Internet
  - Ex., Virtual Private Networks - VPN Access
- Establishing secure extranet and intranet connectivity with partners
  - Secure internetworking between private intranets
- Enhancing security in supporting internetworking infrastructures for different applications
  - Electronic commerce infrastructures
  - Critical infrastructures and related secure systems and applications

Benefits and Support of IPSec

Protection below transport layer (network level):

- Secure IP Traffic between IP Sec endpoints
- Transparency: provides security to transport or application/transport protocols

IPSec is supported:

Via routers or network firewalls

- Provides strong security to all traffic crossing the perimeter (perimeter protection strategy)
- Resistant to bypass
- NAT supported

Tunnel Mode

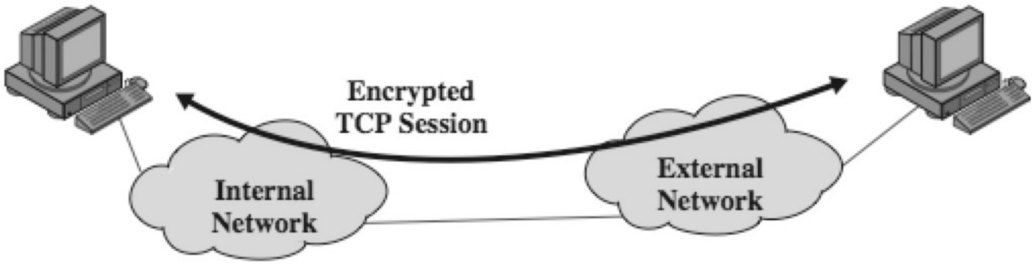
Between end-hosts (Host-Host)

- Provides end-to-end (host-to-host) traffic
- Resistant to bypass w/ local-enabled firewalls

Transport Mode

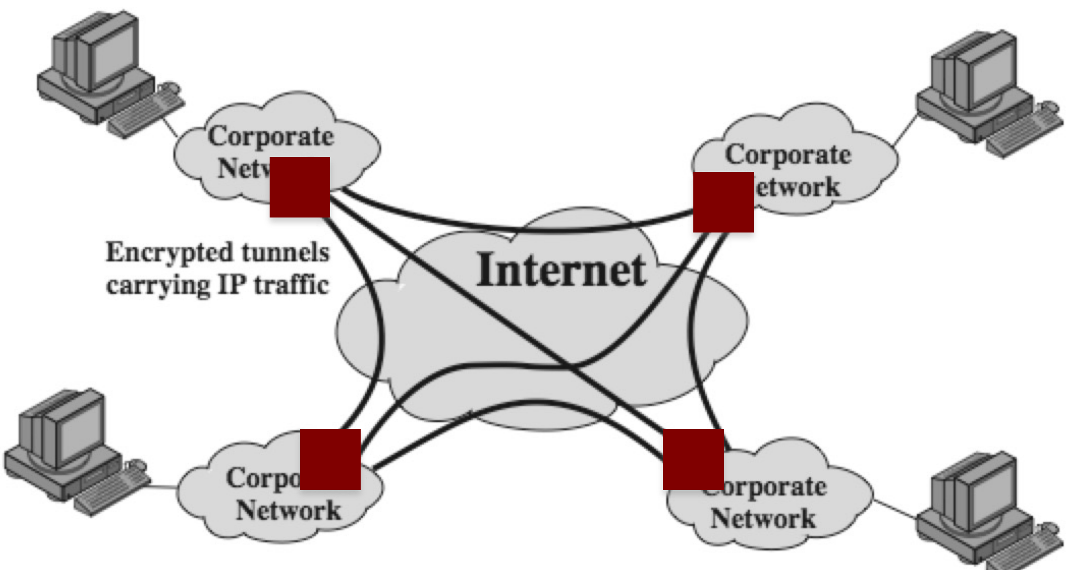
Transport and Tunnel Modes

- **Transport Mode:**
  - End-to-End Security
  - Host-to-Host



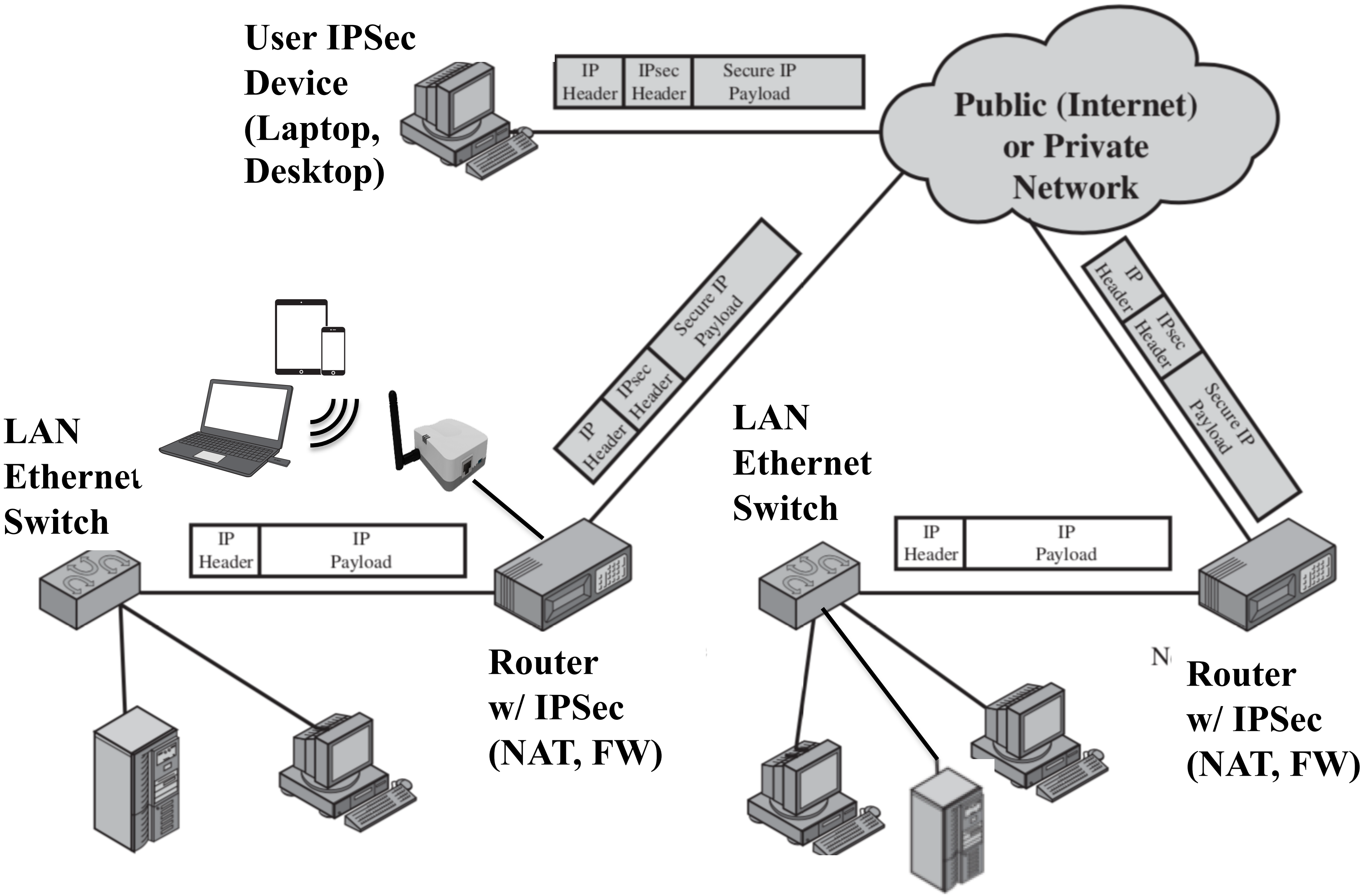
(a) Transport-level security

- **Tunnel Mode:**
  - Intermediary-Support
  - via Routers, Firewalls, VPN Servers or Gateways
  - NAT supported

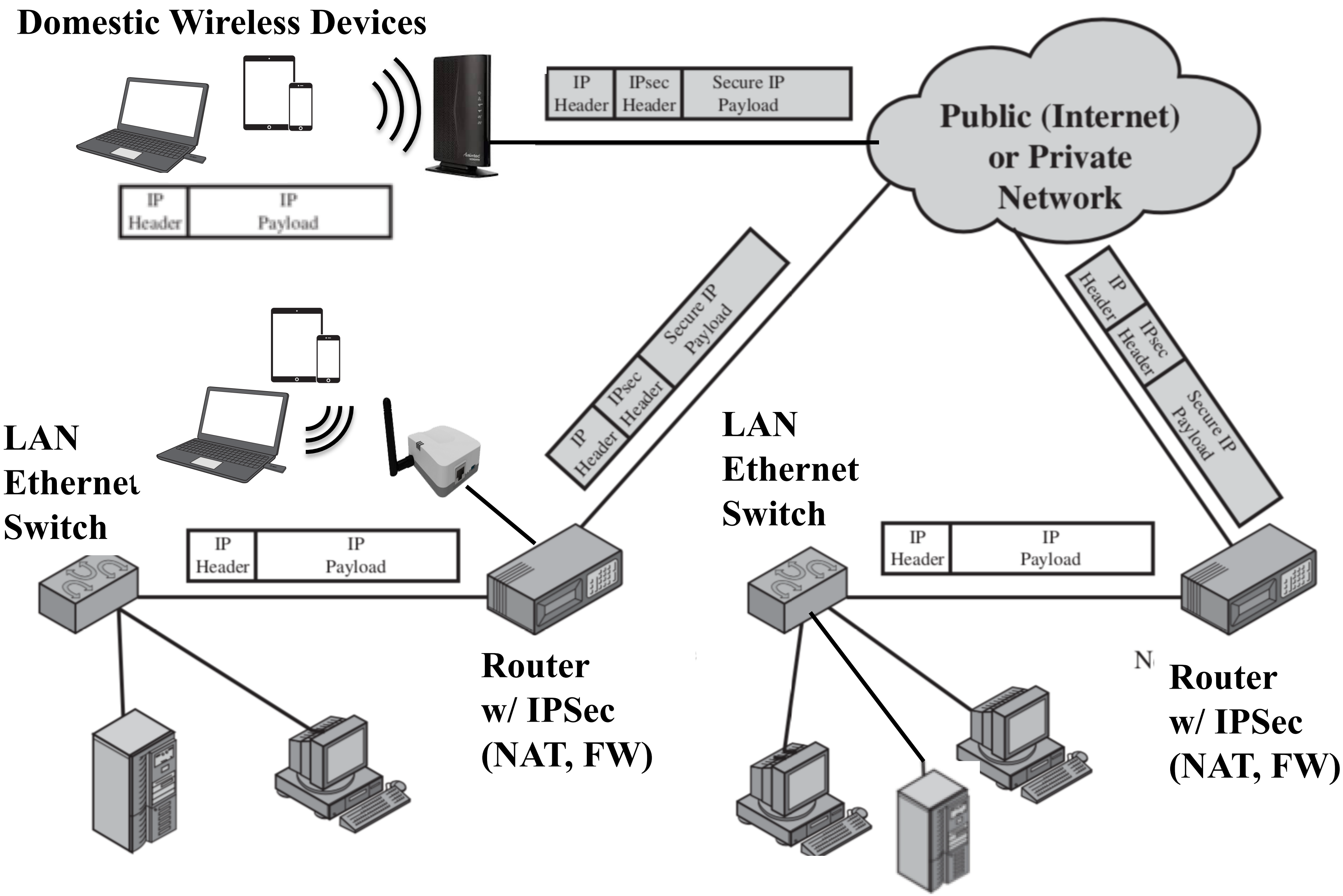


(b) A virtual private network via Tunnel Mode

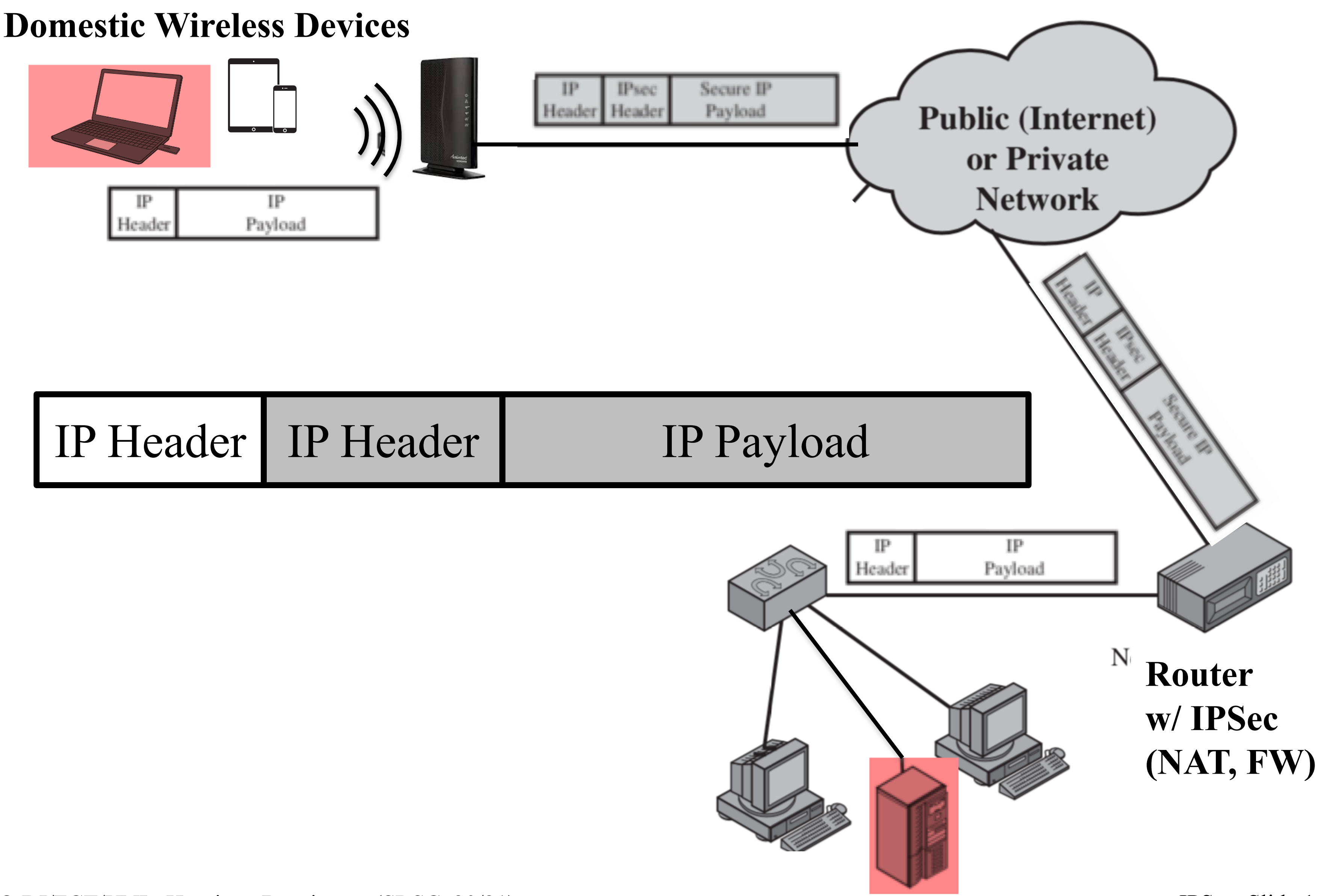
IPSec Internetworking scenario



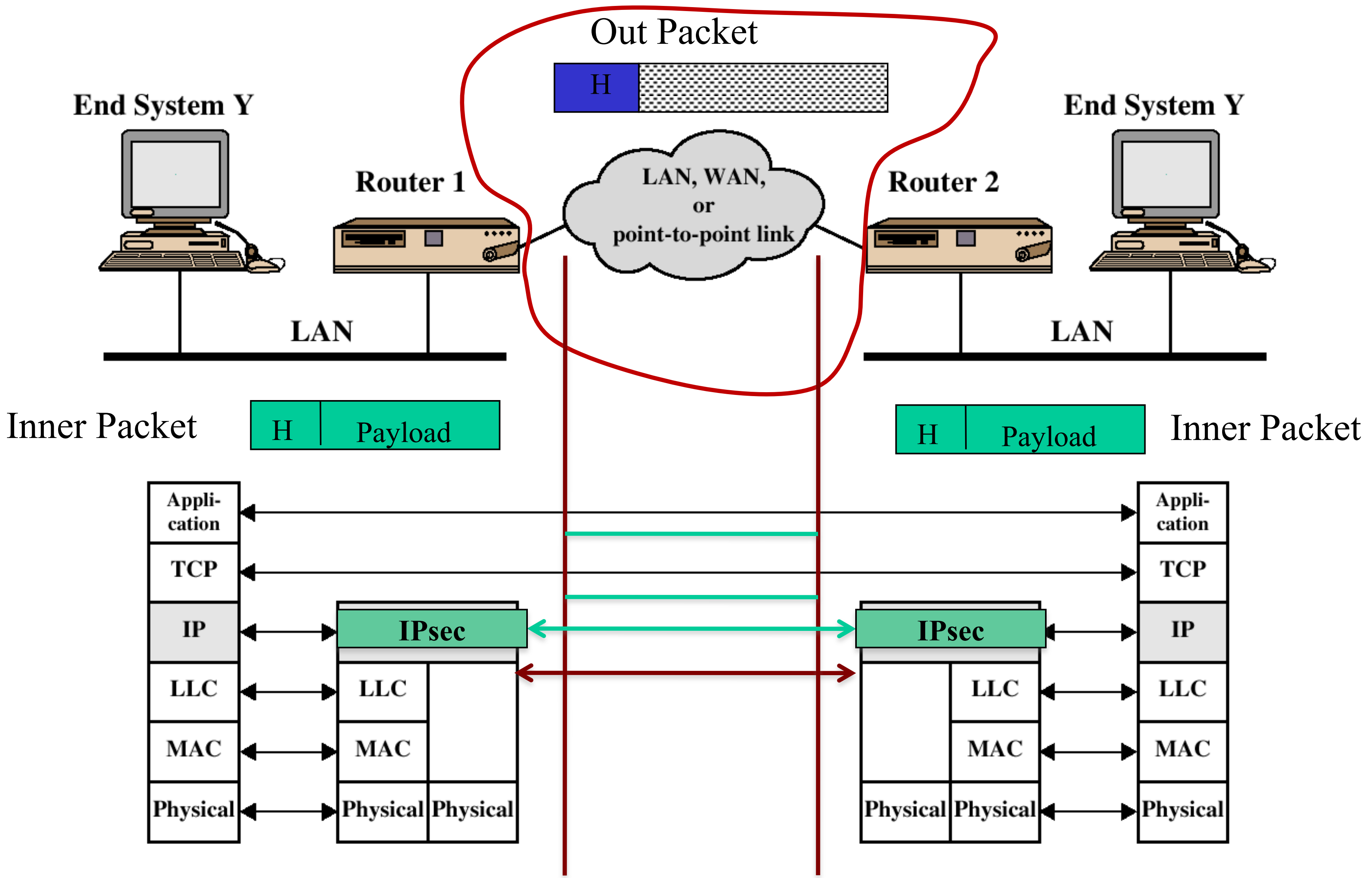
IPSec Internetworking scenario



IPSec Internetworking scenario



Secure LAN to LAN interoperability



Other benefits of IPSec

Helps in securing routing architecture (and other "control plane" management protocols, ex: ARP, RARP, ICMP ...)

- Could be protection of router advertisements: authentication/authorization of advertisements, control of authenticated/authorized neighbors, authentication of redirections, contra-measures against forged update announcements
- What about protection for routing protocols (OSPF RIP, BGP) or DNS traffic protection ? Other alternatives: BGPSec, DNSSec

Some issues in playing well together:

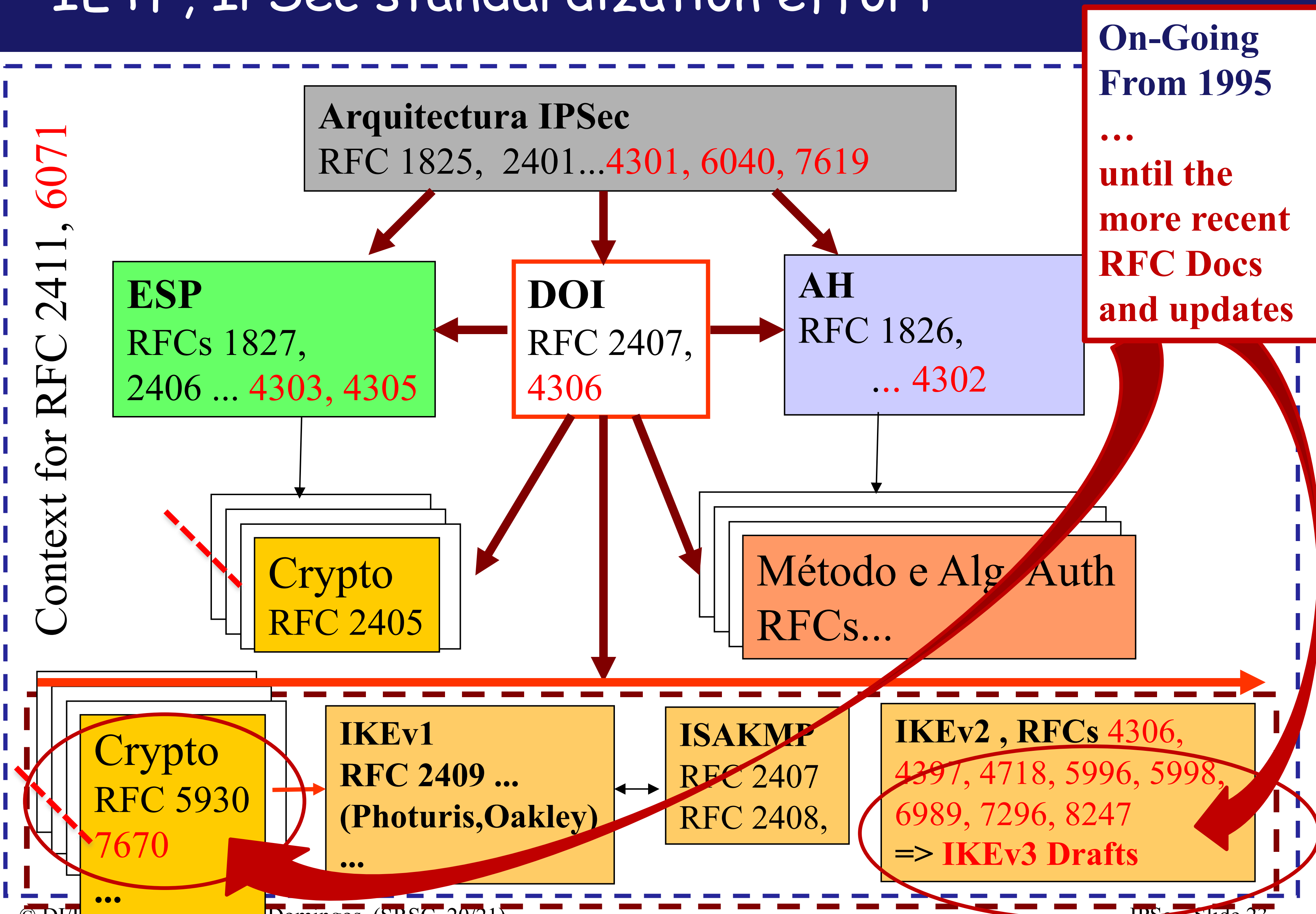
- Performance penalties due to IP Sec rekeying (IKE sub protocol)
- Outages due to "missed or desync. keys and security associations" or lack of global IPSec coverage
- DoS / DDoS Issues due to overheads imposed by IPSec processing
- Ex., general BGP routers have layered DoS protection that encapsulated IPSec BGP packets may weaken
  - Mitigation requires that routers must have access to the BGP packets
  - Alternatives: Secure BGP without IPSec (S-BGP, BGPSec, RFC8205)

Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IETF, IPSec standardization effort



IPSec standardization (currently v3)

IPSec is a Security Suite with different dimensions involved in the on-going standardization effort:

- **Conceptual bases**
  - IPSec Domain of Interpretation
  - IPSec Architecture Reference
- **IKEv2 and ISAKMP (currently IKEv3 working drafts)**
- **Sub-protocols (IPSec protocol stack)**
  - ESP (ESP AE, ESP A only), AH
- **Configuration and Management Protocols**
  - IPSec Security Association Parameters and Security Policies
- **IPSec Standardized Cryptography and Techniques**
- **Adaptation and integration issues (TCP/IP stack)**
  - IPV4 and IPV6 Support and Encapsulation
  - Adoption of other forms of IPSec encapsulation

IPSec suite: Architecture, AH and ESP

- **Architecture:** Covers the general concepts, security requirements, definitions, and mechanisms defining IPsec technology.
- **Authentication Header (AH Protocol):**
  - AH is an extension header to provide message authentication.
  - Because message authentication is also provided by ESP, the use of AH is now deprecated.
  - It is included in IPsecv3 for backward compatibility but should not be used in new applications. We do not discuss AH in this chapter.
- **Encapsulating Security Payload (ESP Protocol):**
  - ESP consists of an encapsulating header and trailer
  - Used to provide encryption (ESP-E) or combined encryption/authentication (ESP-AE)

IPSec suite: IKE, Crypto and SA/SP Management

- **Internet Key Exchange (IKE):** This is a collection of documents describing the key management schemes for use with IPsec. The initial specification is RFC 4306, *Internet Key Exchange (IKEv2) Protocol*, but there are a number of related / evolved RFCs.
  - Evolution effort for **IKEv3**

- **Cryptographic algorithms:** This category encompasses a large set of docu- ments that define and describe cryptographic algorithms for encryption, mes- sage authentication, pseudorandom functions (PRFs), and cryptographic key exchange.

- **Others:**
  - There are a variety of other IPsec-related RFCs, including those dealing with security policy and management information base (MIB) content.
  - Other IETF RFCs on different IPSec encapsulations in TCP/IP Stacks

IETF, IETF WorkGroups and OnGoing Work

- IETF
  - https://www.ietf.org
- IETF WG Charter .. See Active WGs
  https://datatracker.ietf.org/wg/
- IPSec, https://datatracker.ietf.org/wg/ipsec/about/
- IPSec Maintenance and Extensions (**ipsecme**)https://datatracker.ietf.org/wg/ipsecme/about/

Last and Ongoing Efforts (1984 …. 2009-2017 … 2019, 2020 …)

Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IP Security Stack (Archit. amd Sub-Protocols)

IPSec Architecture and vast related standardization effort

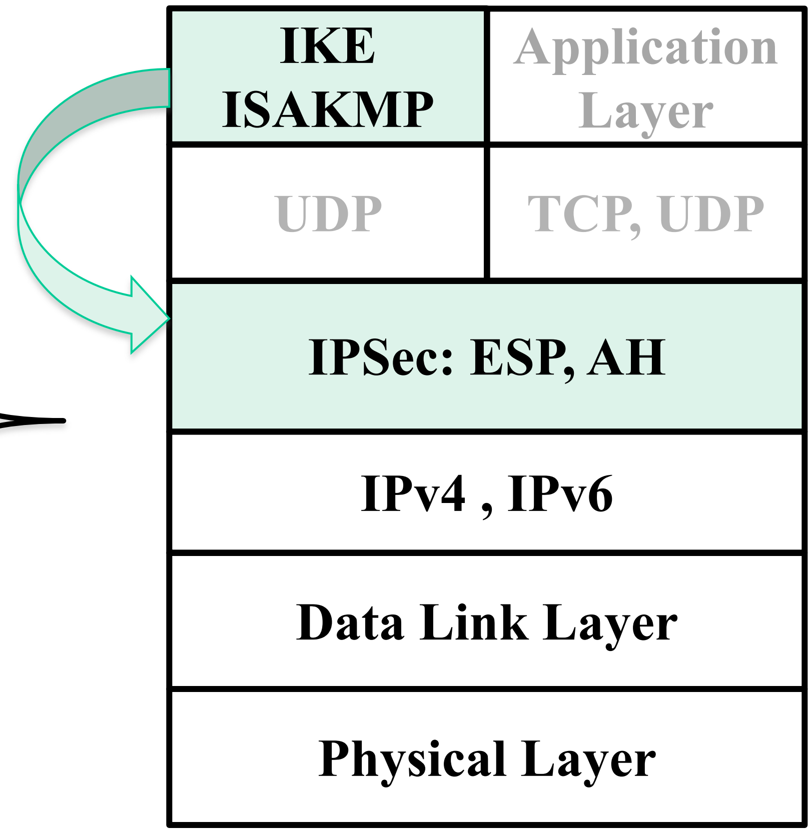
Ipsec Protocols Stack: IKE, ISAKMP, ESP and AH Protocols

IKE
Internet Key Exchange

ISAKMP
Internet Security Association and Key Management Protocol

ESP
Encapsulating Security Payload

AH
Authentication Header



IPSec Encapsulation (IPv4)

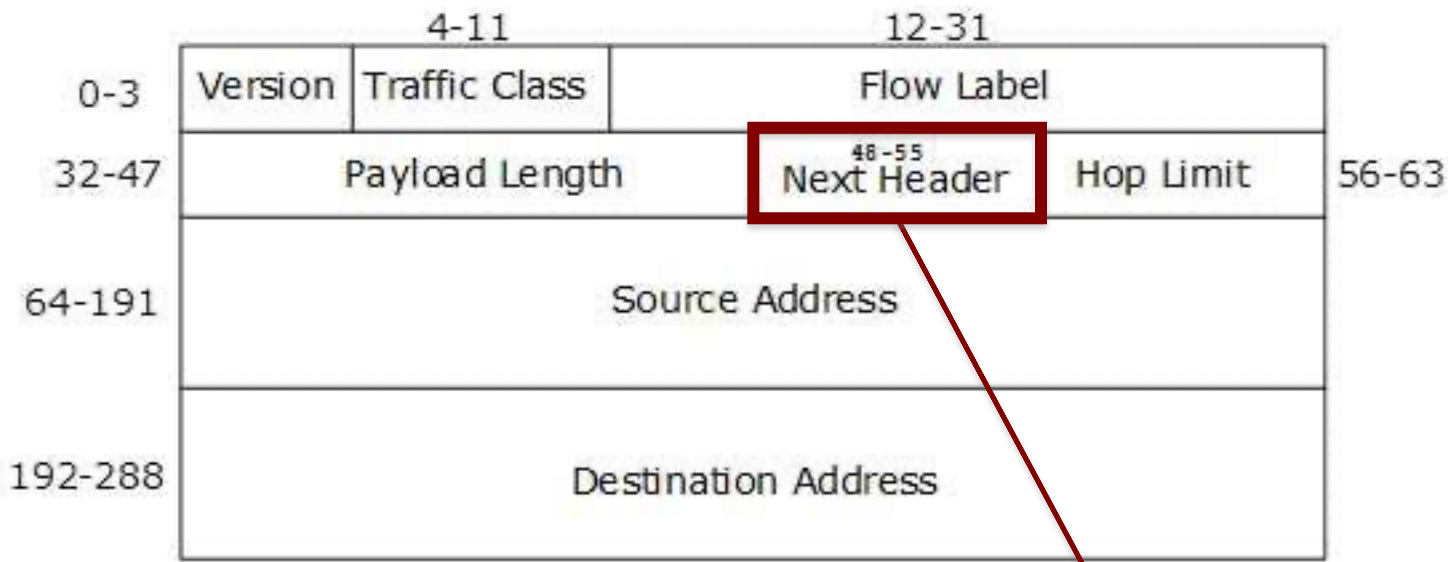
<table>
<tr><th>4</th><th>4</th><th>8</th><th colspan="2">16</th></tr>
<tr><td>Version</td><td>IHL</td><td>Type of Service</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source Address</td></tr>
<tr><td colspan="5">Destination Address</td></tr>
<tr><td colspan="5">Options (+ Padding)</td></tr>
<tr><td colspan="5">Data (Variable)</td></tr>
</table>

Protocol Codes, ex:
1 (ICMP Control Packet)
6 (TCP packet)
41 (IPV6 encapsulation), 4 (IPV4 encapsulation)
17 (UDP datagram packet …
51 (AH Encapsulated Packet)
50 (ESP Encapsulated Packet)

https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml

IPSec encapsulation (IPv6)

see https://en.wikipedia.org/wiki/IPv6_packet for details



Next Header (encapsulated packet), ex:
1 (ICMP Control Packet)
6 (TCP packet)
41 (IPV6 encapsulation), 4 (IPV4 encapsulation)
17 (UDP datagram packet …
51 (AH Encapsulated Packet)
50 (ESP Encapsulated Packet)

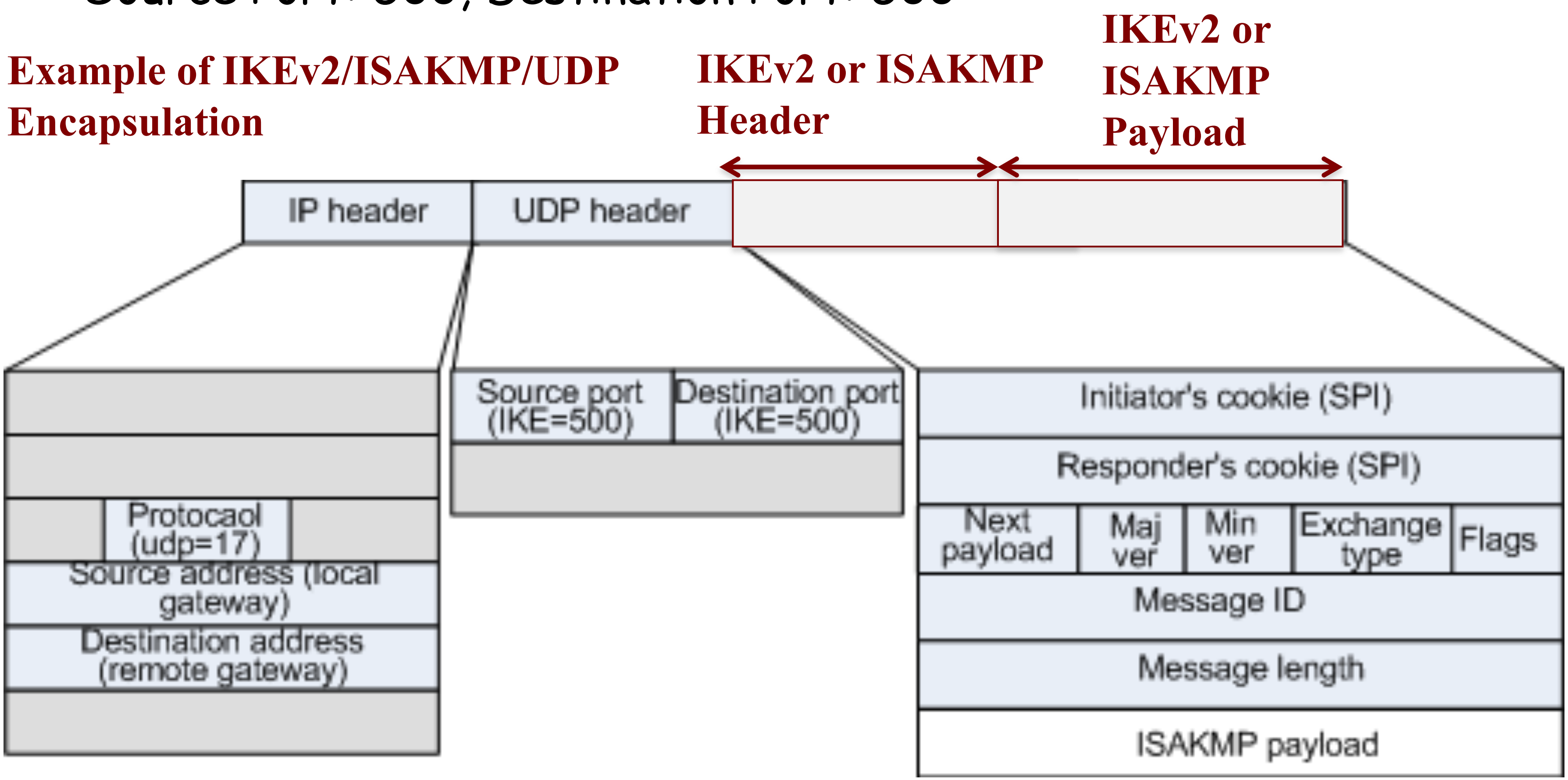
IKE / ISAKMP encapsulation: UDP, TCP, HTTP, Encapsulations

IKE is usually encapsulated on UDP Packets

(On-going/recent RFC proposals on TCP and also HTTP encapsulation)

- Via IKEv2 or ISAKMP Headers
- Source Port: 500, Destination Port: 500

Example of IKEv2/ISAKMP/UDP Encapsulation



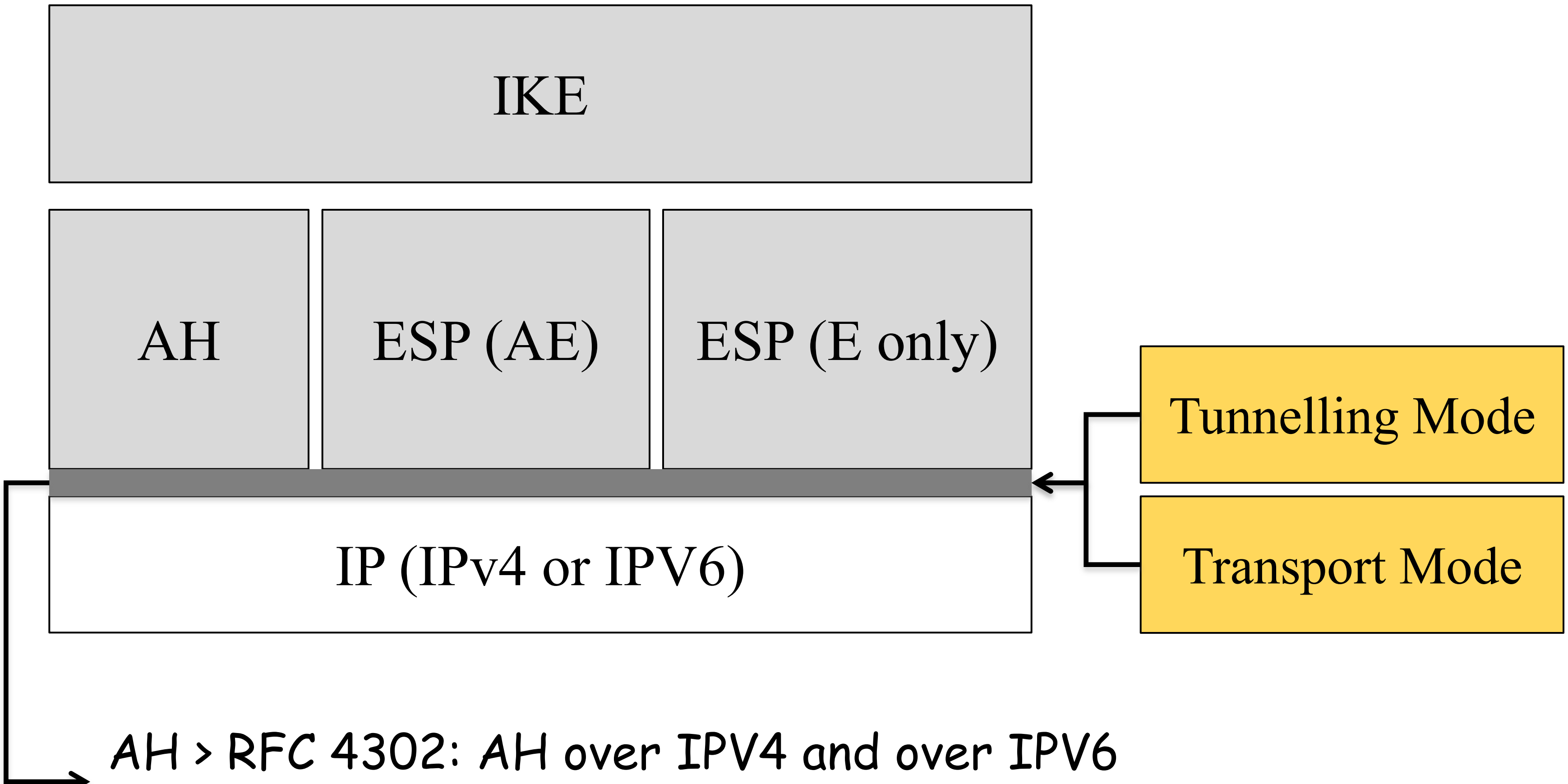
Roadmap / Outline

- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IP Security Stack (and Sub-Protocols)

- Sub-Protocols and Modes + Encapsulation (IPV4 or IPV6), … as well as other (tunneling) encapsulation options



AH > RFC 4302: AH over IPV4 and over IPV6
ESP > RFCs 4303, 4305: ESP over IPV4 and IPV6

Specific encapsulation of IPsec modes

- Depending on the IPSec modes, encapsulation of ESP and AH is done in a different way
- Combinations:
  - AH in Transport mode
  - AH in Tunnel mode

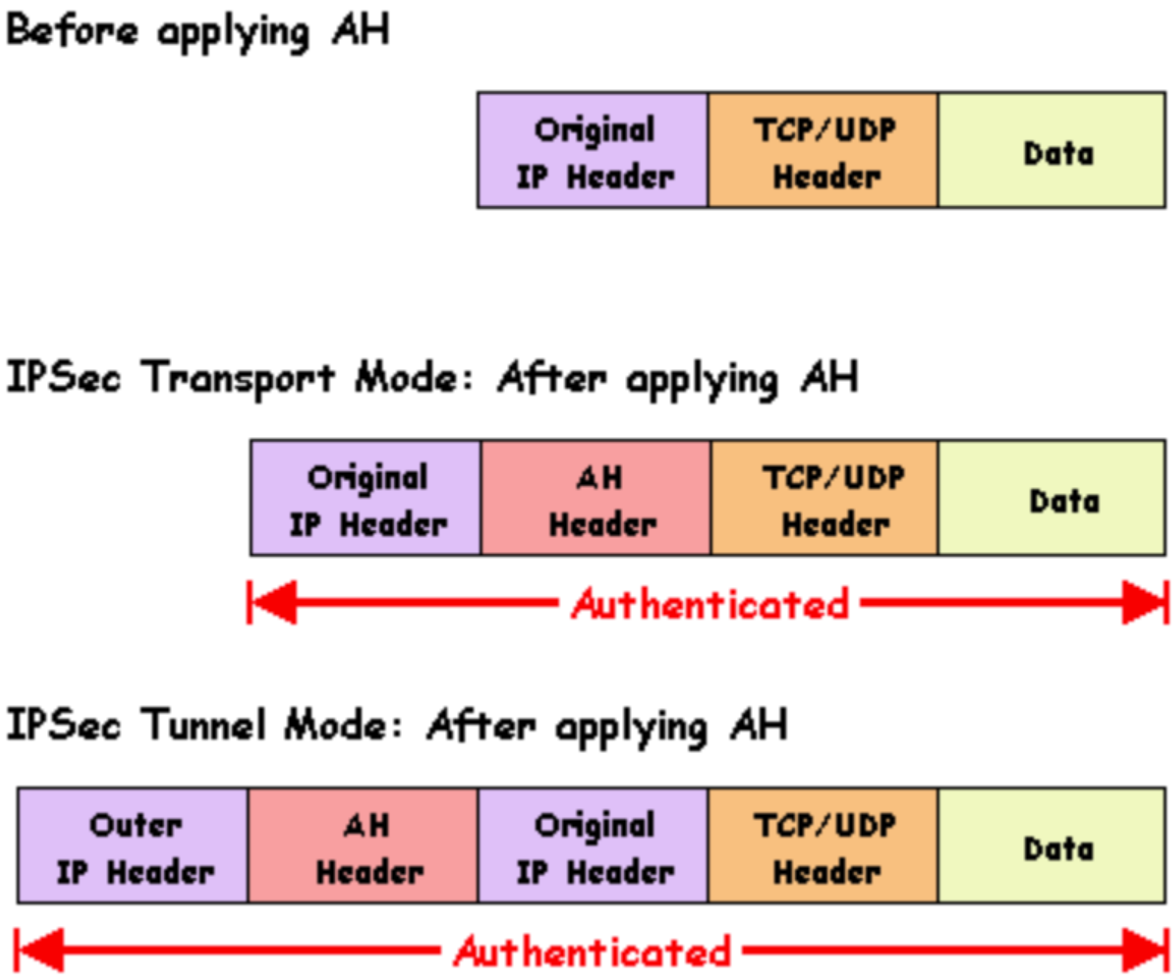
  - ESP-Authentication Only in Transport mode
  - ESP-Authentication Only in Tunnel mode
  - ESP-Auth & Encryption in Transport mode
  - ESP-Auth & Encryption in Transport mode
- Combinations imply on different provided security properties

IPSec Sub-Protocols and Modes

Support for six different protection behaviours and related SAs (IPSec Security Associations)

	Transport Mode SA	Tunnel Mode SA
AH	Authenticates IP payload and selected portions of IP header and IPv6 extension headers.	Authenticates entire inner IP packet (inner header plus IP payload) plus selected portions of outer IP header and outer IPv6 extension headers.
ESP	Encrypts IP payload and any IPv6 extension headers following the ESP header.	Encrypts entire inner IP packet.
ESP with Authentication	Encrypts IP payload and any IPv6 extension headers following the ESP header. Authenticates IP payload but not IP header.	Encrypts entire inner IP packet. Authenticates inner IP packet.

AH Processing in Transport and Tunnel modes



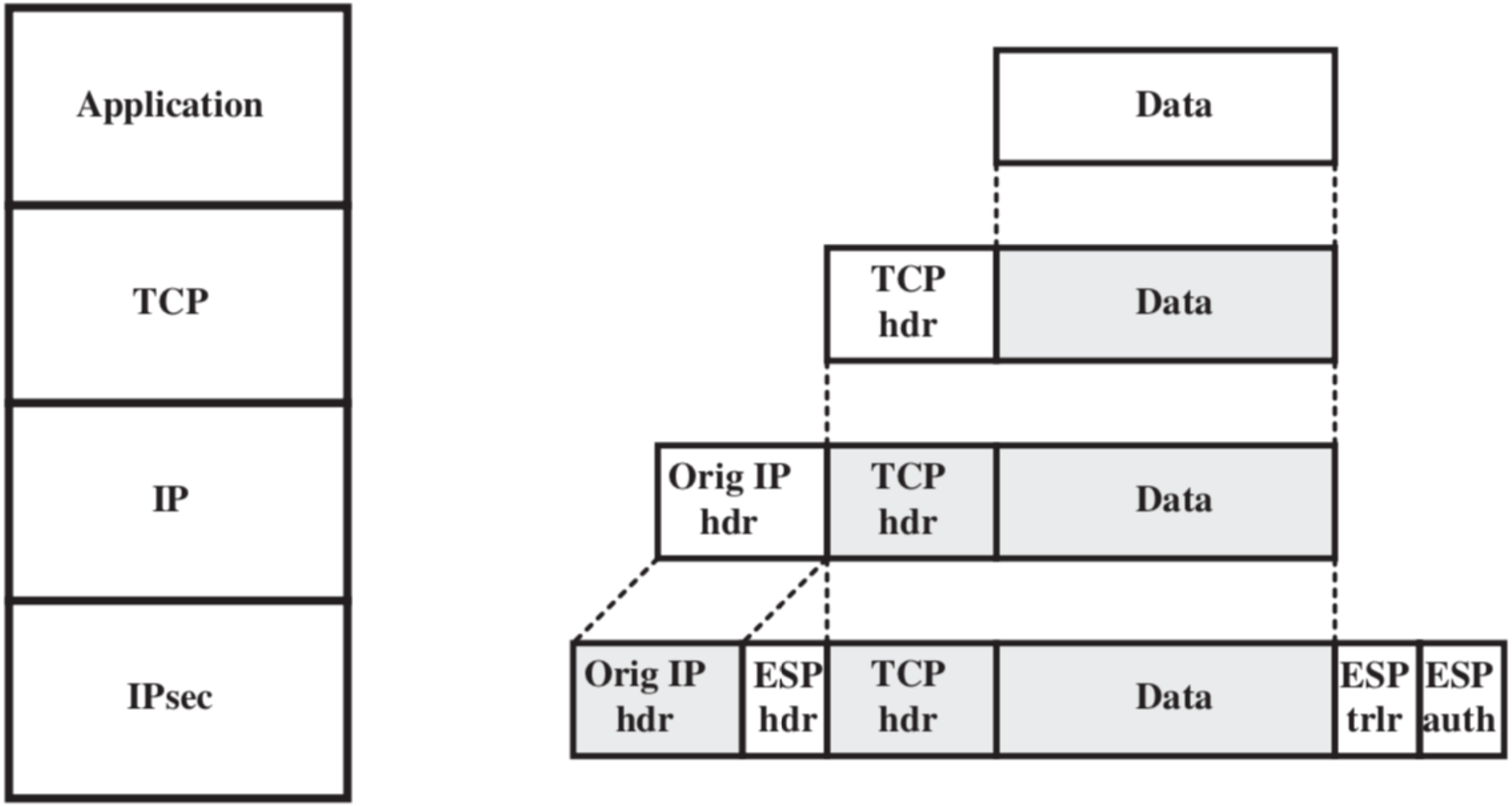
Transport Mode

Tunnel Mode

ESP Processing: Transport Mode

Transport Mode:

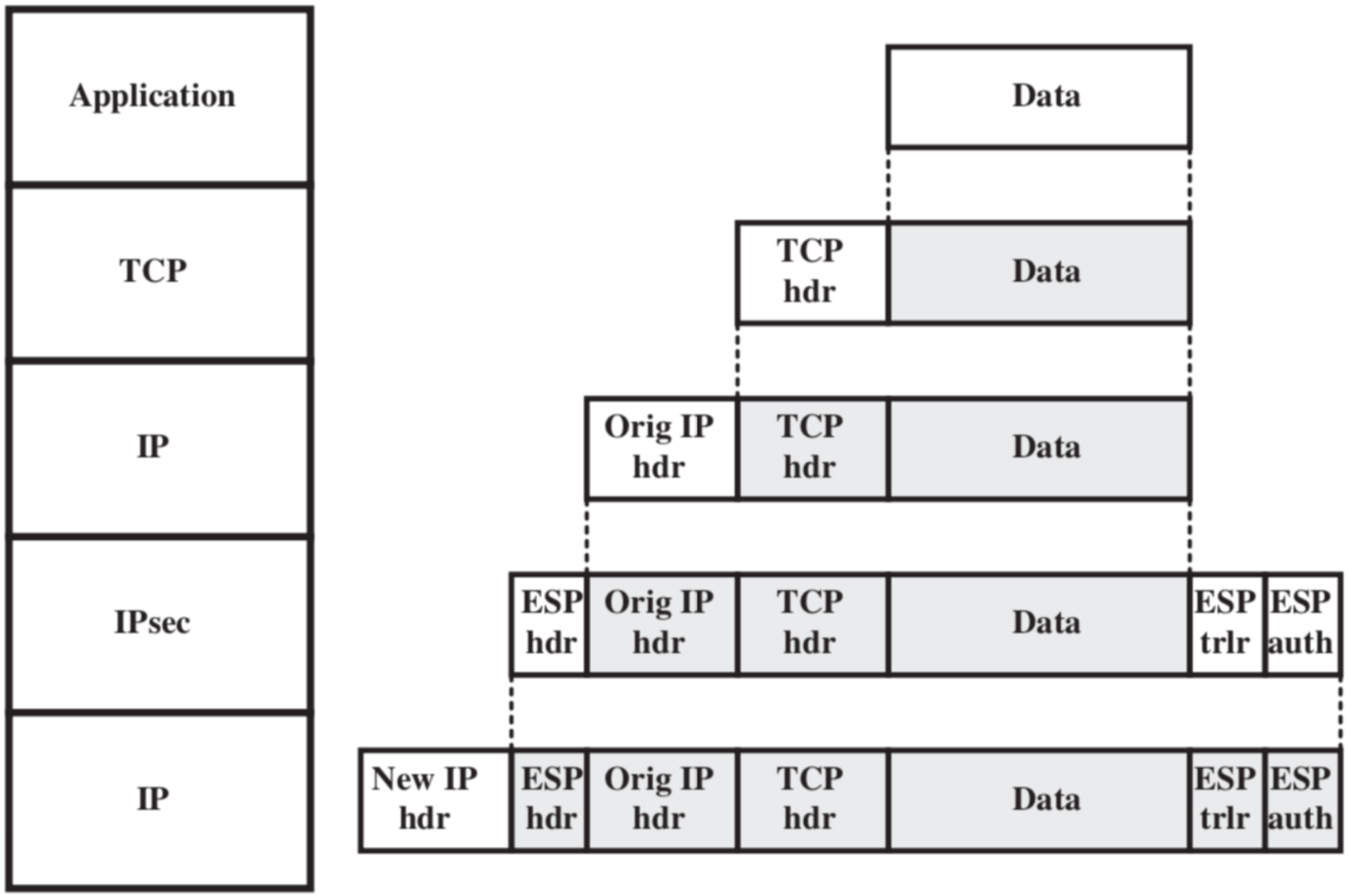
- End-to-End Security
- Host-to-Host



ESP Processing: Tunnel Mode

Tunnel Mode:

- Intermediary-Support
- Routers, Firewalls



ESP AE in Tunnel and Transport Modes

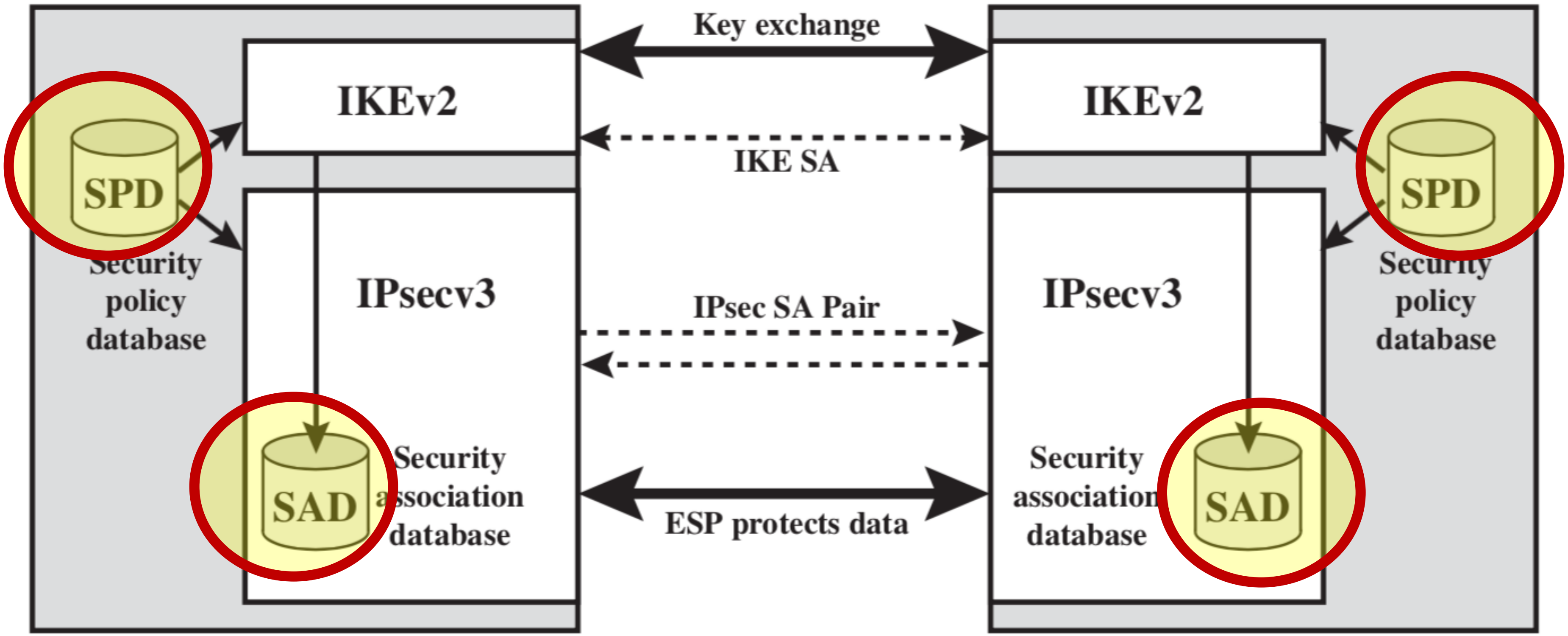
- **Transport modes (Host-Based, End-to-End)**
  - Encrypts entire IP packet
  - Limited Traffic Flow Protection. Why and How ?
  - End-to-End Protected Packets
  - No Switches nor LAN-to-LAN MiM on way can examine inner IP header and Payloads ! Issues ? How to address ?

- **Tunnel Mode (Router or FW Intermediation, possible use of NAT)**
  - Encrypts entire IP packet
  - Limited Traffic Flow Protection. Why and How ?
  - Add new header for "each" next hop
  - But no routers/firewalls on way can examine inner IP header and Payloads ! Issues ? How to address ?
  - Good for Secure VPNs, Gateway to Gateway security or Host-to-Relay Security

Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IPSec operation review



Management of Security Associations (SAD) and Security Policies (SPD) established and managed in IPSec endpoints
SPD and SAD as two persistent "Databases":

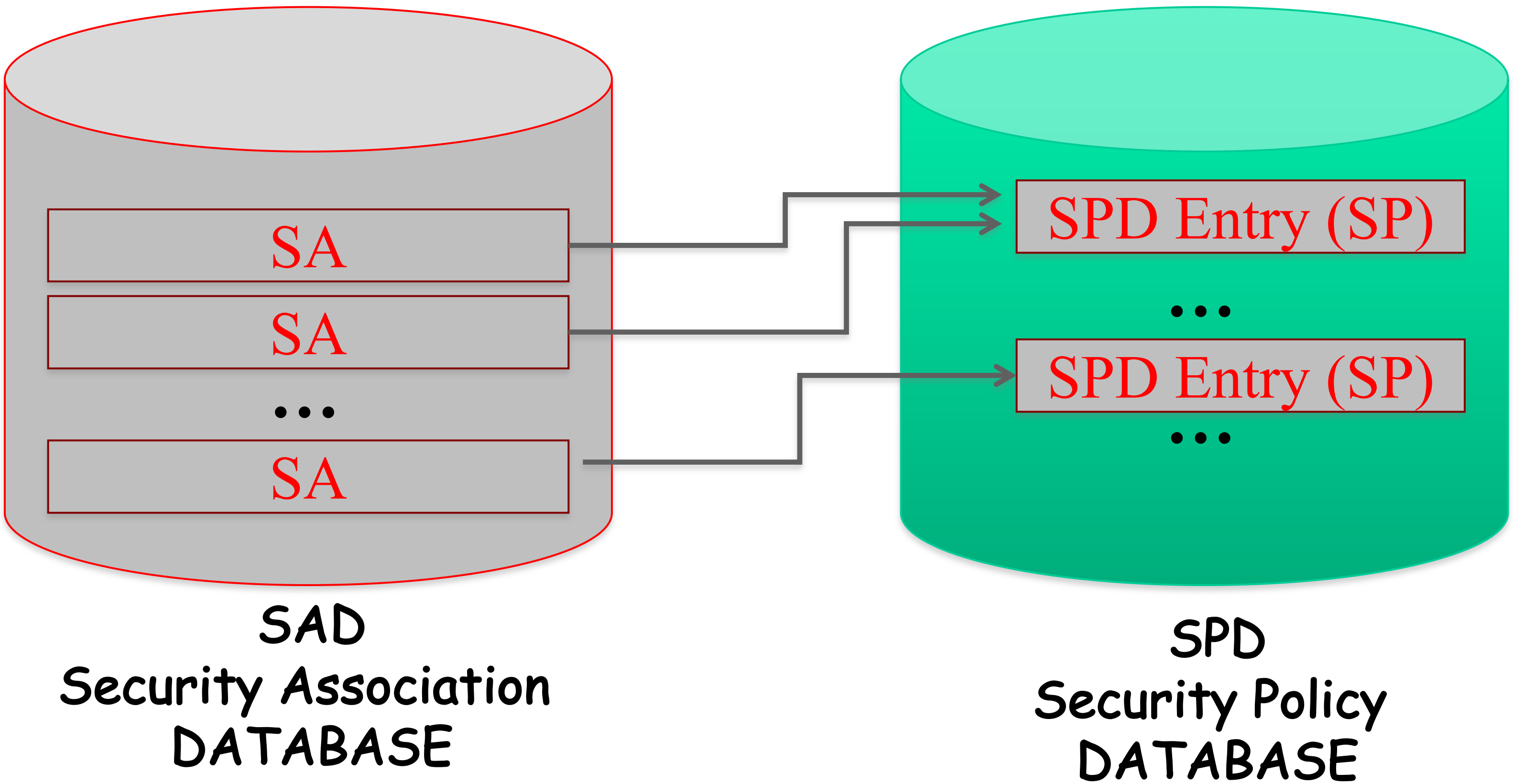
SADs, SPDs, and SAs

- **SAD (Security Association Database)**
  - Contains SAs (Security Associations) as entries
  - SA entries correspond to entries in the SPD

- **SPD (Security Policy Database)**
  - In the SPD, the IPSec policies for each Security Association are established and managed
  - Different SAs may share the same IPSec policy

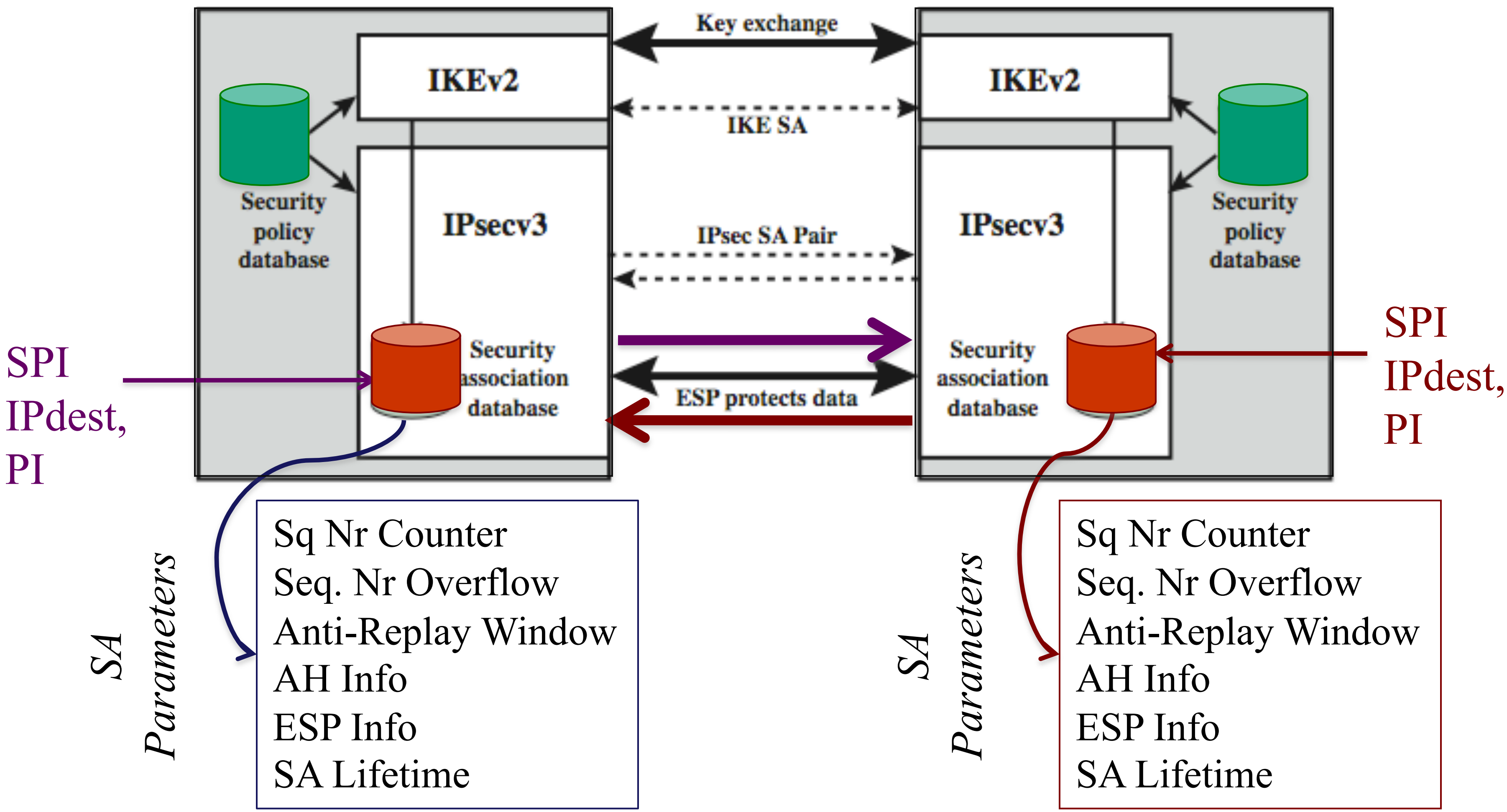
SADs, SPDs, and SAs

Each SA: defines a "One-Way" Relationship related to One-Way" IP FLOW between an IPSender sender & IPSec receiver that affords security policies for traffic flow in the right sense

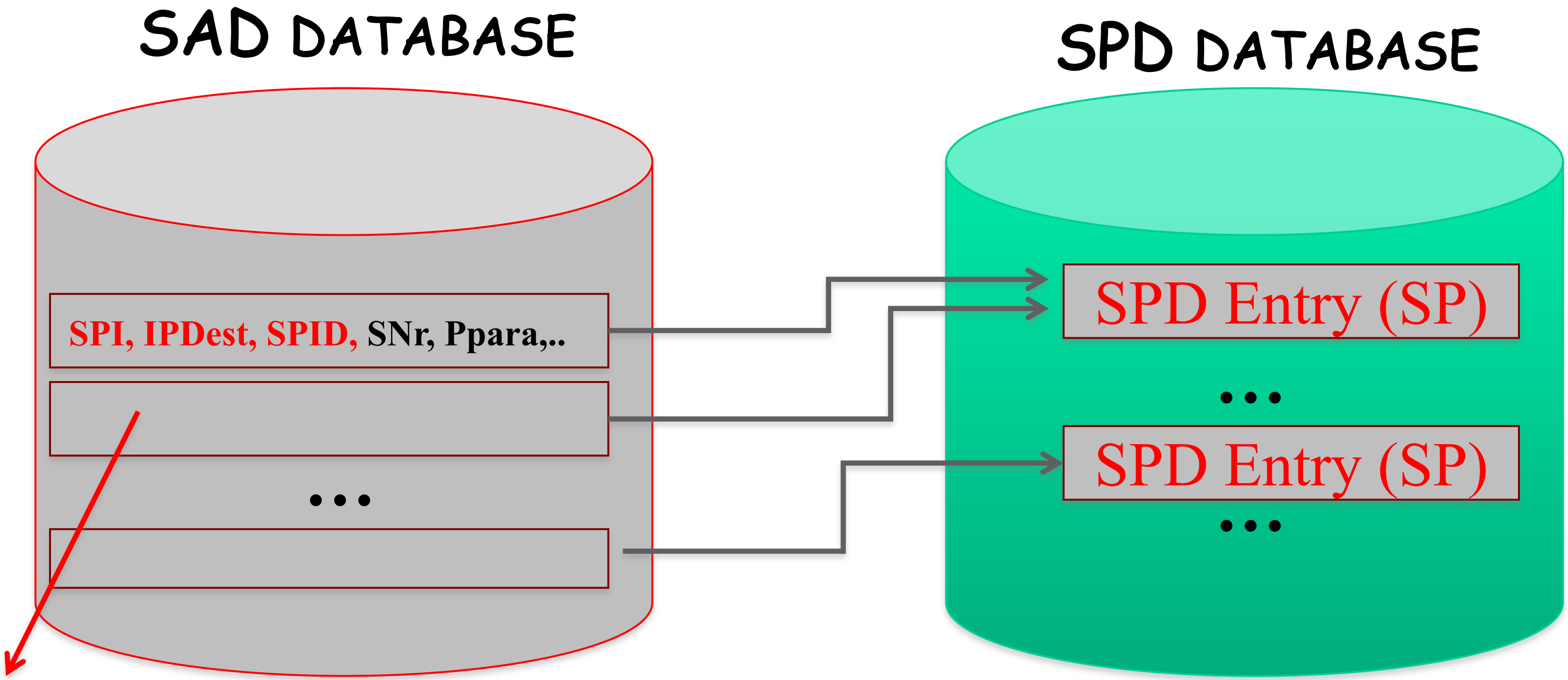


IPSec security policy management

- IPSec architecture: IKEv2 + SPD and SAD
- Unidirectional Security Associations



SADs, SPDs, SAs (and info in SAs)

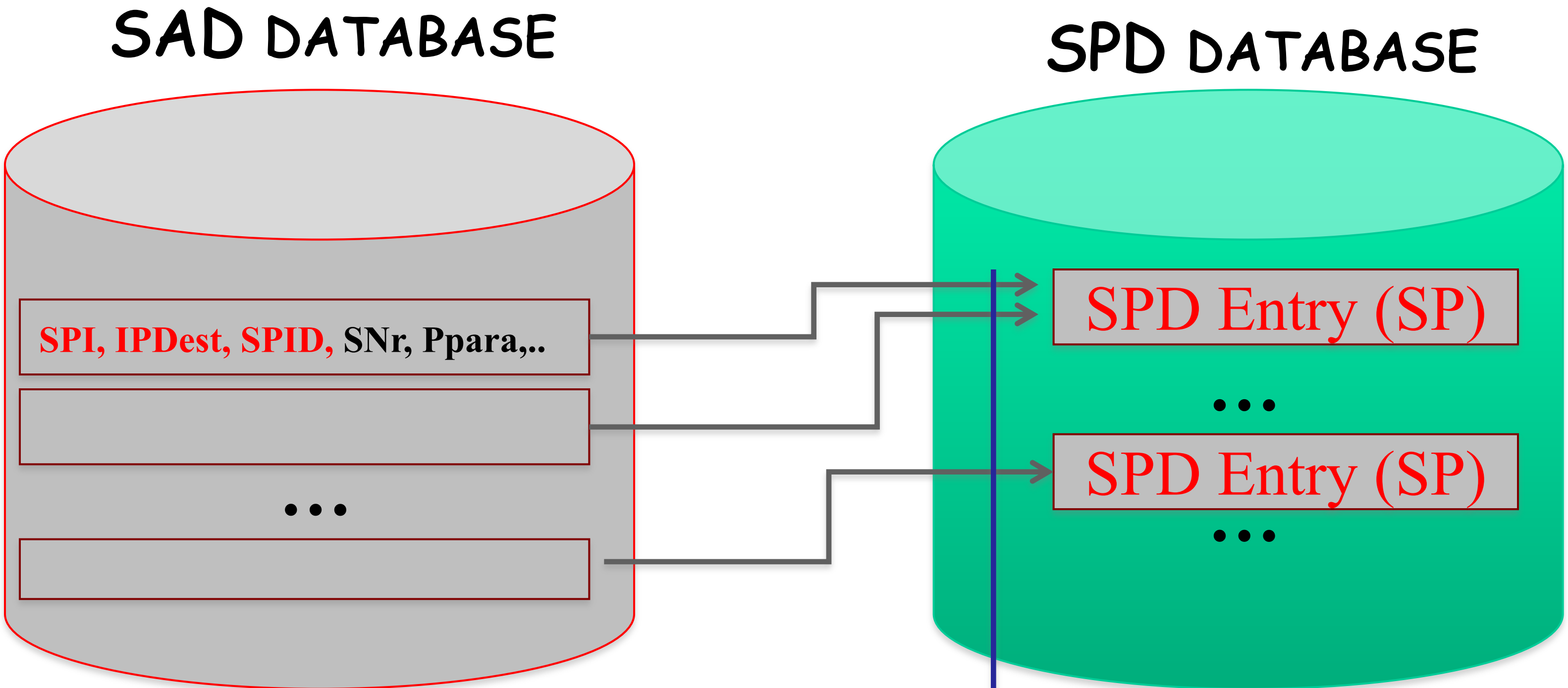


- An SA is defined by 3 parameters:
  - SPI: Security Parameters Index (SPI)
    - Identifier travelling in the IPSec packet headers
  - IP Destination Address
  - Security Protocol Identifier (SPID)

… and additionally some other parameters
  Seq nr., AH & ESP info, SA lifetime, etc

Sq Nr Counter
Seq. Nr Overflow
Anti-Replay Window
AH Info (Keys)
ESP Info (Keys, IVs)
SA Lifetime

SADs, SPDs, SAs (and info in SAs)



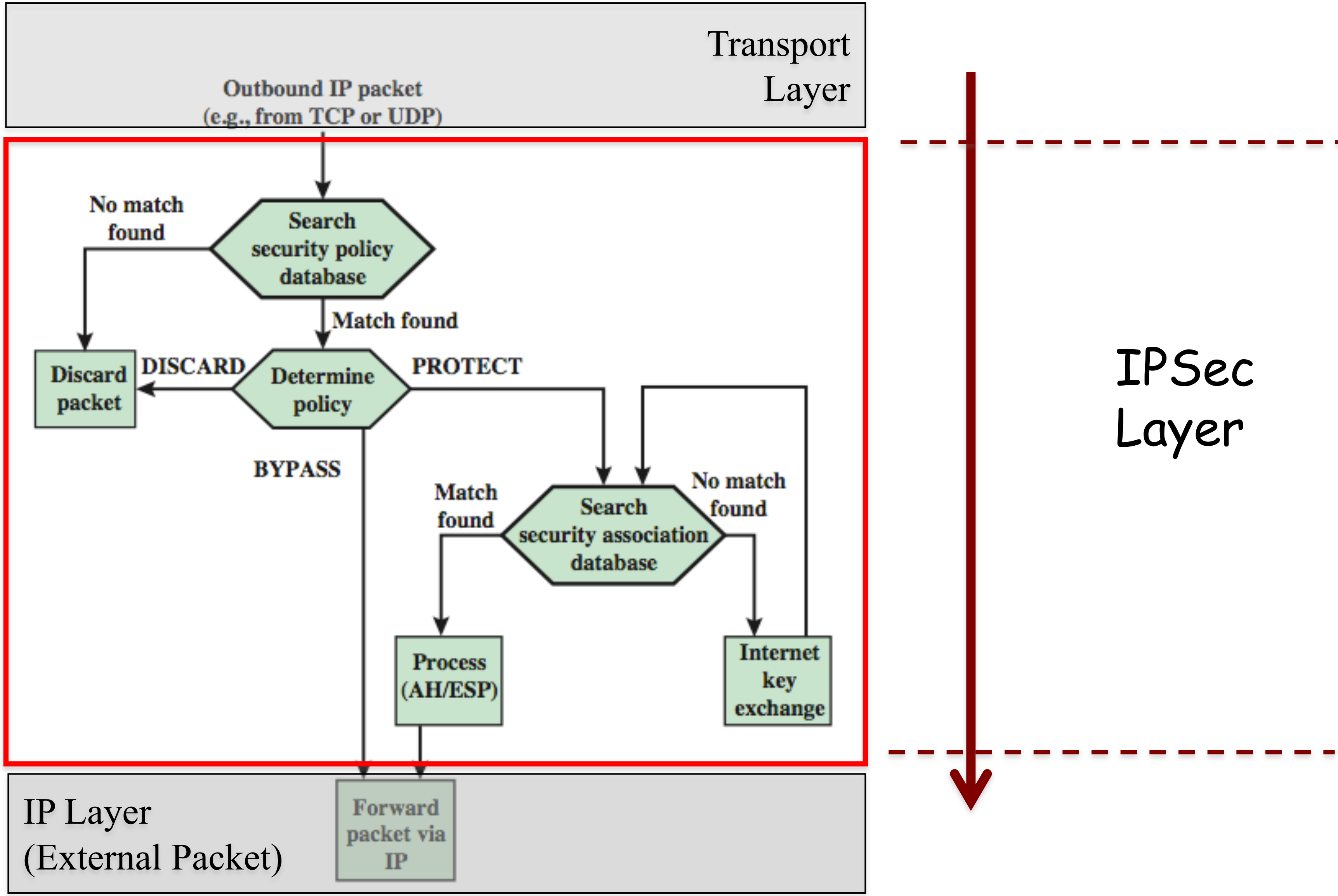
Protocol	Local IP	Port	Remote IP	Port	Action	Comment
UDP	1.2.3.101	500	*	500	BYPASS	IKE
ICMP	1.2.3.101	*	*	*	BYPASS	Error messages
*	1.2.3.101	*	1.2.3.0/24	*	PROTECT: ESP intransport-mode	Encrypt intranet traffic
TCP	1.2.3.101	*	1.2.4.10	80	PROTECT: ESP intransport-mode	Encrypt to server
TCP	1.2.3.101	*	1.2.4.10	443	BYPASS	TLS: avoid double encryption
*	1.2.3.101	*	1.2.4.0/24	*	DISCARD	Others in DMZ
*	1.2.3.101	*	*	*	BYPASS	Internet

Security Policy Database Implementation

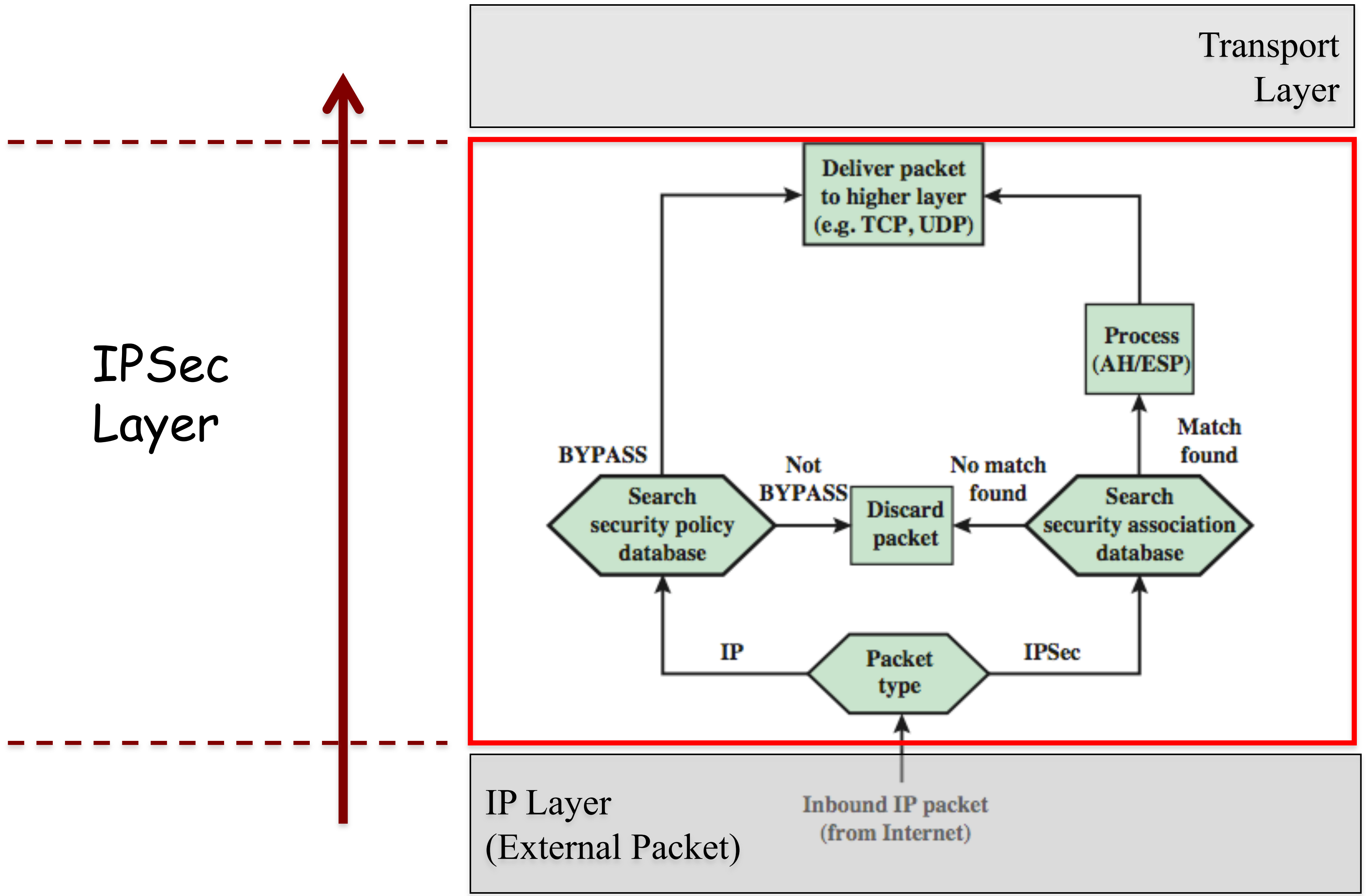
- Relates IP traffic to specific SAs
  - Match subset of IP traffic to the relevant SA
  - Use Selectors to filter outgoing traffic to map
    - Different selectors can be used (see bibliography)
  - Based on: Local & Remote IP addresses, Next layer Protocol, Name, Local & Remote Ports

Protocol	Local IP	Port	Remote IP	Port	Action	Comment
UDP	1.2.3.101	500	*	500	BYPASS	IKE
ICMP	1.2.3.101	*	*	*	BYPASS	Error messages
*	1.2.3.101	*	1.2.3.0/24	*	PROTECT: ESP intransport-mode	Encrypt intranet traffic
TCP	1.2.3.101	*	1.2.4.10	80	PROTECT: ESP intransport-mode	Encrypt to server
TCP	1.2.3.101	*	1.2.4.10	443	BYPASS	TLS: avoid double encryption
*	1.2.3.101	*	1.2.4.0/24	*	DISCARD	Others in DMZ
*	1.2.3.101	*	*	*	BYPASS	Internet

IPSec: Processing of Outbound Packets

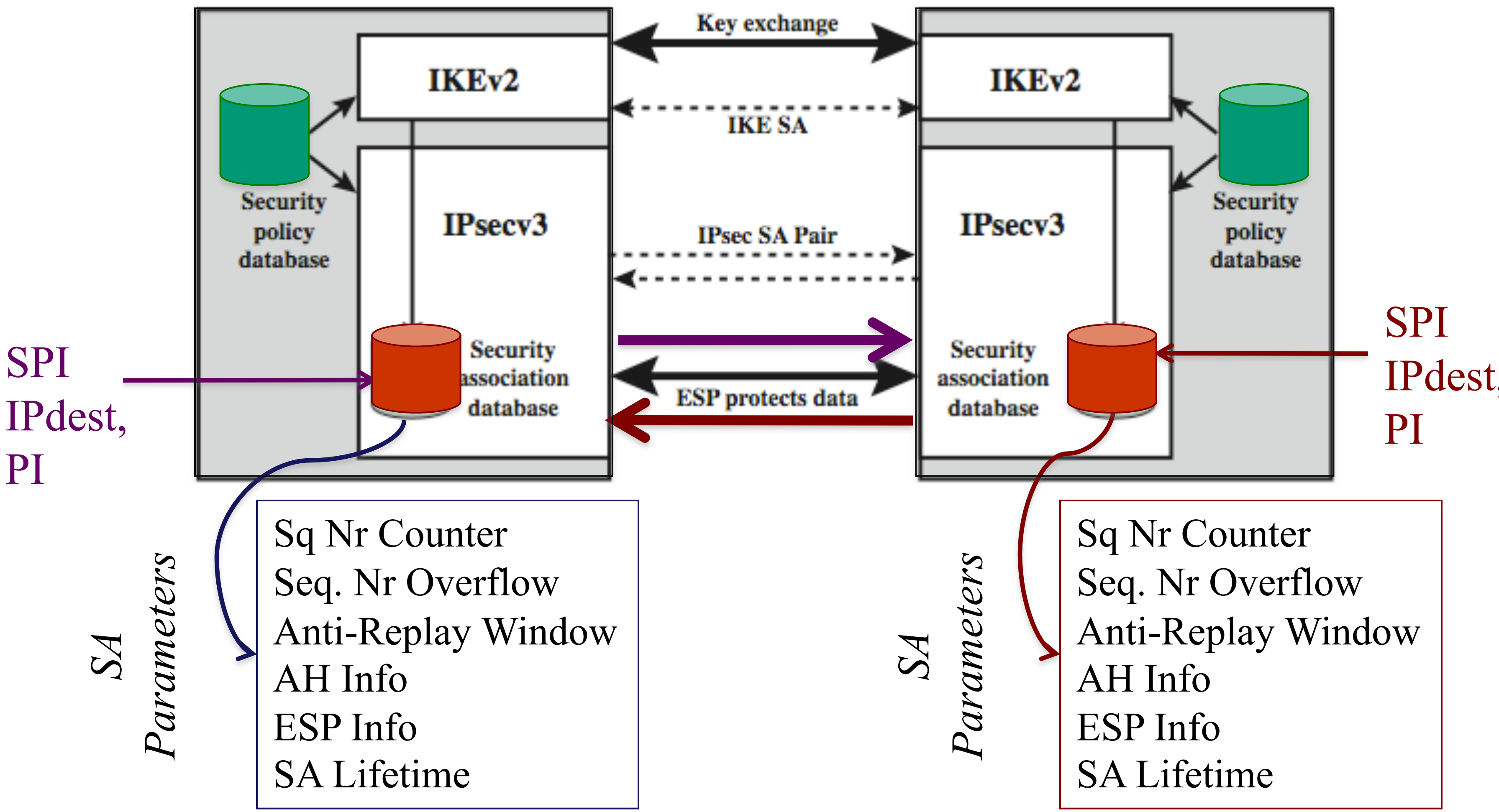


IPSec: Processing of Inbound Packets



IPSec security policy management

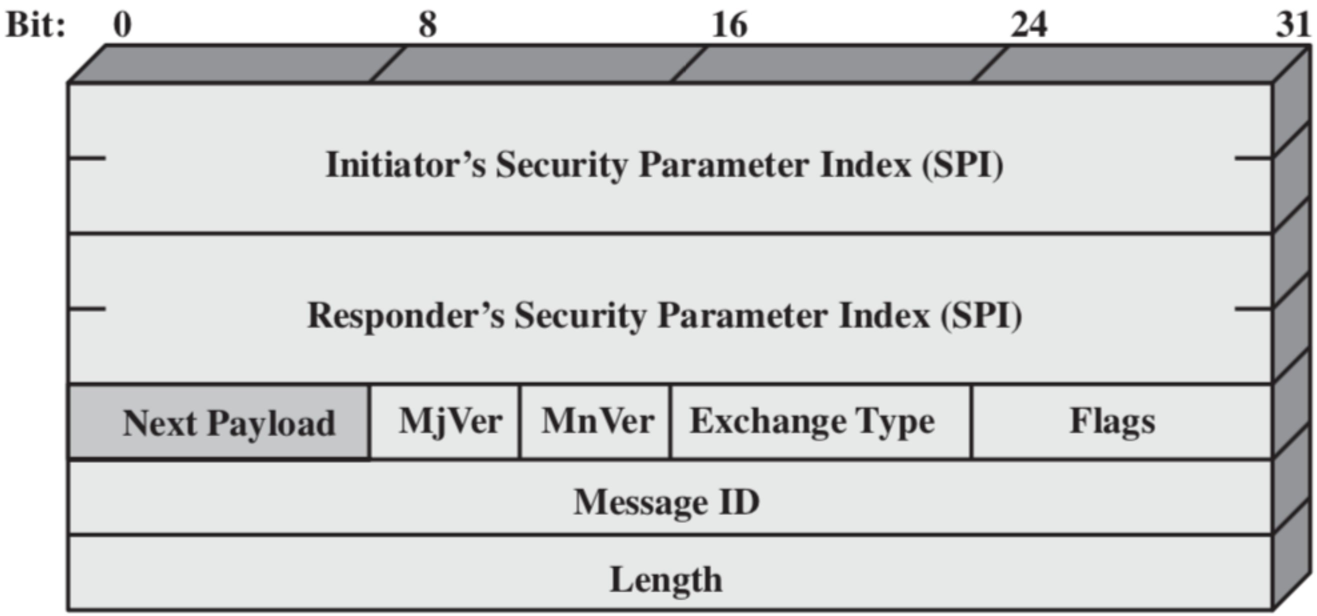
- IPSec architecture: IKEv2 + SPD and SAD
- Unidirectional Security Associations



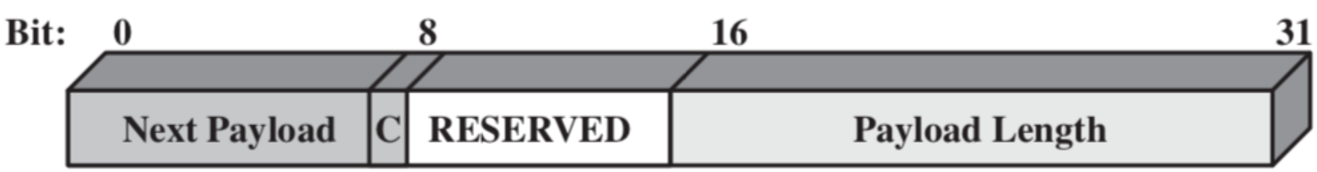
Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IKEv2 and ISAKMP

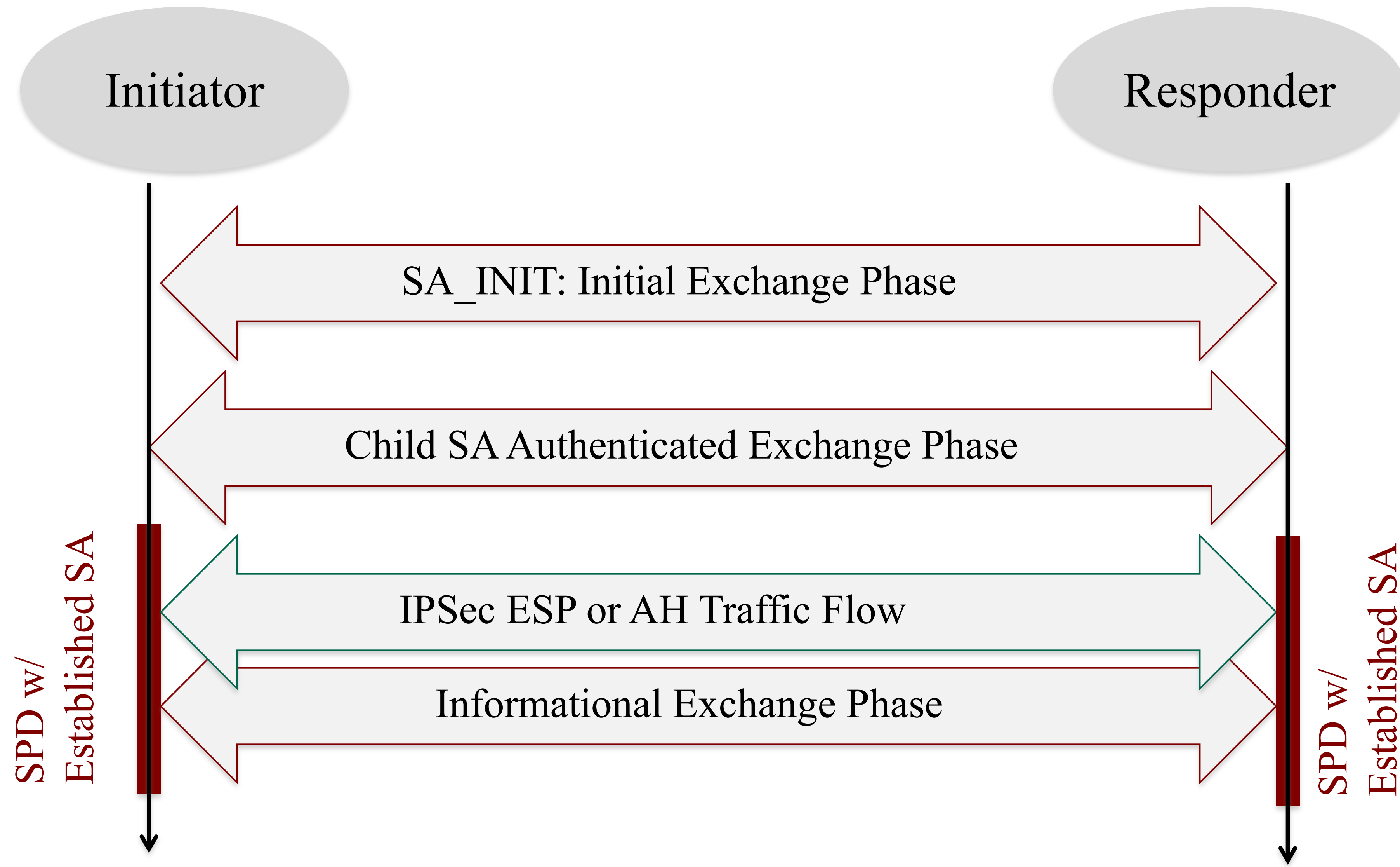


(a) IKE header

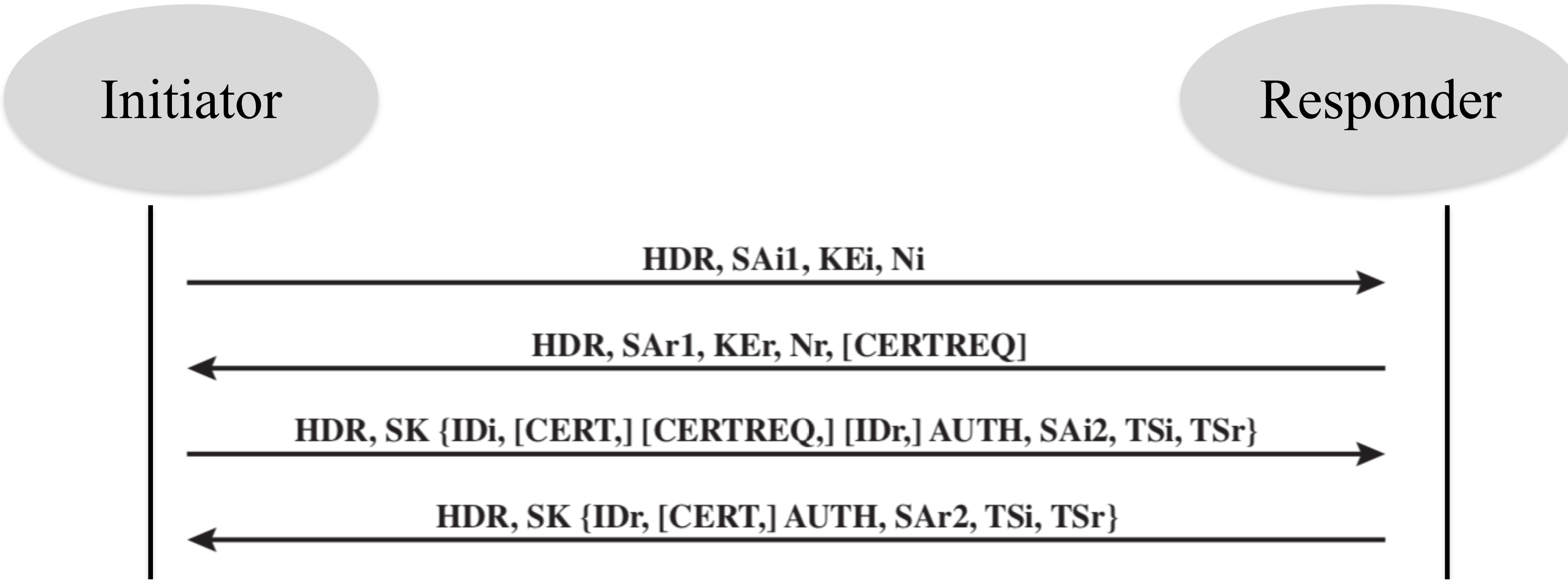


(b) Generic Payload header

IKEv2 protocol exchanges: establishment of SAs (in SAD) and SPs (in SPD)



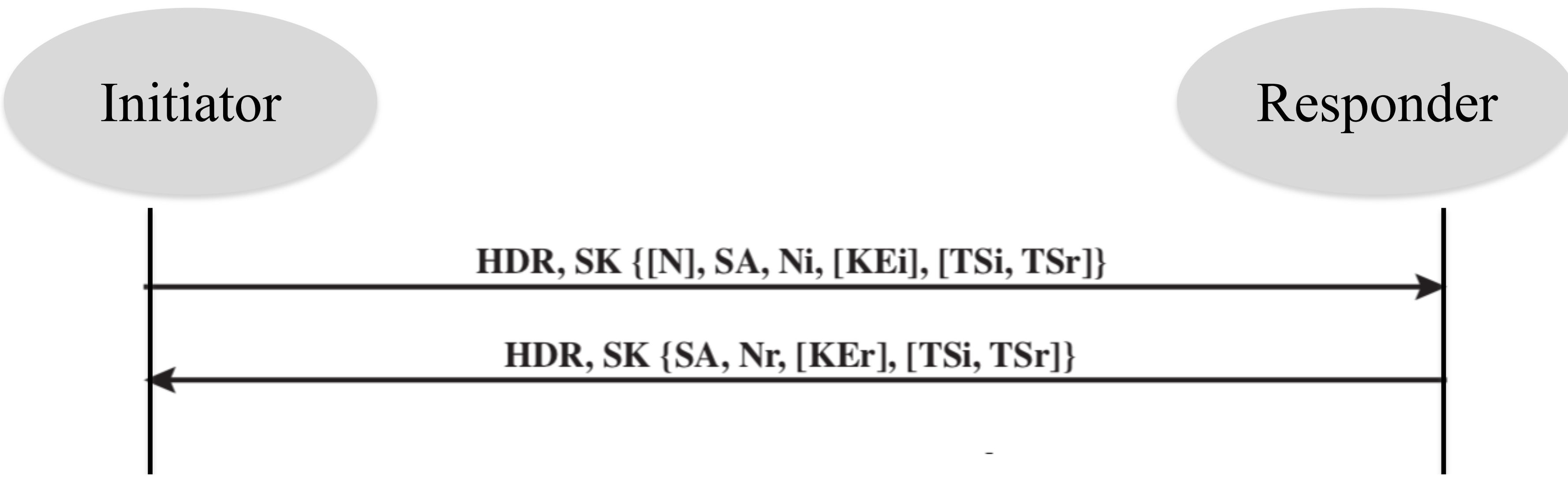
IKEv2 Exchanges: SA_INIT Phase



HDR = IKE header
SAx1 = offered and chosen algorithms, DH group
KEx = Diffie-Hellman public key
Nx= nonces
CERTREQ = Certificate request
IDx = identity
CERT = certificate

SK {...} = MAC and encrypt
AUTH = Authentication
SAx2 = algorithms, parameters for IPsec SA
TSx = traffic selectors for IPsec SA
N = Notify
D = Delete
CP = Configuration

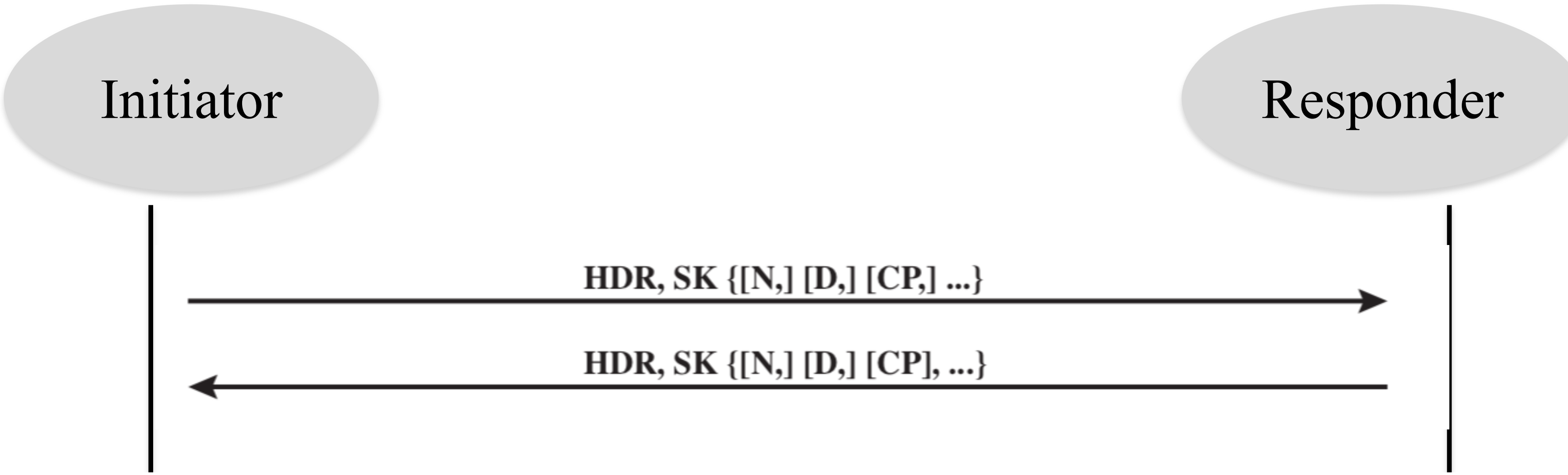
IKEv2 Exchanges: Child SA Phase



HDR = IKE header
SAx1 = offered and chosen algorithms, DH group
KEx = Diffie-Hellman public key
Nx= nonces
CERTREQ = Certificate request
IDx = identity
CERT = certificate

SK {...} = MAC and encrypt
AUTH = Authentication
SAx2 = algorithms, parameters for IPsec SA
TSx = traffic selectors for IPsec SA
N = Notify
D = Delete
CP = Configuration

IKEv2 Exchanges: Informational Phase



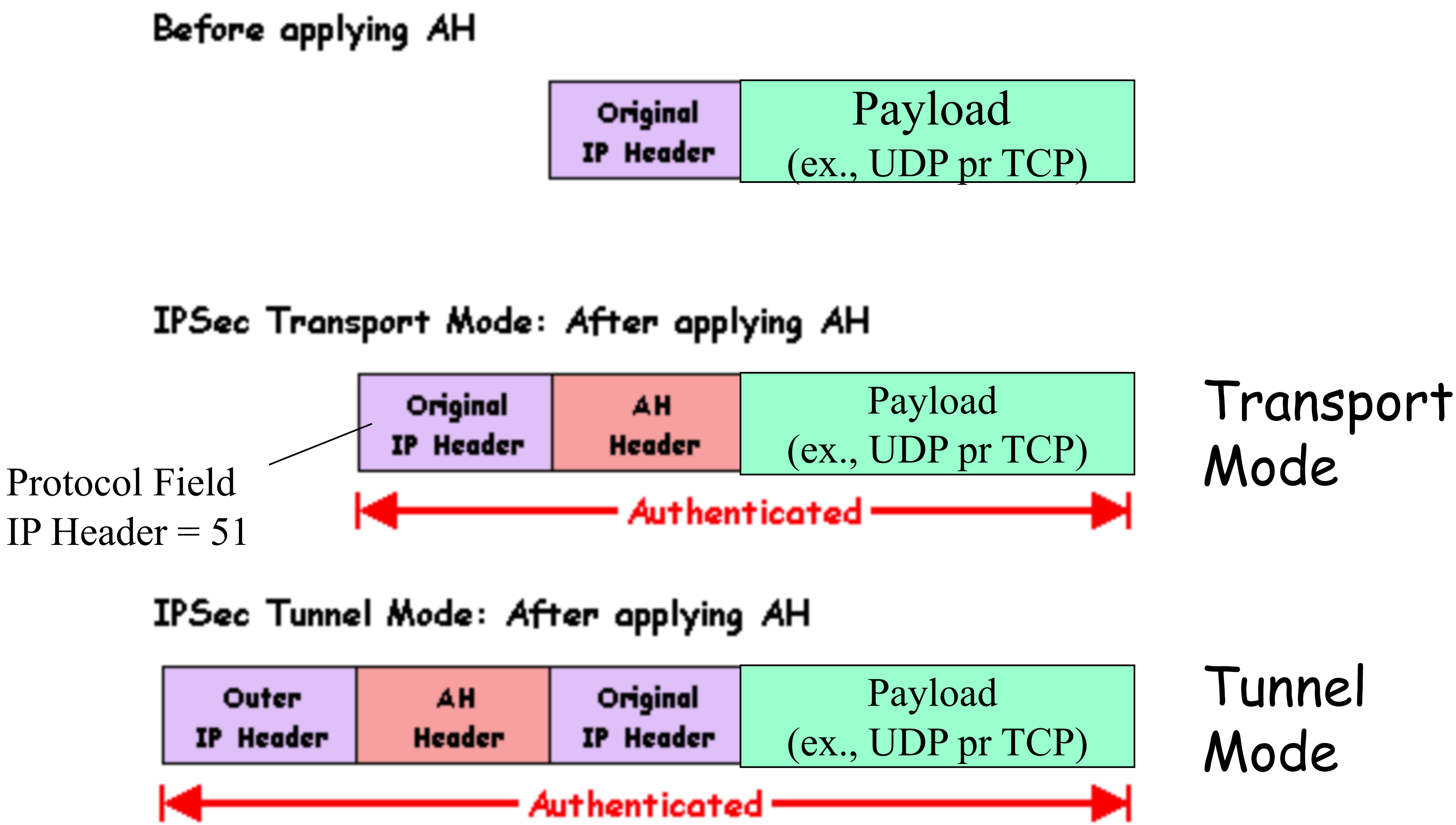
HDR = IKE header
SAx1 = offered and chosen algorithms, DH group
KEx = Diffie-Hellman public key
Nx= nonces
CERTREQ = Certificate request
IDx = identity
CERT = certificate

SK {...} = MAC and encrypt
AUTH = Authentication
SAx2 = algorithms, parameters for IPsec SA
TSx = traffic selectors for IPsec SA
N = Notify
D = Delete
CP = Configuration

Roadmap / Outline

- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

AH/IP in Transport and Tunnel modes



AH Protocol

AH Header

0	8	16	31
Next Header	Payload Length	RES (NU)	RES (NU)
SPI			
Seq. number			
Auth Data (variable size Depending on the Message Authentication Method in the SA setup) (Default values in current implementations: HMAC => 96 bits)			

Auth Data (described in RFC 2402 ... => RFC 4302)
Contains an ICV (Integrity Check Value) computed as a 96 bit MAC (HMAC-MD5-96, ou HMAC-SHA-1.96)

Example of AH encapsulation (Wireshark)

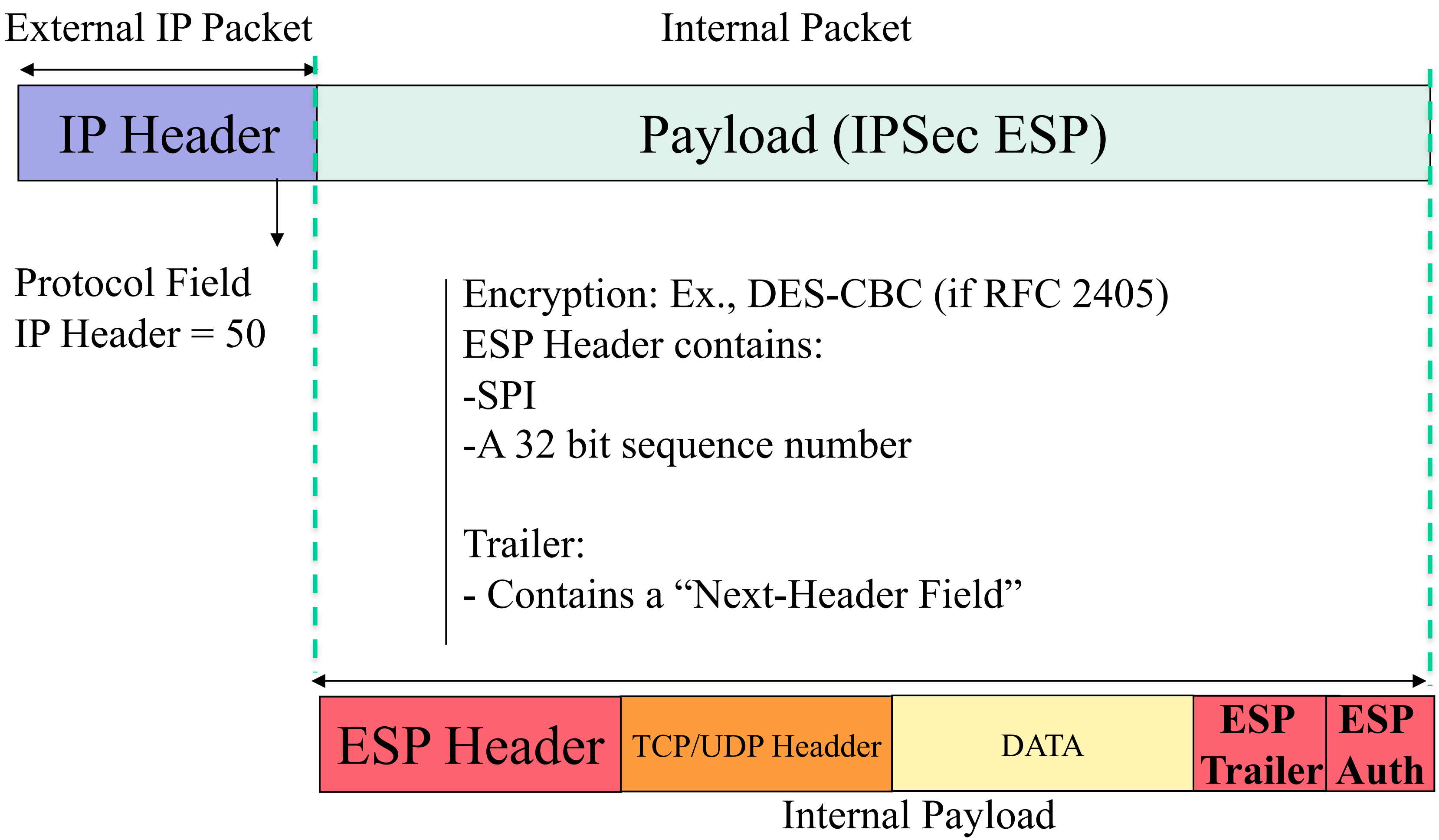
```
⊞ Frame 1: 158 bytes on wire (1264 bits), 158 bytes captured (1264 bits) on interface 0
⊞ Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
⊟ Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 144
    Identification: 0x0215 (533)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Authentication Header (51)
  ⊞ Header checksum: 0x1fd2 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
⊟ Authentication Header
    Next Header: IPIP (0x04)
    Length: 24
    AH SPI: 0x646adc80
    AH Sequence: 5
    AH ICV: 606d214066853c0390cfe577
⊟ Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 100
    Identification: 0x003c (60)
  ⊟ Flags: 0x00
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
    Fragment offset: 0
    Time to live: 255
    Protocol: ICMP (1)
  ⊞ Header checksum: 0x2209 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
⊞ Internet Control Message Protocol
```

Trace Ex:

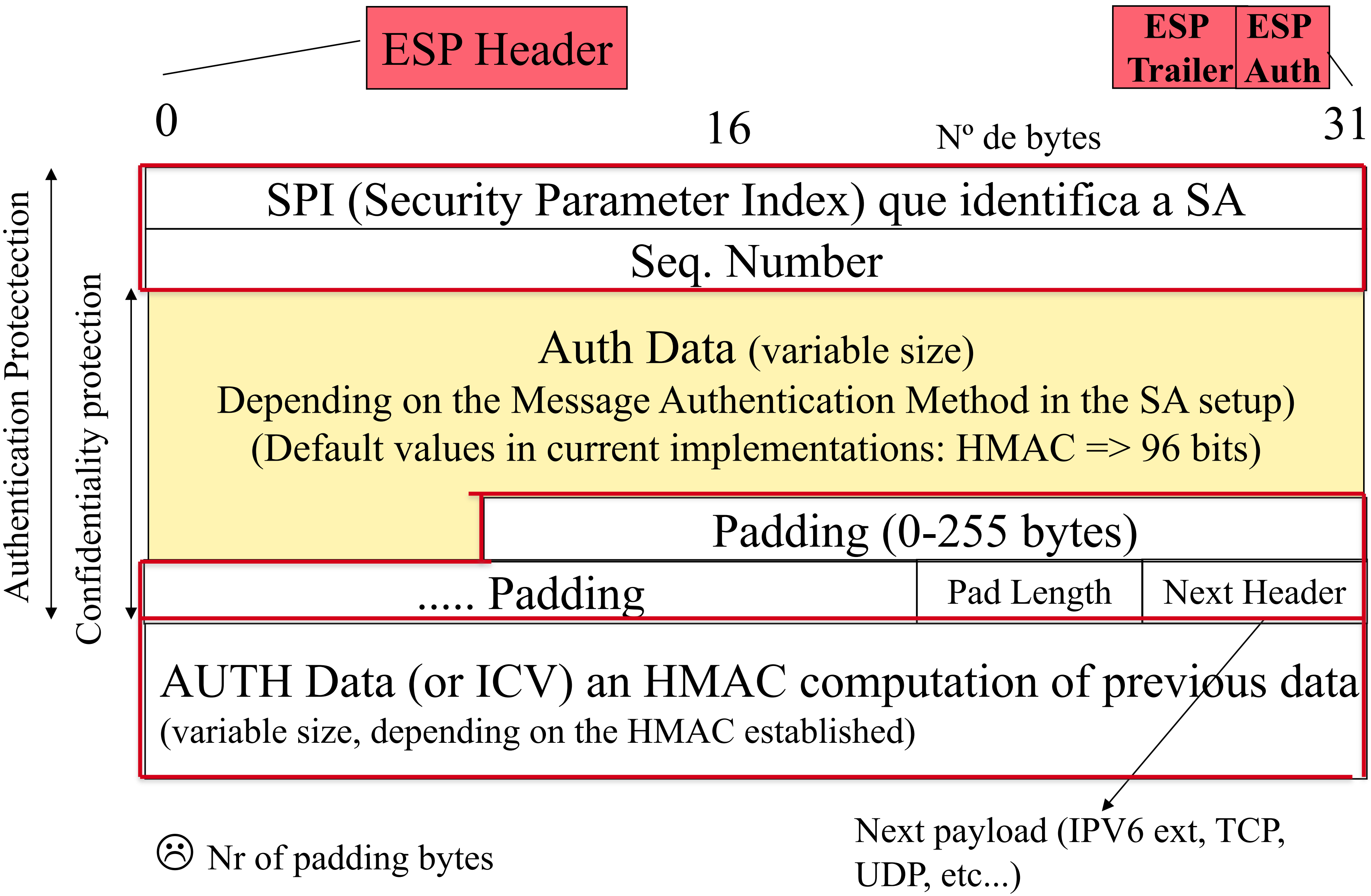
Protected ICMP w/ AH / IP

ESP - Encapsulation Security Payload

- More complex than AH (more overhead but more security concerns)



ESP


ESP Header
ESP Trailer
ESP Auth
0
16
Nº de bytes
31
Authentication Protectection
Confidentiality protection
SPI (Security Parameter Index) que identifica a SA
Seq. Number
Auth Data (variable size)
Depending on the Message Authentication Method in the SA setup)
(Default values in current implementations: HMAC => 96 bits)
Padding (0-255 bytes)
..... Padding
Pad Length
Next Header
AUTH Data (or ICV) an HMAC computation of previous data
(variable size, depending on the HMAC established)
☹ Nr of padding bytes
Next payload (IPV6 ext, TCP, UDP, etc...)

Encryption & Authentication Algorithms & Padding Processing

- ESP can encrypt payload data, padding, pad length, and next header fields
  - If needed have IV at start of payload data
  - Provides message content confidentiality, data origin authentication, connectionless integrity, an anti-replay service, limited traffic flow confidentiality
- ESP can have optional ICV for integrity
  - Is computed after encryption is performed
- ESP uses padding
  - To expand plaintext to required length
  - To align pad length and next header fields
  - To provide partial traffic flow confidentiality

ESP (wireshark trace example)

```
⊞ Frame 2: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits) on interface 0
⊞ Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
⊟ Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 168
    Identification: 0x023e (574)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Encap Security Payload (50)
  ⊞ Header checksum: 0x1f92 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
⊟ Encapsulating Security Payload
    ESP SPI: 0x8bb181a7 (2343666087)
    ESP Sequence: 5
```

Trace Ex:

Protected IP w/ESP / IP

Roadmap / Outline

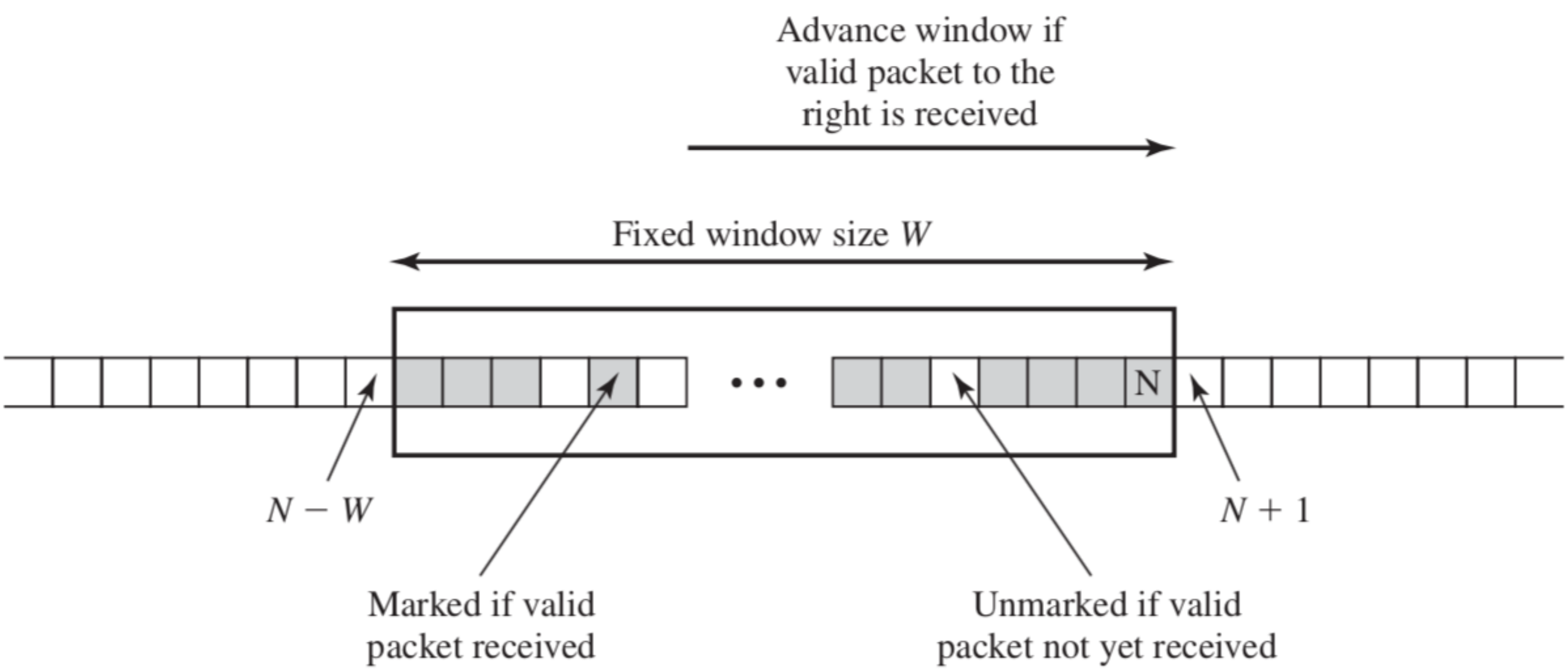
- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

Anti-Replay Service

- Replay: what if attacker resends a copy of an authenticated packet (IP Packet Replaying attack) ?
- IPSec Countermeasure: Use of protected sequence number (SN) to thwart the attack
  - Sender initializes sequence number to 0 when a new SA is established (ex., establishment via IKE/ISAKMP)
    - Increment SN for each packet
    - Must not exceed limit of $2^{32} - 1$
      - Danger of reuse (overflow):
  - Receiver only accepts packets with valid authentication proof and seq numbers within a window of ($N - W + 1$)
- … What what if packets arrive out of order ?
    - Remember: IP Traffic can arrive out-of-order

Out-of-Order packets and control

- IPSec solution: Sliding window control



Processing of anti-replay windows and control of the advance of the control window

- If received packet falls with in the window and is new
  - Check IPSec packet and MAC validity
  - If the packet is authenticated, the corresponding slot in the window is marked (valid – authenticated packet)
- If received packet is to the right of the window and is new
  - Check IPSec packet and MAC validity.
  - If the packet is authenticated, the window is advanced
  - so that this sequence number is the right edge of the window, and the corresponding slot in the window is marked (valid packet).
- If received packet is to the left of the window or if authentication fails
  - the packet is discarded; generates a local auditable event (logging).

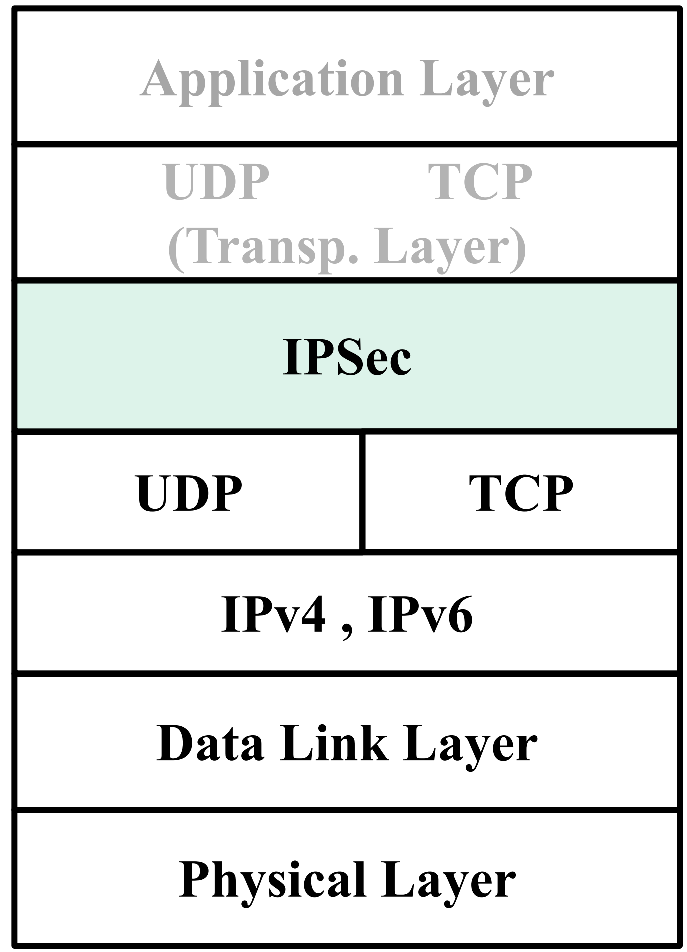
Roadmap / Outline

- IPSec (IP Security)
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

Possible IP Security Encapsulations IPsec/Transport Layer

Base encapsulation: IPSec/IP (v4 or v6)
But other flexible encapsulation forms in a TCP/IP stack ar also possible

Ex: IPsec/Transport Layer (TCP or UDP)

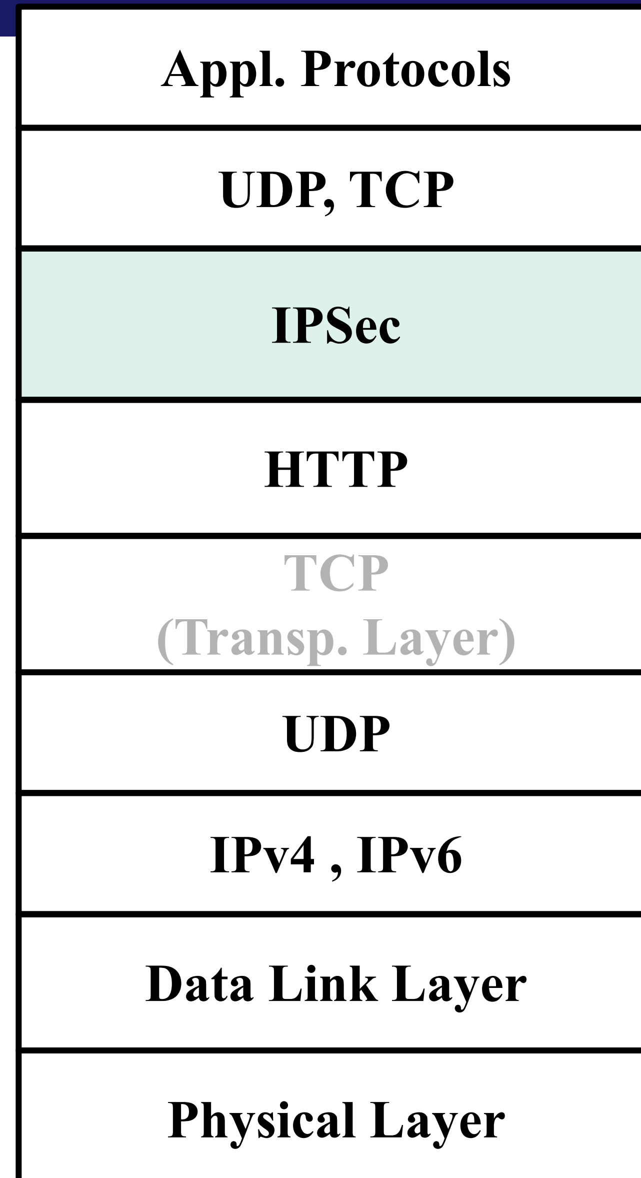


Other Possible IP Security Encapsulations

Application -Level Encapsulation

IPSec is a Security Stack of Sub Protocols for IP Traffic Protection

Ex: supported by HTTP

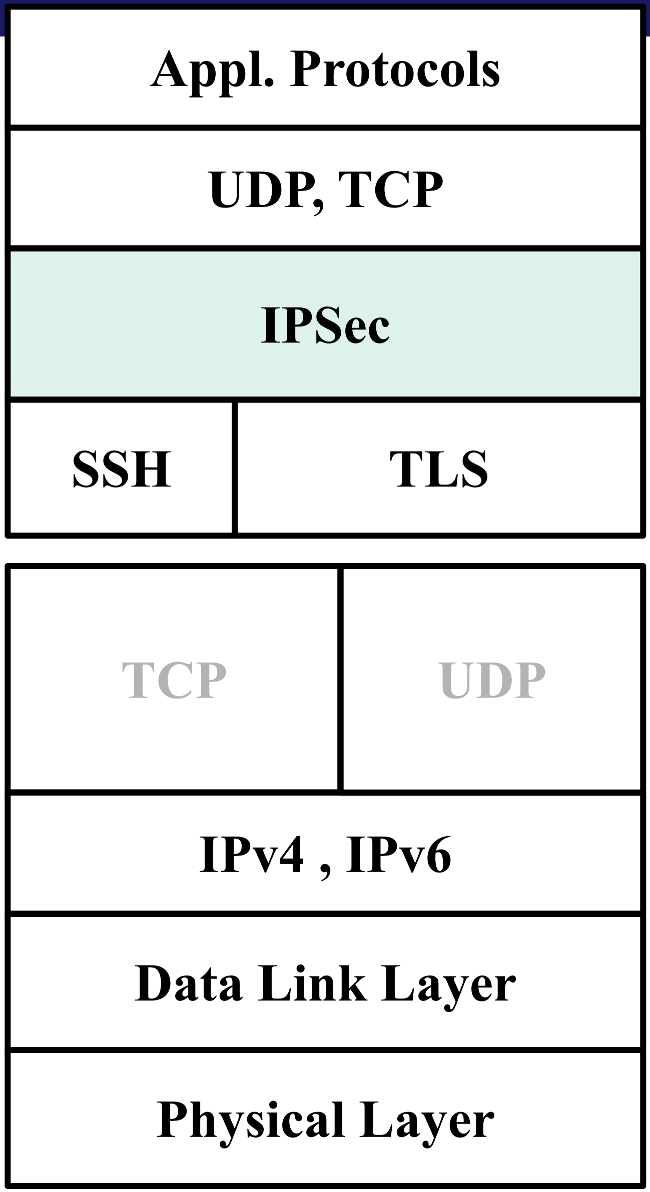


Other Possible IP Security Encapsulations

SSH or TLS encapsulation

IPSec is a Security Stack of Sub Protocols for IP Traffic Protection

Ex: supported by SSH or TLS Tunnels

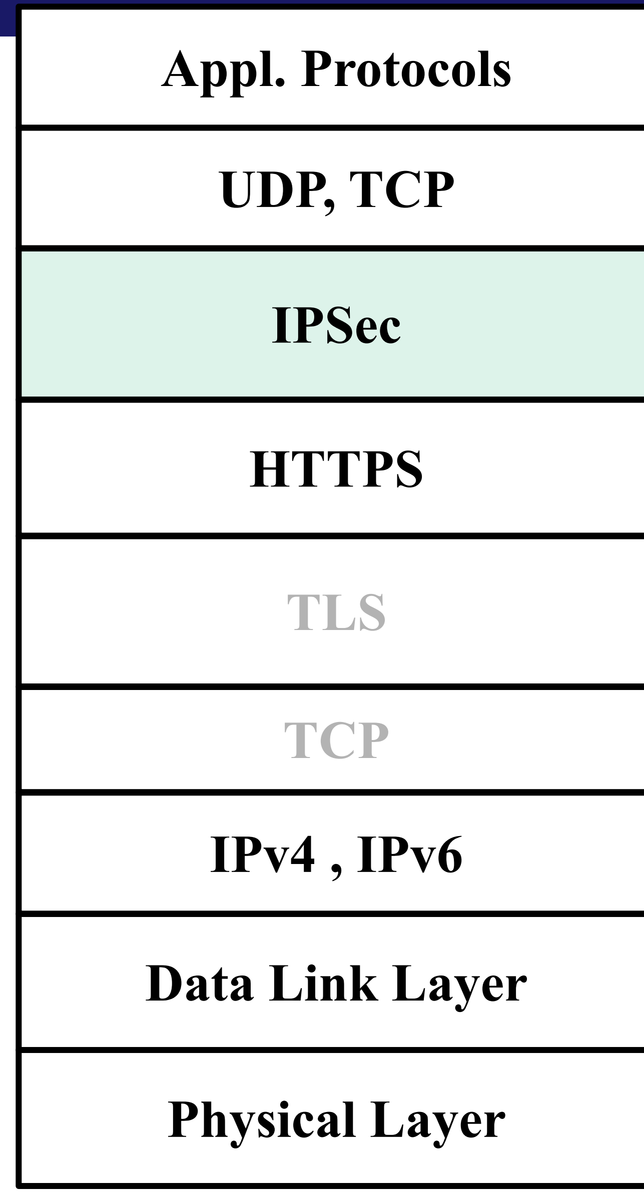


Other Possible IP Security Encapsulations

Application -Level Encapsulation

IPSec is a Security Stack of Sub Protocols for IP Traffic Protection

Ex: supported by HTTPS



Other Possible IP Security Encapsulations

Dara-Link Level Encapsulation

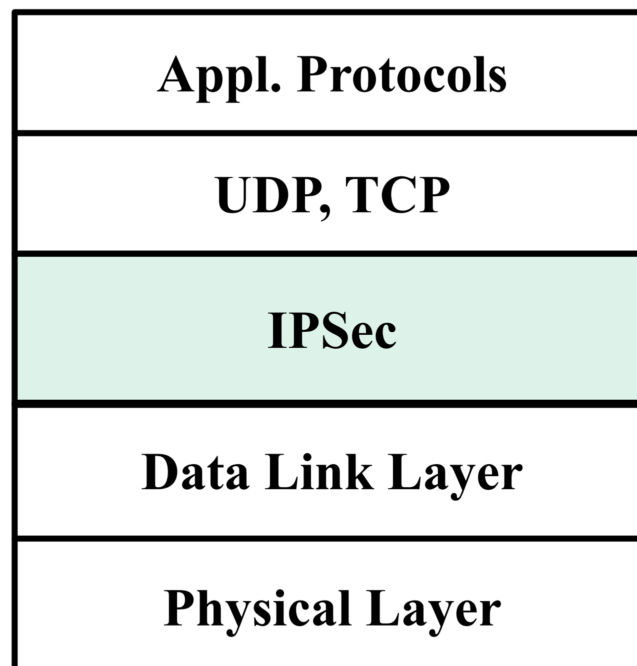
IPSec is a Security Stack of Sub Protocols for IP Traffic Protection

Ex: supported by Data Link Layer Protocols:

PPPT, P2PT or

IEE 802.1

IPSec Encapsulations



Ex: forms of encapsulation …

- Can have IPSec (ESP-E, ESP-AE or AH packets) encapsulated in other options

- Can also have IP (not necessarily IPSec) encapsulated in other stackable solutions for VPNs, ex:
  - VPN SSL/TLS
  - VPN IPSec
  - VPN PPPT
  - VPN L2PT

- Other IP Protection solutions by tunneling: STUNNEL (TLS tunnels), SSH Tunnels

Ex., Solutions (opensource):

- Stunnel https://www.stunnel.org
- OpenVPN https://openvpn.net

Ex: Secure VPN access (fct.unl.pt)

- VPN Service https://www.div-i.fct.unl.pt/servicos/vpn
- Available by using VPN Server: vpn.fct.unl.pt
  - VPN (endpoint): 193.136.124.131

See Wireshark Traces in LABs to observe VPN Traffic in Remote VPN Access to FCT/UNL (VPN endpoint)

- Use of VPN Client-Side Software:
  - Check Point Endpoint Security SW (MacOS, Windows)
- IKE/ISAKMP / UDP Handshaking for Establishment of SA and SP
- ESP Encapsulation

OBS) Can use for example Wireshark for Traffic Inspection and Analysis

Roadmap / Outline

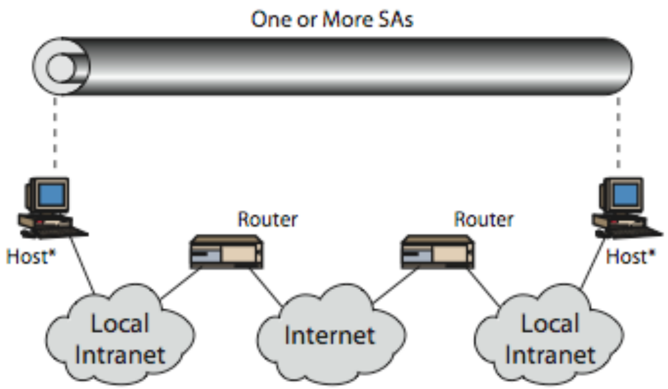
- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

Combining Security Associations

- SA's can implement either AH or ESP (not both)
- But we can implement both combining them
  - In general: we cn combine SA's for flexibility vs. security tradeoffs: This is called enforcement of **Security Association Bundles (SABs)**
- A SAB may terminate at different or same endpoints
  - Combination can be done in different ways:
    - **Transport adjacency**
    - **Iterated tunneling**
- So, SA bundling can combine authentication & encryption w/ different IPSec sub-protocols and different transport adjacency or iterated tunneling strategies
  - ESP with authentication
  - Bundled inner ESP & outer AH
  - Bundled inner transport & outer ESP

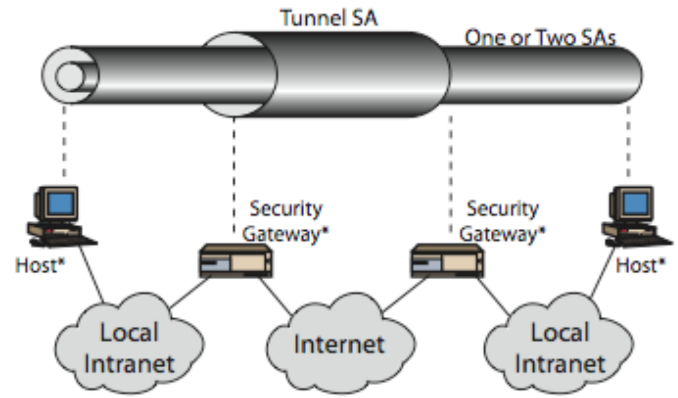
SA combinations and Bundles

(1)
2-transport SABs

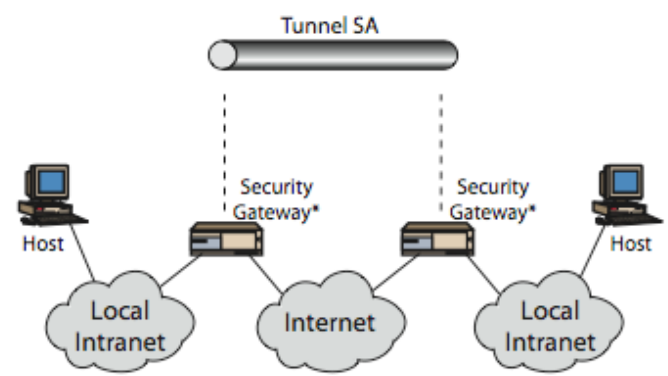


(a) Case 1

(3) 2-transport SABs and 1-tunnel SAB
End-to-End security added to (2)

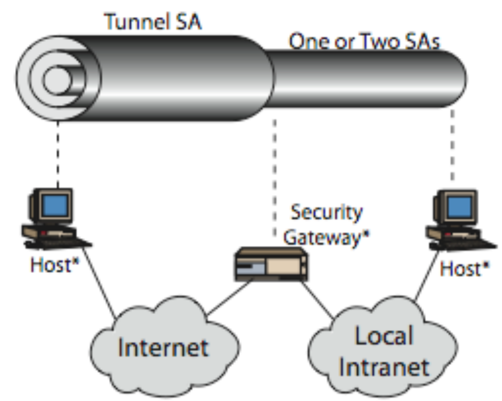


(c) Case 3



(b) Case 2

(2) 1-tunnel SAB:
ex of single tunneled VPN solution



(4) 1-2 Transport SABs and
1 Tunnel SA: A secure Remote Access

Example (wireshark traffic: ESP/AH/IP)

```
⊞ Frame 5: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface 0
⊞ Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
⊟ Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 164
    Identification: 0x0056 (86)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Authentication Header (51)
  ⊞ Header checksum: 0x217d [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
⊟ Authentication Header
    Next Header: Encap Security Payload (0x32)
    Length: 24
    AH SPI: 0xa90dc9aa
    AH Sequence: 1
    AH ICV: 157ba6cc340b1a30049ea551
⊟ Encapsulating Security Payload
    ESP SPI: 0xd2264f7a (3525726074)
    ESP Sequence: 1
```

Ex:
Manifestation of
Iterated tunneling:
ESP/AH/IP

Roadmap / Outline

- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service
  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IPSec Cryptographic Suites

- IPSec uses a variety of cryptographic algorithm types
  - RFC4308 defines VPN cryptographic suites
    - VPN-A matches common corporate VPN security using 3DES & HMAC
    - **VPN-B has stronger security for new VPNs implementing IPsecv3 and IKEv2 using AES**
  - RFC4869 updated to RFC 6379 defines four cryptographic suites compatible with US NSA specs
    - Provide choices for ESP & IKE
    - AES-GCM, AES-CBC, HMAC-SHA, ECP, ECDSA
- … Ongoing / Evolving standardization (IETF): IPSec WG

IPSec cryptosuite (summary)

As defined for VPNs (RFC 4308)

IPSec w/ IKE v1 IPSec w/ IKE v2,v3

	VPN-A	VPN-B
ESP encryption	3DES-CBC	AES-CBC (128-bit key)
ESP integrity	HMAC-SHA1-96	AES-XCBC-MAC-96
IKE encryption	3DES-CBC	AES-CBC (128-bit key)
IKE PRF	HMAC-SHA1	AES-XCBC-PRF-128
IKE Integrity	HMAC-SHA1-96	AES-XCBC-MAC-96
IKE DH group	1024-bit MODP	2048-bit MODP

As defined for VPNs NSA suite (RFC 6379)

IPSec w/ NSA Security Level Suite B

	GCM-128	GCM-256	GMAC-128	GMAC-256
ESP encryption/ Integrity	AES-GCM (128-bit key)	AES-GCM (256-bit key)	Null	Null
ESP integrity	Null	Null	AES-GMAC (128-bit key)	AES-GMAC (256-bit key)
IKE encryption	AES-CBC (128-bit key)	AES-CBC (256-bit key)	AES-CBC (128-bit key)	AES-CBC (256-bit key)
IKE PRF	HMAC-SHA-256	HMAC-SHA-384	HMAC-SHA-256	HMAC-SHA-384
IKE Integrity	HMAC-SHA-256-128	HMAC-SHA-384-192	HMAC-SHA-256-128	HMAC-SHA-384-192
IKE DH group	256-bit random ECP	384-bit random ECP	256-bit random ECP	384-bit random ECP
IKE authentication	ECDSA-256	ECDSA-384	ECDSA-256	ECDSA-384

IPSec, ECC and more recent developments

- RFC 8031 *(was draft-ietf-ipsecme-safecurves)*
  - **Curve25519 and Curve448** for the Internet Key Exchange Protocol Version 2 (IKEv2) Key Agreement

- **Curve25519: public Keys w/ 256 bits**
  - Curve25519 is intended for the ~128-bit security level, comparable to the 256-bit random ECP Groups (group 19) defined in RFC 5903, also known as NIST P-256 or secp256r1. Curve448 is intended for the ~224-bit security level.
- **Curve448: public keys w/ 448 bits**

Curve25519 and Curve448 are designed to facilitate the production of high-performance constant-time implementations. Implementers are encouraged to use a constant-time implementation of the functions. This point is of crucial importance, especially if the implementation chooses to reuse its ephemeral key pair in many key exchanges for performance reasons.

IPSec Cryptosuites (Some Improvements)

RFC 8031 *(was draft-ietf-ipsecme-safecurves)*

•Curve25519 and Curve448 for the Internet Key Exchange Protocol Version 2 (IKEv2) Key Agreement

RFC 8019 *(was draft-ietf-ipsecme-ddos-protection)*

•Protecting Internet Key Exchange Protocol Version 2 (IKEv2) Implementations from Distributed Denial-of-Service Attacks

RFC 7619 *(was draft-ietf-ipsecme-ikev2-null-auth)*

•The NULL Authentication Method in the Internet Key Exchange Protocol Version 2 (IKEv2)

RFC 7427 *(was draft-kivinen-ipsecme-signature-auth)*

•Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)

RFC 7321 *(was draft-ietf-ipsecme-esp-ah-reqts)*

•Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)

RFC 6989 *(was draft-ietf-ipsecme-dh-checks)*

•Additional Diffie-Hellman Tests for the Internet Key Exchange Protocol Version 2 (IKEv2)

Cryptosuite updates: RFCs 4308 to 7321

- ESP Authenticated Encryption (Combined Mode Algorithms)
  - SHOULD+ AES-GCM with a 16 octet ICV [RFC4106]
  - MAY AES-CCM [RFC4309]
- ESP Encryption Algorithms
  - MUST NULL [RFC2410]
  - MUST AES-CBC [RFC3602]
  - MAY AES-CTR [RFC3686]
  - MAY TripleDES-CBC [RFC2451]
  - NO DES-CBC [RFC2405]
- ESP Authentication
  - MUST HMAC-SHA1-96 [RFC2404]
  - SHOULD+ AES-GMAC with AES-128 [RFC4543]
  - SHOULD AES-XCBC-MAC-96 [RFC3566]
  - MAY NULL [RFC4303]

Authentication for IKE v2 (RFC 7427)

- Hash Algorithm
  - SHA1
  - SHA2-256
  - SHA2-384
  - SHA2-512
- Digital Signatures:
  - PKCS#1 1.5 RSA
  - SHA1, SHA2-256, SHA2-384, SHA2-512 WithRSAEncryption
  - DSA with SHA1 and SHA2-256
  - ECDSA with SHA1, SHA2-256, SHA2-384, SHA2-512
  - RSASSA-PSS
  - RSASSA-PSS and SHA-256
- Keysizes: Standardization conservative: in general, the statement recommends to be aware of "transitions" in keysizes, according to PKI management recommendations (currently >= 2048 bits)

Roadmap / Outline

- **IPSec (IP Security)**
  - IPSec overview
  - IPSec uses and benefits
  - IPSec standardization
  - IPSec architecture (and IPSec Stack)
  - IPSec: Transport vs. Tunneling Modes
  - IPSec Security Associations (SAs) and Security Policies (SPs)
  - IKE/ISKMP: establishment of SAs and SPs
  - IPSec Protocols and encapsulation
  - Anti-Replaying Service

  - Security and encapsulation flexibility
  - Combination of SAs: Security Associations
  - IPSec crypto-suites
  - More on Key Management options

IPSec Key Management

Manual key management: Sysadmin manually configures every system

- Setup (different admin facilities in different systems) for SAs establishment and SP Enforcements

Automated key management: Dynamic (on-demand) establishment of SAs and SPs

- IKEv2 emerged as the current standards for IPSec Key management protocol
- Handles key generation & distribution
  - SA establishment process

History of IKE

- Early contenders (in the IKE standardization origin):
  - Photuris: Authenticated DH with cookies & Identity Hiding
  - SKIP: Auth. DH with long-term exponents
- **ISAKMP:**
  - A protocol specifying only payload formats & exchanges (i.e., an empty protocol)
  - Adopted by the IPsec working group

Photuris and Oakley: a Modified Photuris;
  - Designed to work on ISAKMP
- **IKE: A particular (evolved) Oakley/ISAKMP combination**
- **Evolution: from IKE v.1 to IKE v2.0**

Revision: Suggested Readings and Study

Readings:

W. Stallings, Network Security Essentials – Applications and Standards
2011 Ed., (Chap.8 – IP Security)
2017 Ed. (Chap.9 – IP Security)

See: Review questions and problems (Bibliography)

Supplementary Materials: Informative References

History before IKEv2:

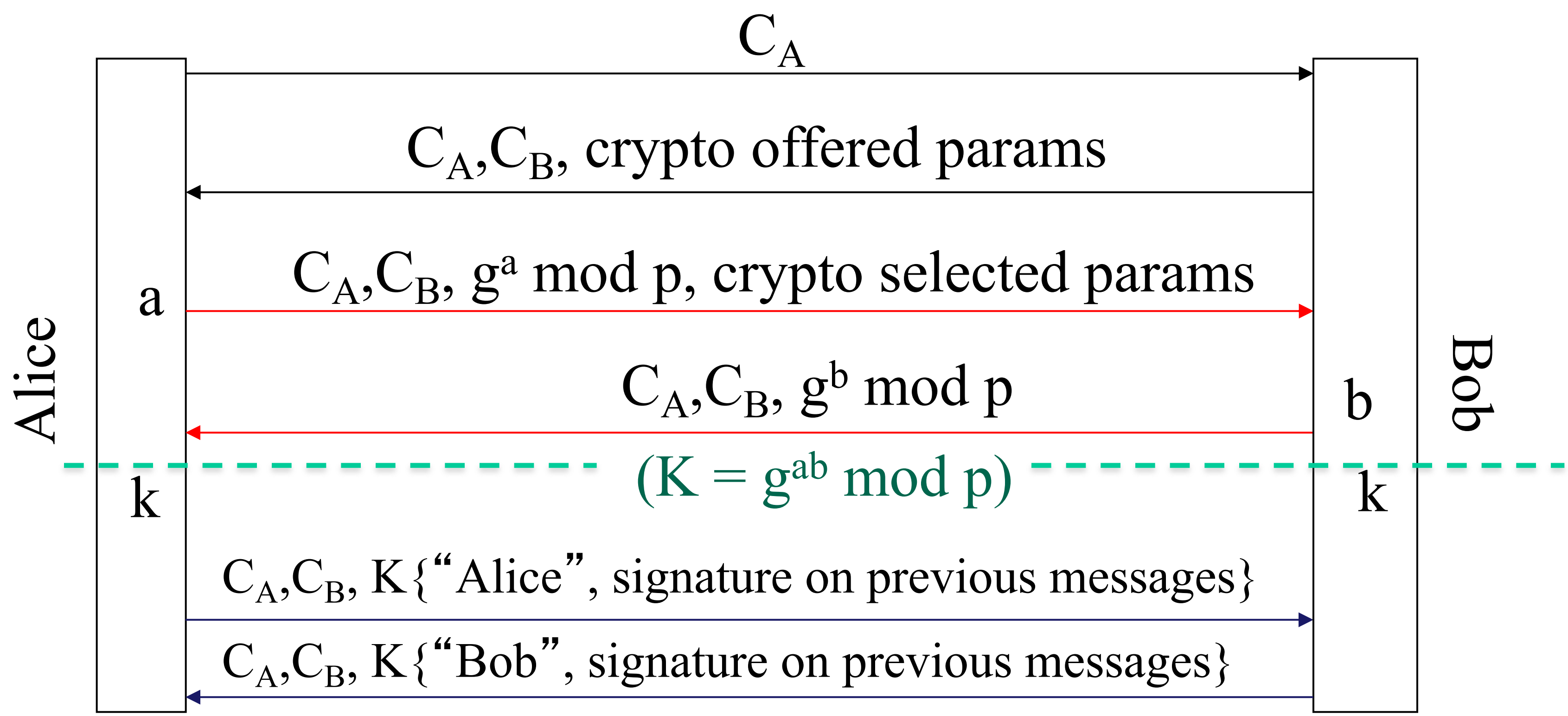
- IKEv1 and IKEv1/ISAMP
- Phitouris and Oakley Schemes

Oakley Key Exchange Protocol

- Based on Diffie-Hellman key exchange
- Adds features to address weaknesses
  - No info on parties, man-in-middle attack, cost
  - Adds cookies, groups (global params), nonces, and DH key exchange with authentication

- Can use ECC (defined curves) for ECDSA agreements

Photuris Model based on DH Key establishment



C_A: Alice's cookie; for connection ID

C_B: Bob's cookie; against DoS

Signed Agreement: ex., ECCDSA

Fast Authentication w/ HMACs

Photuris - Features

- DoS protection by cookies
  (note: C_B can be stateless)
- Authentication & integrity protection of the messages by a combined signature at the last rounds
- Identity hiding from passive attackers (How?)

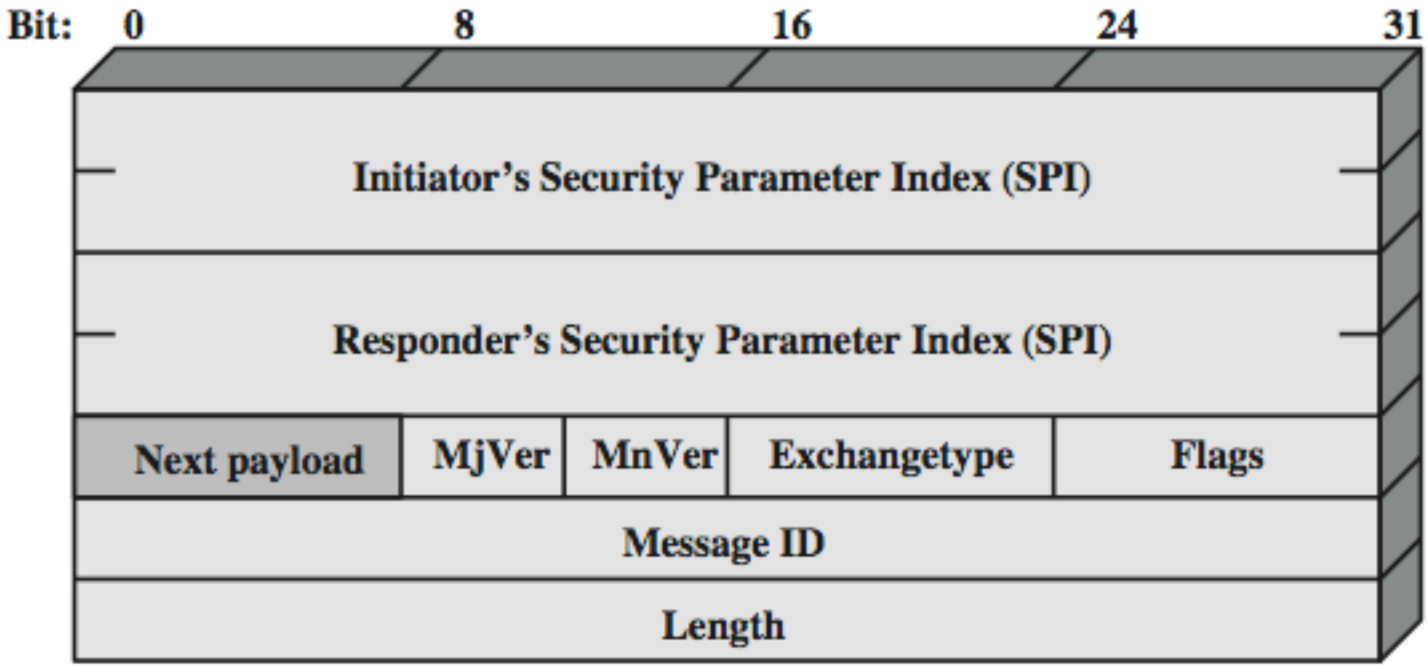
ISAKMP

Internet Security Association and Key Management Protocol

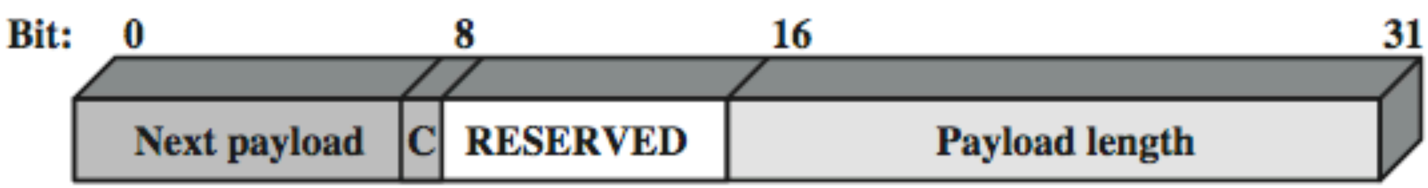
- Provides framework for key management
- Defines procedures and packet formats to establish, negotiate, modify, & delete Sas
- Independent of key exchange protocol, encryption alg, & authentication method

- **Used by IKEv1 (IKE v1/ISAKMP)**
- **IKEv2 no longer uses Oakley & ISAKMP terms … introduced simplifications and improvements … but basic functionality is same**

ISAKMP message formats



(a) IKE Header



(b) Generic Payload Header

IKE(v1) /ISAKMP

- IKE v1 is now under a smooth deprecation process…

IKE(v1) /ISAKMP : Two Phases

Phase 1:

- does authenticated DH, establishes session key & "ISAKMP SA"
- two possible modes: Main & Aggressive
- two keys are derived from the session key:
  SKEYID_e: to encrypt Phase 2 messages
  SKEYID_a: to authenticate Phase 2 messages

Phase 2:

- IPsec SA & session key established; messages encrypted & authenticated with Phase 1 keys
- Additional DH exchange is optional (for PFS)

IKE v.1: Phase 1 Exchange

Two possible modes:

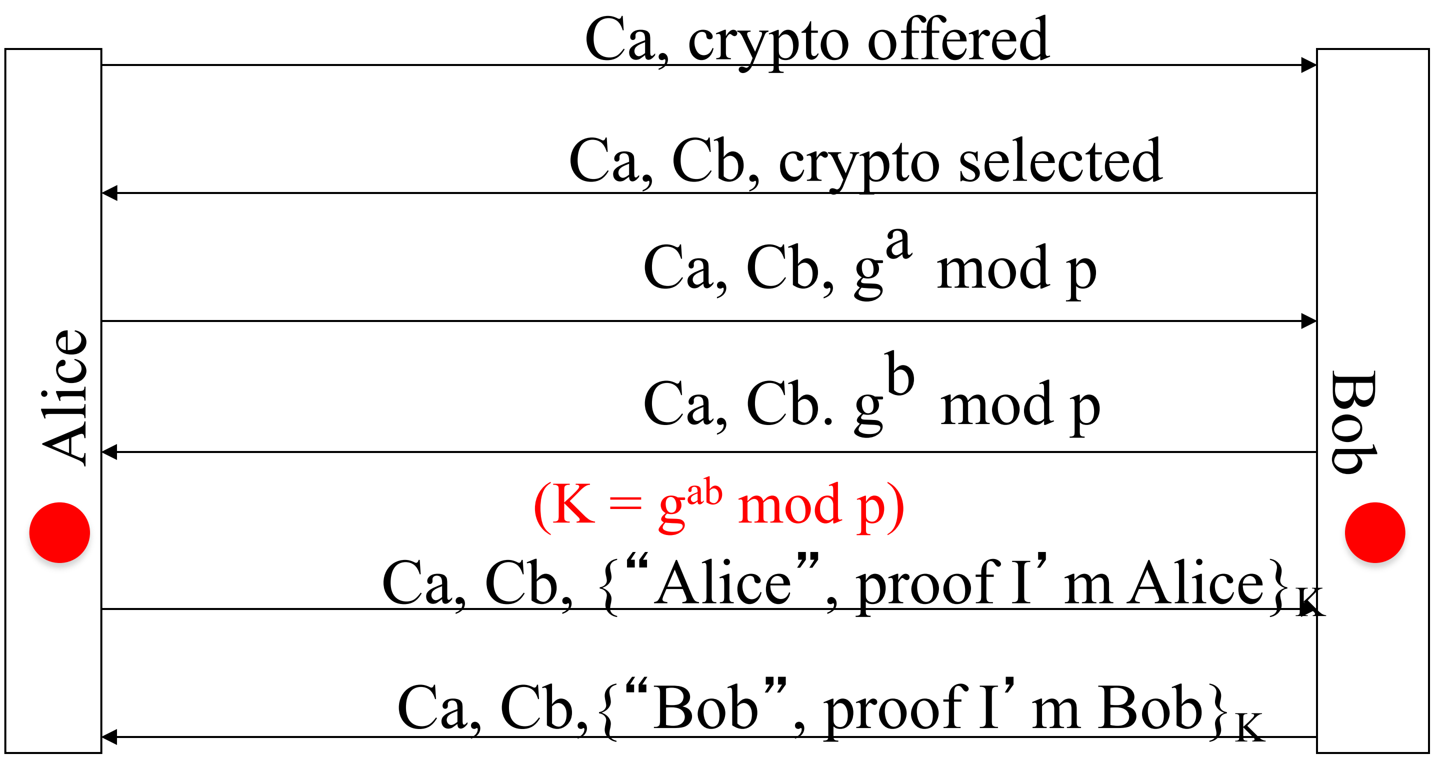
- ***Main mode:* 6 rounds; provides identity hiding**
- ***Aggressive mode:* 3 rounds**

Types of authentication:

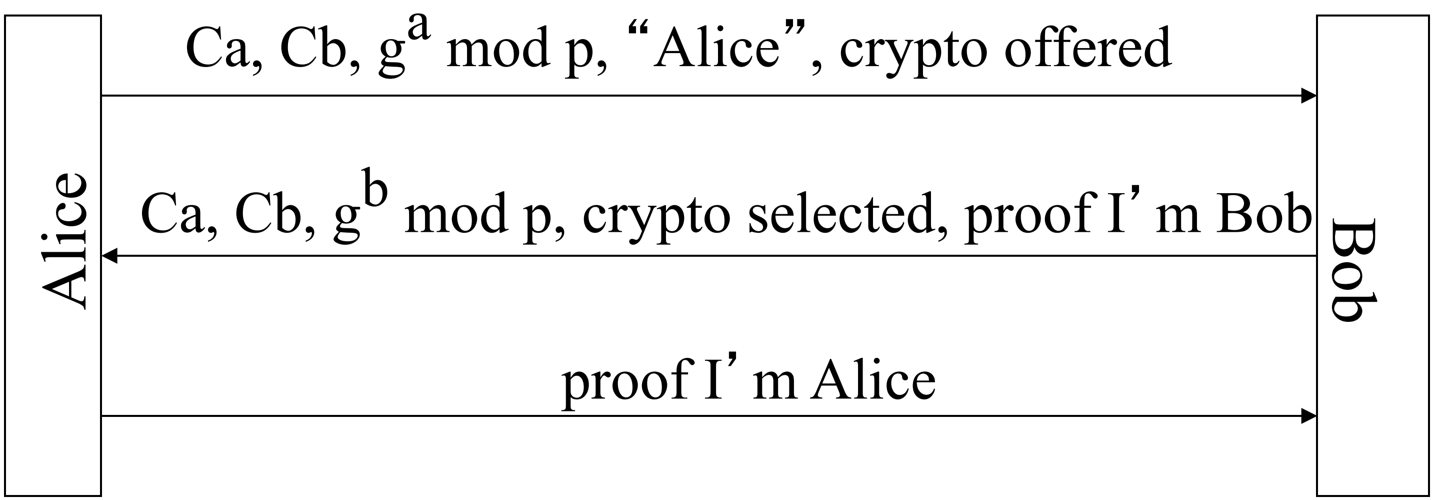
- MAC with pre-shared secret key
- digital signatures
- public key encryption
  - original: all public key encryption
  - revised: public + secret key encryption

(Each type has its benefits; but is it worth the complexity?)

IKE v.1: Phase 1 – Main Mode (generic)



IKE v.1 Phase 1 – Aggressive Mode (generic)



IKE v.1 : Phase 1 Issues & Problems

Crypto parameters:

Alice presents all algorithm combinations she can support (may be too many combinations)

Authentication:

- Certain fields (why not all?!) of the protocol messages are hashed & signed/encrypted in the final rounds
- Not included: Bob's accepted parameters (problematic)

Cookies & Statelessness:

- Cookie protection: similar to "Photuris cookies"
- Bob is no longer stateless (problematic) since "crypto offered" must be remembered from message 1.

IKE v.1: Phase 1 Issues (cont)

Session Keys:

- **2 session keys** (1 for enc. & 1 for auth.) are generated (from the initial established K).
- **So, there are 4 keys; 2 for each direction**

Complexity:

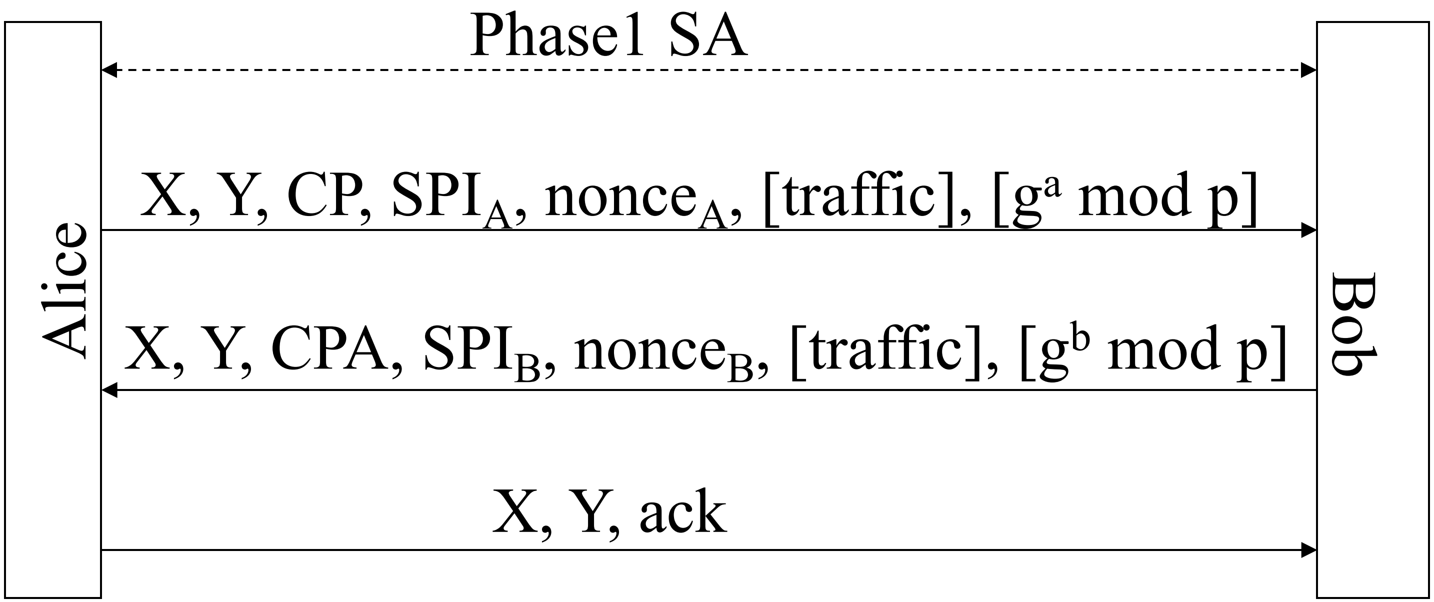
- 8 different protocols are defined
  - 2 modes
  - Each with 4 types of authentication methods

- **Regarded as unnecessarily flexible, lack of relevant issues and complex**

IKE v.1: Phase 2 Exchange

- Establishes IPsec SA & session key
- Runs over the IKE SA established in Phase 1. (message are encrypted/authenticated with Phase 1 keys)
- Key generation: based on Phase 1 key, SPI, nonces.
- DH exchange: Optional (for PFS).
- IPsec Traffic Selector: Established optionally. Specifies what traffic is acceptable. (e.g., What port numbers are allowed to use this SA.)

IKE v.1 : Phase 2



- X: pair of cookies generated in Phase 1
- Y: session identifier
- traffic: IPsec traffic selector (optional)

IKEv2 Protocol

Aims of

- Simplifying IKEv1
- Fixing some bugs (vulnerabilities)
- Fixing ambiguities
- While remaining as close to IKEv1 as possible. (… "no gratuitous changes")

IKEv2 History ... (IETF standardization roadmap)

From 1998 (First IKE)
RFC 2409

5/2005	RFC 4109, IKE v1
12/2005	RFC 4306, IKE v2 (1st version)
08/2008	RFC 5282, IKE v2 : Auth. Enc. Algorithms w/ Encrypted Payload Basically: AES w/ GCM and AES w/CCM
09/2010	RFC 5996, IKE v2 (bis, revision of 1st version)
	RFC 5998, IKE v2 (bis): update for EAP-Only Authent. Flexibility/resuse of EAP Auth. Methods and configurable options
07/2013	RFC 6989, IKE v2 (bis) w/ Additional D-H Tests (DH Imp. Validations, ECCDSA Sign.)
10/2014	RFC 7296, IKE v2 (bis, obsolets 5996)

IKEv2 History … (IETF standardization roadmap)

10/2014 RFC 7296, IKE v2 (bis, obsolets 5996)

- ECCDSA – DH Param. Redifinitions, nd integration of EAP
- Update for ambiguity issues on verifications, error handling…
- Optimizing latency: 2 round trips (4 messages)
- Rekeying schemes w/ 1 round trips (2 messages)

→ HDR, SAi1, KEi, Ni

← HDR, SAr1, KEr, Nr, [CERTREQ]

→ HDR, SK {IDi, [CERT,] [CERTREQ,] [IDr,] AUTH, SAi2, TSi, TSr}

← HDR, SK {IDr, [CERT,] AUTH, SAr2, TSi, TSr}

IKEv2 History ... (IETF standardization roadmap)

10/2014 RFC 7296, IKE v2 (bis, obsolets 5996)

-)

01/2015 RFC 7427, IKE v2 (update of 7296)

Signature Authentication and clarifications
sha1 and sha2 (256,384,512) w/RSA and PKCS#1
DSA w/ sha1 and sha256
ECDSA w/ sha1, sha256, sha384 and sha512
RSASSA-PSS

01/2016 RFC 7670 (updates 7296)

Use of other RAW PK types
(not only DER encodings, as PKCS#1)
due to the need of interpretation of the
SubjectPublicKeyInfo in X509v3 (RFC 5280)
- New Cerificate encoding formats
- Ambiguity/Lack of suport: IDs vs. related Public keys

IKEV2 Exchanges

- Different exchanges are defined for flexibility
  - Addressing security and performance tradeoffs
  - Interesting to automatic setup in different SAs, different iterated or adjacent combinations and different modes for each specific purposes

IKEv2 – Main Features

- Only one mode of authentication: Public key signatures based on X509 Certificates

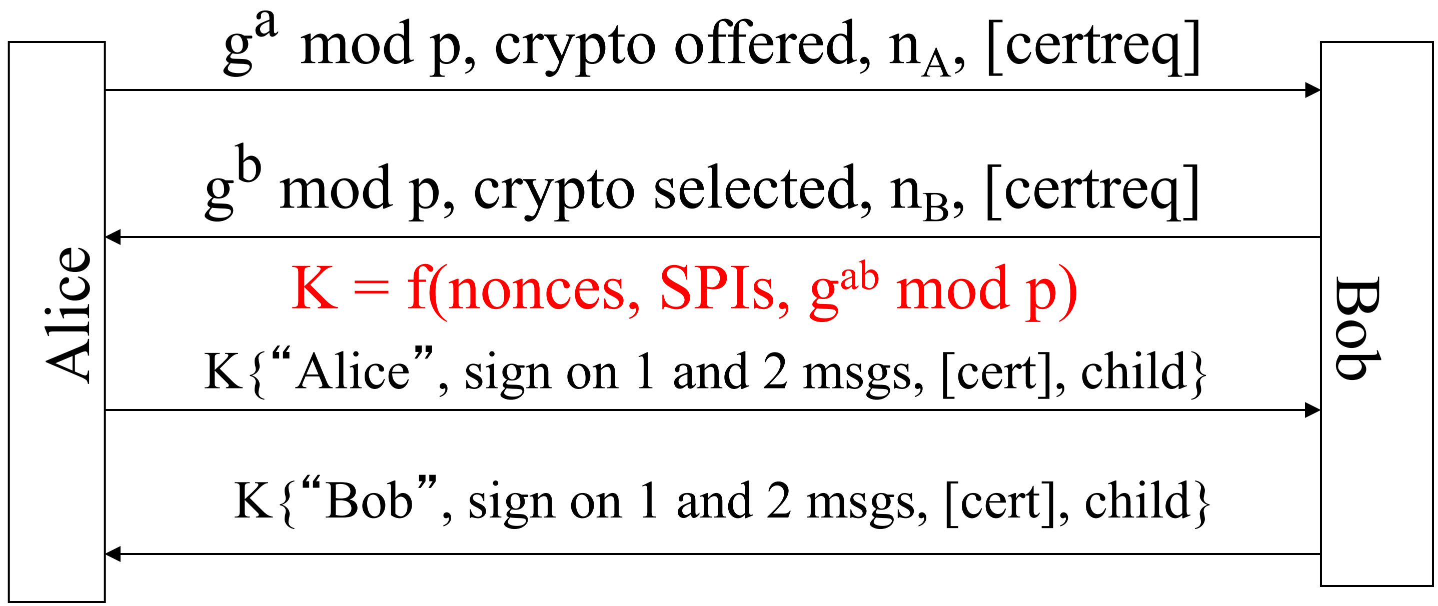
- Three possible runs
  - **Initial:** IKE SA + IPsec SA are established in the same protocol, in 4 messages. (~ Phase 1)
  - **Child-SA-Exchanges**: Additional child SAs, if needed, are established in 2 messages. (~ Phase 2)
  - **Informational Exchanges**

- DoS protection optional, via cookies (stateless).
- Crypto negotiation is simplified
  - Support for well defined / standardized “cryptosuites”
  - Ability to say “any of these enc., with any of these hash...”

IKEv2 - The Exchange Protocol (cont)

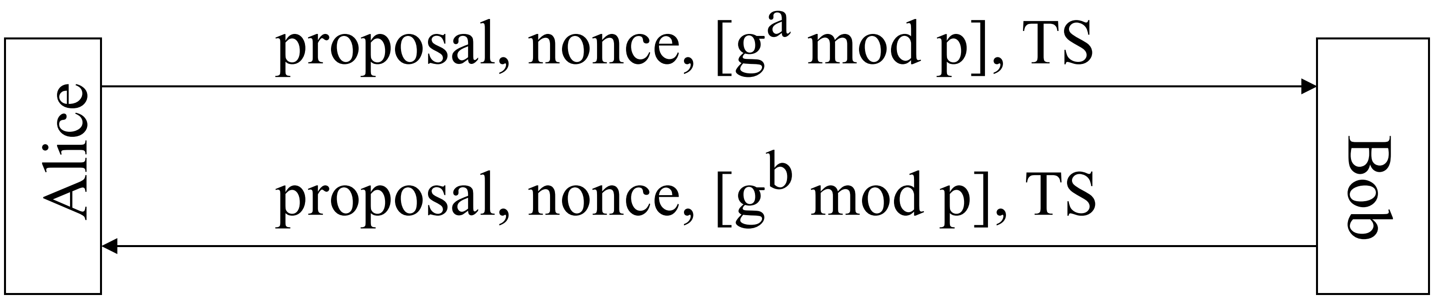
- DoS protection: Optional; by Bob responding the first message with a (stateless) cookie.
- Originally, designed with 3 rounds. Later 4 rounds is agreed on:
  - Initiator needs a 4th message anyway to know when to start the transmission.
  - Extra msgs for cookie exchange can be incorporated into 4 msgs, if Alice repeats msg.1 info in msg.3
- Preserves identity hiding from passive attackers.

IKEv2 - The base exchange protocol



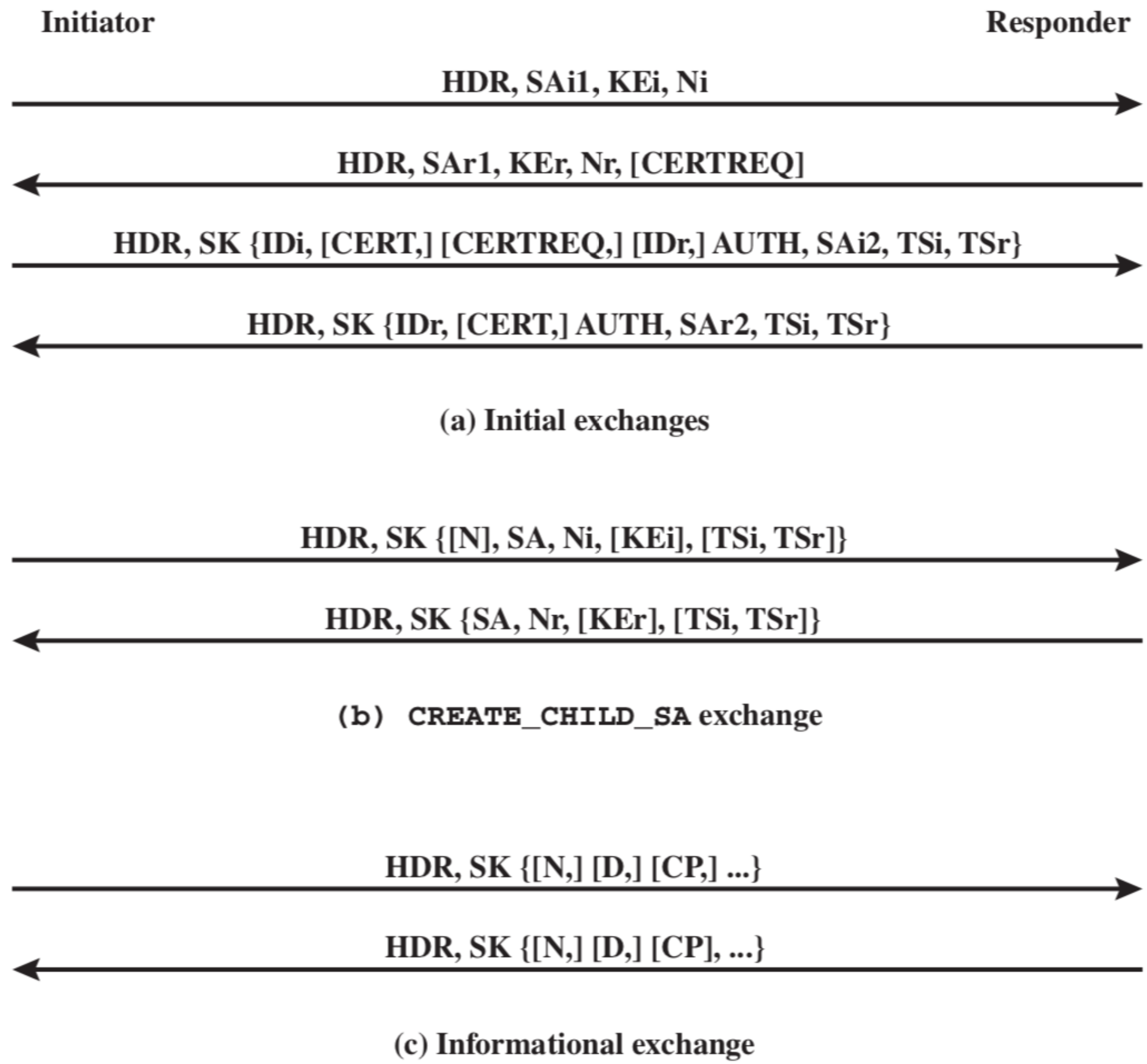
- Bob can optionally refuse the first message and require return of a cookie.
- Adds extra 2 messages.

IKEv2 - Child SA Creation



- Proposal: crypto suites, SPI, protocol (ESP, AH, IP compression)
- TS: Traffic selector
- Derived keys: Function of IKE keying material, nonces of this exchange, plus optional DH output.

IKEv2 complete exhange



HDR = IKE header
SAx1 = offered and chosen algorithms, DH group
KEx = Diffie-Hellman public key
Nx= nonces
CERTREQ = Certificate request
IDx = identity
CERT = certificate

SK {...} = MAC and encrypt
AUTH = Authentication
SAx2 = algorithms, parameters for IPsec SA
TSx = traffic selectors for IPsec SA
N = Notify
D = Delete
CP = Configuration

Other IKEv2 Features

Reliability:

- All messages are request/response.
- Initiator is responsible for retransmission if it doesn't receive a response.

Traffic selector negotiation:

- In IKEv1: Responder can just say yes/no.
- In IKEv2: Negotiation ability added.

Rekeying:

- Either side can rekey at any time.
- Rekeyed IKE-SA inherits all the child-SAs.

IKEv2 (v2): still on going discussion

- **Ex: many draft proposals in 2020 to improve IKEv2:**
  - Configurations for use with encrypted DNS
  - Group Key Management
  - Multiple (negotiated) key-exchange methods
  - Intermediated key-exchanges
  - Control and notification status on using IKEv2 v2 in IPv4 and IPv6 coexistence
  - Deprecation of crypto algorithms and definition for the use of new cryptographic algorithms
  - Use of compression or compact message formats
  - TCP encapsulation for IKE and IPSec
  - Control of maximum payload sizes

  See …

  https://datatracker.ietf.org/doc/search?name=IKE&sort=&rfcs=on&activedrafts=on

IKEv3 is now also under way: IETF Working Drafts

- Summary (motivation for IKEv3):

IKEv1	IKEv2: proposed to fix IKEv1 issues but	IKEv2 reality

Numerous issues and complexity
Too many permutations of options
• Confusing and wordy
• Hard to implement – needed lots of “bakeoffs”, or “misdefined” open implementation issues
	IKEv2 has, arguably:
	• more options than IKEv1
	• less wordy and confusing than IKEv1 but that is:
	  • “arguable”
	  • a backhanded compliment
	
	The fact is that it has gone through ~40 iterations and “clarifications”, and a few bakeoff
	
	and still …interoperability is problematic in implementations from different developers and players
		In practice, IKEv2 has growing pains from poor design choices:
		• Notify payload is now taking on negotiation responsibilities
		• ECDSA integration as been criticized as an inelegant graft;
		• ECC itself is an aterthought

What is behind the IKEv3 motivation ?

- IKEv3 as a a slimmed down key exchange for IPsec
- More simple: fewer options*:
  - D-H Group
  - A focused and defined authentication method
  - Hash algorithm, and AEAD scheme (for use in HMACs)
- Different security levels give rise to options (level --> key length, hash, D-H group, etc)
- "Only need 1 way" to skin a cat

- "Less is better", "Complexity is enemy of security and portability"
  - see also the same trends in TLS 1.3 compared with TLS 1.1 or 1.2

IKEv3 working draft concerns

- A fully-specified state machine specification!
  - Authentication method doesn't change message flow
  - Concise specification of required and expected behavior, not a collection of "colloquialisms"
  - True peer-to-peer protocol
    - Both sides can initiate at the same time
    - No initiator/responder, no client/server …. just peers

What is intended for the future IKEv3 standard ?

- Simpler, clear and easier-to-implement specification
  - Compliance to defined state machine to ensure interoperability
  - Protocol defined from view of a reference implementation, not a broad, 3rd party, description of packet flows
- Hit a functionality/complexity sweet spot
  - X% of the functionality causes Y% of the complexity (X < 20%, Y > 70%? Maybe)
  - Keep "need to have" functionality; shed "nice to have" functionality if the consequence is to cause a "spec bloat"

New in IKEv3

- One-and-done, no long-lived IKE SA
- No issues with keep-alives, no issues with deletion of IKE SAs, no delete exchanges, no state to maintain
- IKEv3 creates IPsec SAs and then goes away
- No ID protection
  - Only entities in the middle can see the IDs and those entities can launch an attack to discover an identity anyway
  - ID protection considered as a dubious value
- Attribute assertion by design, not negotiation
  – Aside from vanity there really isn't a need for numerous attributes to negotiate – it's just a key exchange!
- No point in identifying unchosen D-H groups
- Simpler: just four messages, two from each side

Differences in IKEv3 details

- Mutual authentication based on credential
  - A *secure* PSK-based method for pre-shared keys
  - Digital signatures for (certified) public keys
    - No authentication asymmetry
    - No EAP!
- Authentication is stated up front, not assumed based on presence/absence/content of payloads
- Assertions defined by attributes
  - No more Proposal/Transform/Attribute cruft
  - No more DOI/IKEv1 baggage
- No need for an encrypted payload
  - Which messages get secured is a matter of the state of the state machine.
  - How they get secured is well-defined.

Critical things in discussion

- Add critical, but missing, features
  - NAT traversal
  - Configuration (for when it really to be used in a client/server model)

- Implementation and verification of premise (well-defined state machine ensures interoperability)