Chapter 13 IPsec

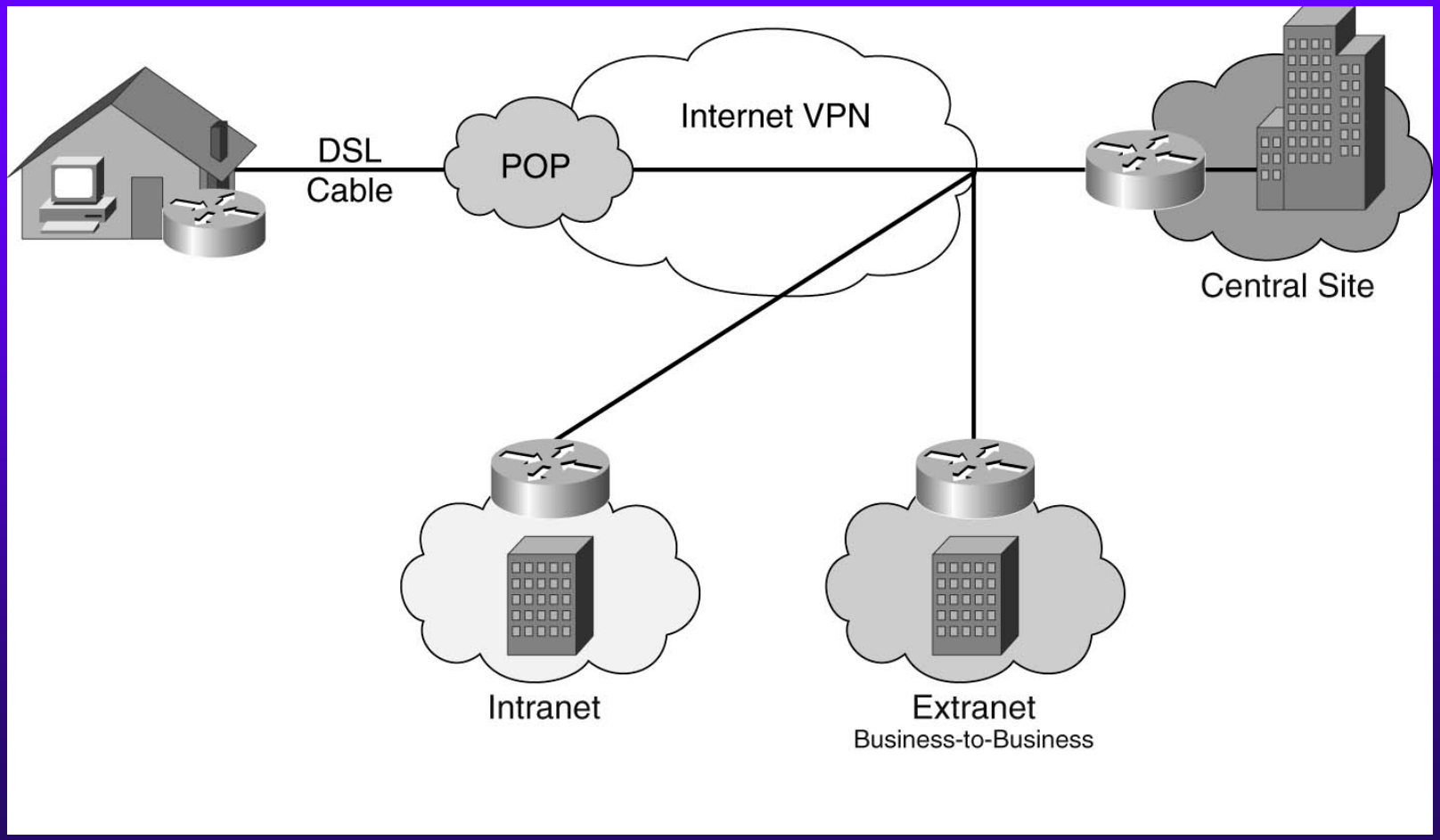
IPsec (IP Security)

- A collection of protocols used to create VPNs
- A network layer security protocol providing cryptographic security services that can support various combinations of authentication, integrity, access control, and confidentiality
- Allows creation of an encrypted tunnel between two private networks
- Supports authentication of the two ends of the tunnel
- Cannot directly encrypt non-IP traffic
- Can encrypt GRE tunnel containing non-IP data
- Comprises of IKE, ESP, and AH

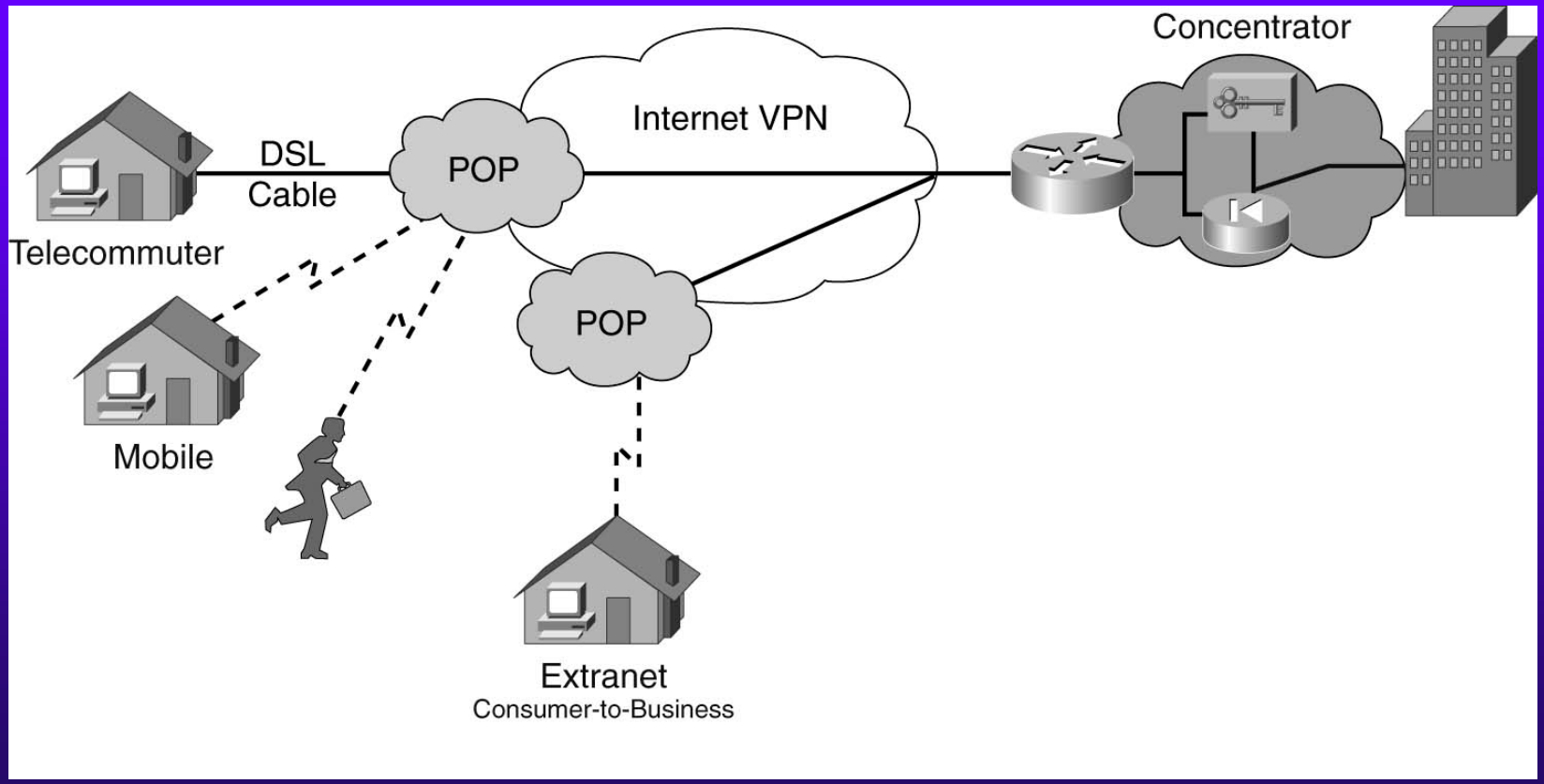
Types of IPsec VPNs

- LAN-to-LAN or site-to-site
  - Used to connect two private networks to form one combined virtual private network
- Remote-access client IPsec
  - Used to allow road warriors to be part of the trusted network

LAN-to-LAN and Site-to-Site IPsec



Remote-Access IPsec



IPsec Protocol Suite

- Internet Key Exchange (IKE) protocol
  - For negotiating security parameters and establishing authenticated keys
  - Uses UDP port 500 for ISAKMP
- Encapsulating Security Payload (ESP) protocol
  - For encrypting, authenticating, and securing data
  - IP protocol 50
- Authentication Header (AH) protocol
  - For authenticating and securing data
  - IP protocol 51

IKE's Responsibilities in IPsec Protocol

- Negotiates IPsec tunnel characteristics between two IPsec peers
- Negotiates IPsec protocol parameters
- Exchanges public keys
- Authenticates both sides
- Manages keys after the exchange
- Automates entire key-exchange process

Composition of IKE

IKE (Internet Key Exchange) (RFC 2409)
Is a Hybrid Protocol

SKEME

Mechanism for Utilizing
Public Key Encryption for Authentication

Oakley

Modes Based Mechanism for
Arriving At an Encryption Key
Between Two Peers

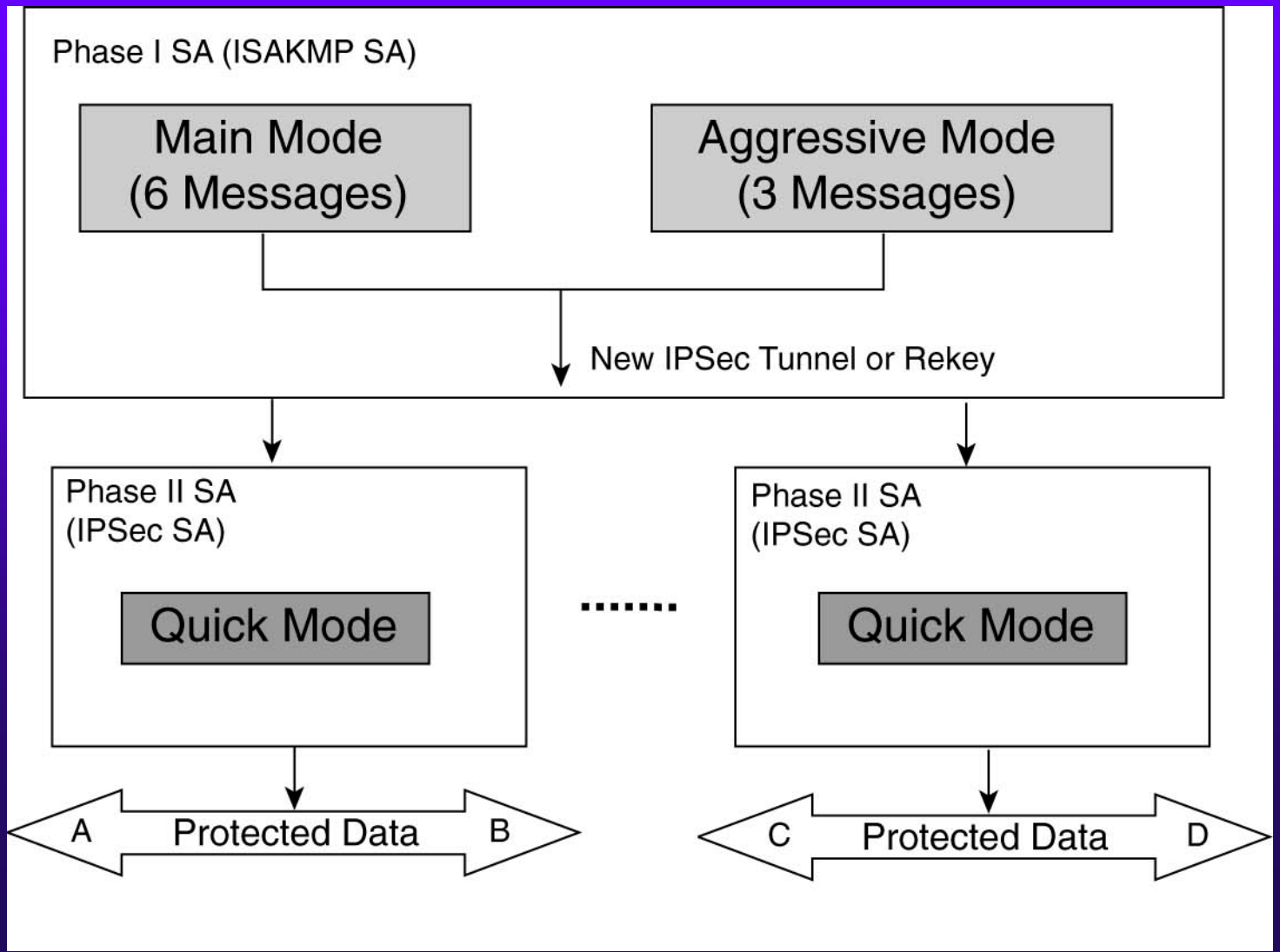
ISAKMP

Architecture for Message Exchange
Including Packet Formats
and State Transitions Between
Two Peers

IPsec Tunnel Creation using IKE

- Identify interesting traffic by an IPsec peer that has been configured to initiate an IPsec session for this traffic
- IPsec peers negotiate a secure authenticated communication channel using main mode or aggressive mode negotiation, resulting in creation of an IKE Security Association (SA) between the two IPsec peers (IKE phase I)
- Create two IPsec SAs between the two IPsec peers via IKE quick mode negotiation (IKE phase II)
- Send data over encrypted tunnel using ESP and/or AH encapsulation

IKE Main Mode, Aggressive Mode, and Quick Mode



Goals of Main Mode and Aggressive Mode

- Agreeing on a set of parameters that are to be used to authenticate the two peers
- Agreeing on parameters used to encrypt a portion of the main mode and all of the quick mode messages
- None of the aggressive mode messages are encrypted
- Authenticate the two peers to each other
- Generate keys used to generate keying material for subsequent encryption of data
- All of the parameters negotiated and the keys used to generate keys for encryption are stored as IKE or ISAKMP security association (SA)

Types of Negotiations by IKE

- Main mode using preshared key authentication followed by quick mode negotiation
- Main mode using digital signature authentication followed by quick mode negotiation
- Aggressive mode using preshared key authentication followed by quick mode negotiation
- Main mode using nonces authentication followed by quick mode negotiation
- Aggressive mode using digital signature authentication followed by quick mode negotiation

Goals of Quick Mode

- To have two peers agree on a set of attributes for creating the IPsec security associations that could be used by ESP to encrypt the data
- To redo Diffie-Hellman (DH) exchange so that new keying material can be used to generate IPsec encryption keys

IKE Main Mode Message 1 using preshared key authentication

The Initiator Proposes a Set of Attributes to Base the SA on

Initiator → Responder

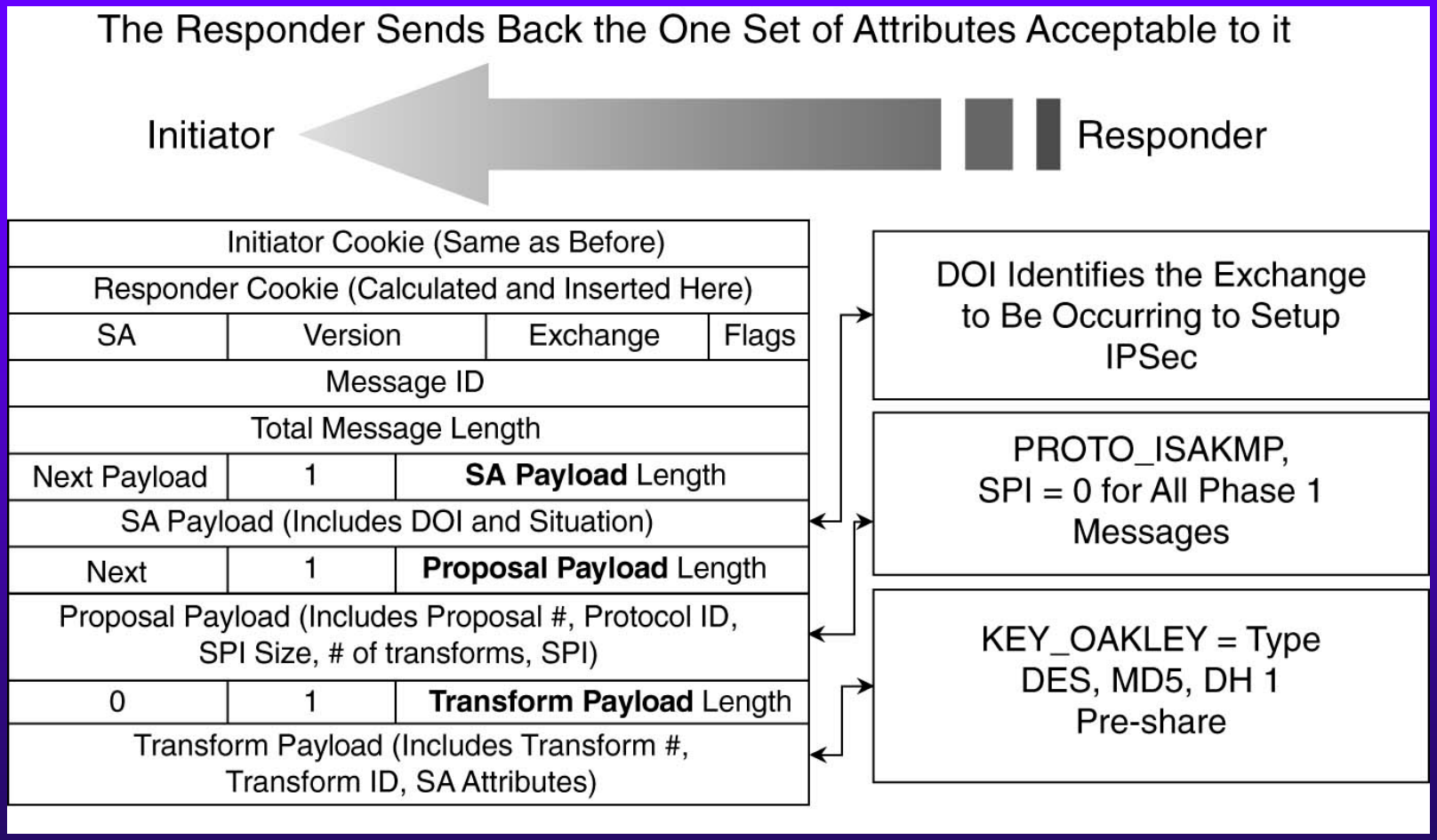
Initiator Cookie (Calculated and Inserted Here)			
Responder Cookie (Left 0 for Now)			
SA	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	1	**SA Payload** Length	
SA Payload (Includes DOI and Situation)			
Next Payload	1	**Proposal Payload** Length	
Proposal Payload			
Next Payload	1	**Transform Payload** Length	
Transform Payload			
Next Payload	1	**Proposal Payload** Length	
Proposal Payload			
0	1	**Transform Payload** Length	
Transform Payload			

DOI Identifies the Exchange to Be Occurring to Setup IPSec

SPI = 0 for All Phase 1 Messages. Includes Proposal #, Protocol ID, SPI Size, # of Transforms, SPI

Includes Transform #, Transform ID, SA Attributes. For Example, DES, MD5, DH 1, Pre-share

IKE Main Mode Message 2



Diffie-Hellman Algorithm

- Used in IKE by two peers to generate a shared DH secret and to generate keying material for later use
- DH secret also used with preshared secret to authenticate two peers to each other

Diffie-Hellman Algorithm (cont.)

- There exists X_a such that $X_a = g^a \bmod p$ where g is the generator, p is a large prime number, and a is a private secret known only to the initiator
- There exists X_b such that $X_b = g^b \bmod p$ where g is the generator, p is a large prime number, and b is a private secret known only to the responder
- Initiator and responder can generate a shared secret known only to the two of them by exchanging the values X_a and X_b with each other
- Initiator secret = $(X_b)^a \bmod p = (X_a)^b \bmod p$ = responder secret = g^{ab}

IKE Main Mode Message 3

The Initiator Sends Its DH Public Value X_a and Nonce N_i

Initiator → Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
Next Payload	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	0	**KE Payload** Length	
KE Payload (Includes DH Public Value)			
0	0	**Nonce Payload** Length	
Nonce Payload (Includes Nonce)			

DH Public Value = X_a

Nonce = N_i

IKE Main Mode Message 4

The Resonder Sends Its DH Public Value X_b and Nonce N_r

Initiator ⟵ Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
Next Payload	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	0	**KE Payload** Length	
KE Payload (Includes DH Public Value)			
0	0	**Nonce Payload** Length	
Nonce Payload (Includes Nonce)			

DH Public Value = X_b

Nonce = N_r

Session Keys Generated by the Initiator

Calculation of Three Keys (Initiator)

SKEYID_d — Used to Calculate Subsequent IPSec Keying Material
SKEYID_a — Used to Provide Data Integrity and Authentication to IKE Messages
SKEYID_e — Used to Encrypt IKE Messages

PRF = A Pseudo Random Function Based on the Negotiated Hash

SKEYID = PRF (Pre-shared Key, N_i | N_r)

SKEYID_d = PRF (SKEYID, g^{ab} | CKY-I | CKY-R | 0)

SKEYID_a = PRF (SKEYID, SKEYID_d | g^{ab} | CKY-I | CKY-R | 1)

SKEYID_e = PRF (SKEYID, SKEYID_a | g^{ab} | CKY-I | CKY-R | 2)

Session Keys Generated by the Responder

Calculation of Three Keys (Responder)

SKEYID_d — Used to Calculate Subsequent IPSec Keying Material
SKEYID_a — Used to Provide Data Integrity and Authentication to IKE Messages
SKEYID_e — Used to Encrypt IKE Messages

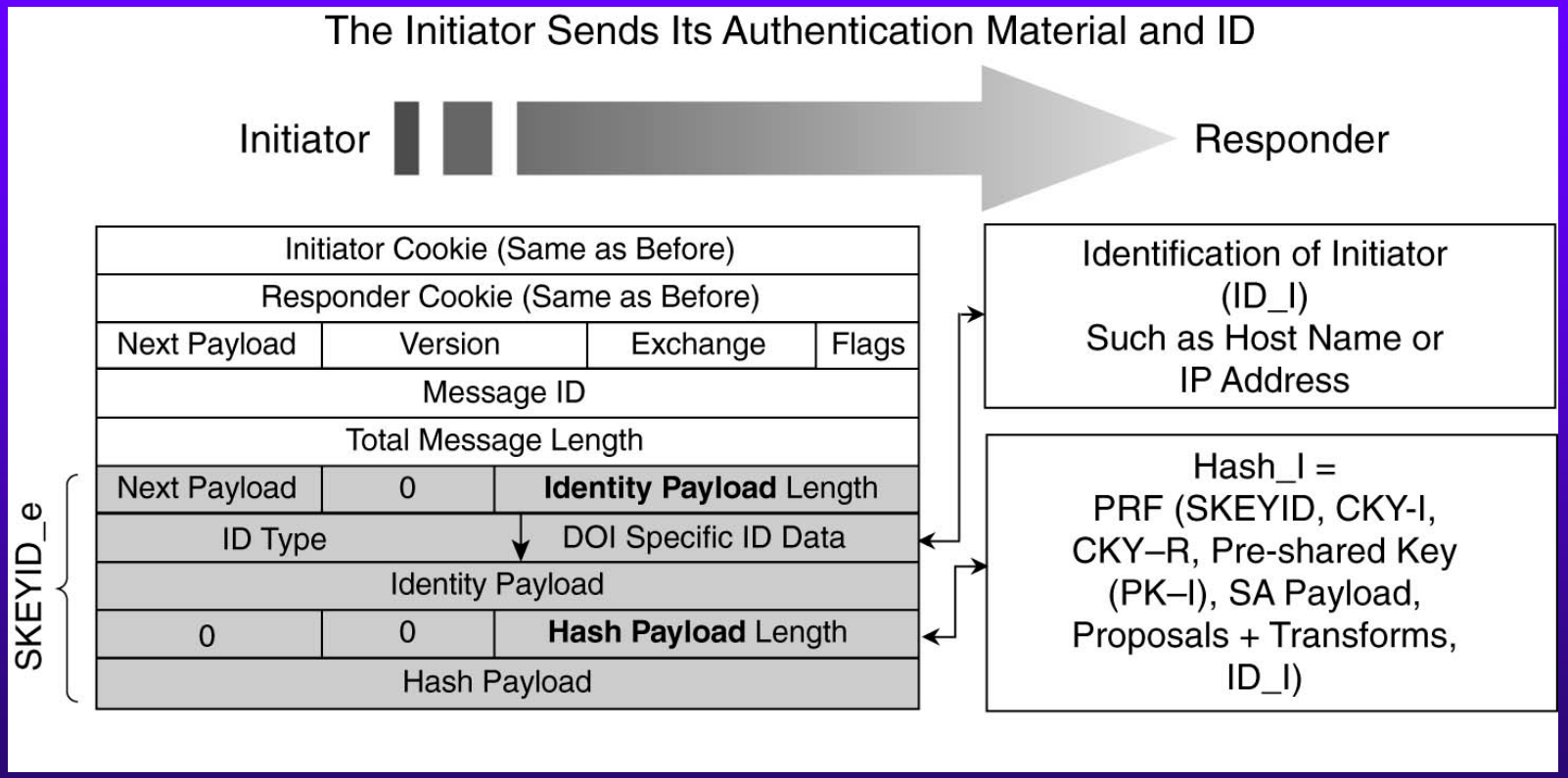
$SKEYID = PRF (Pre\text{-}shared\ Key, N_i \mid N_r)$

$SKEYID\_d = PRF (SKEYID, g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 0)$

$SKEYID\_a = PRF (SKEYID, SKEYID\_d \mid g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 1)$

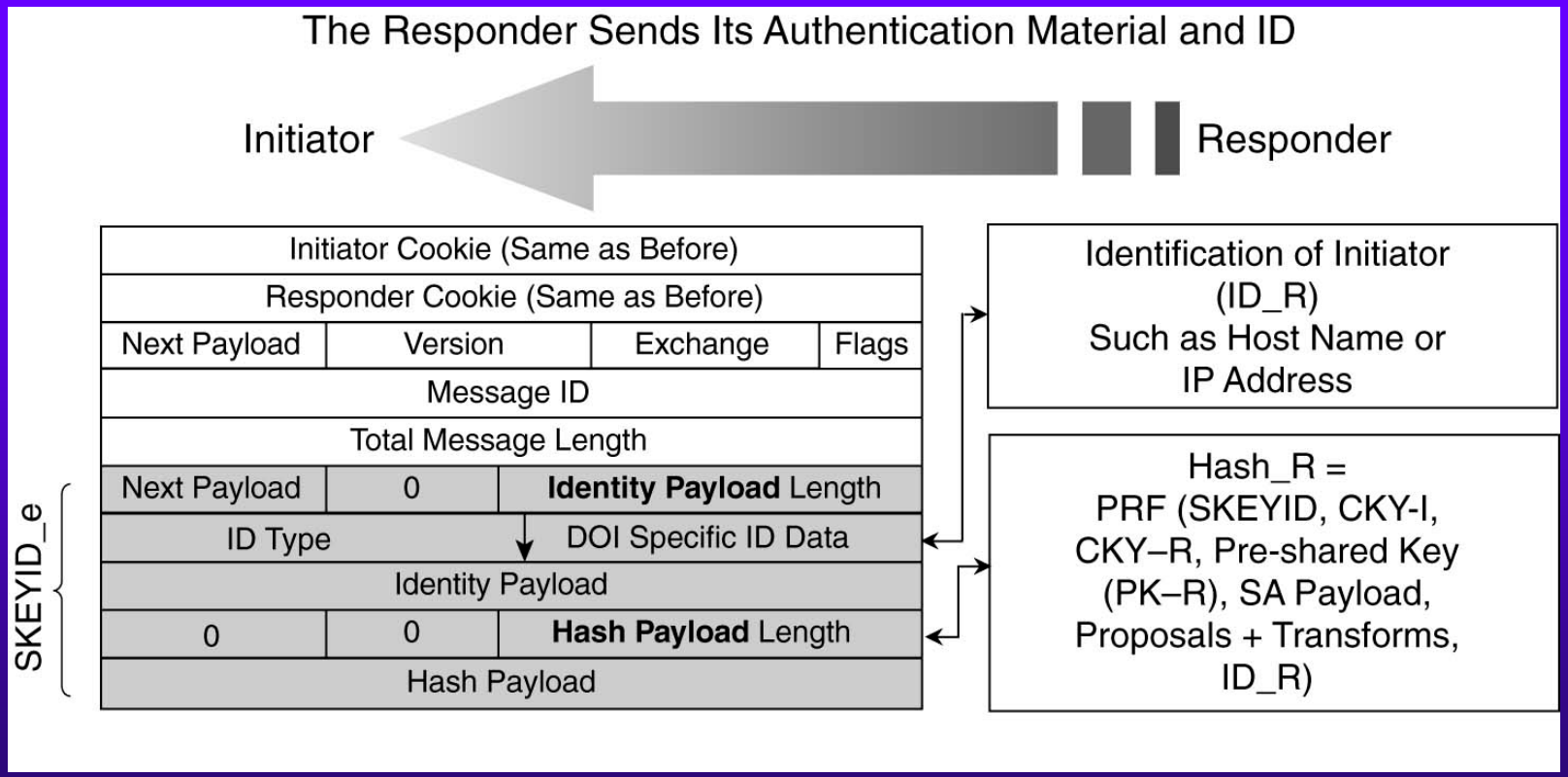
$SKEYID\_e = PRF (SKEYID, SKEYID\_a \mid g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 2)$

IKE Main Mode Message 5



-Hash payload and ID_I are used by responder to authenticate initiator
-Identity and hash payloads are encrypted using skeyid_e

IKE Main Mode Message 6



-Hash payload and ID_R are used by initiator to authenticate responder
-Identity and hash payloads are encrypted using skeyid_e

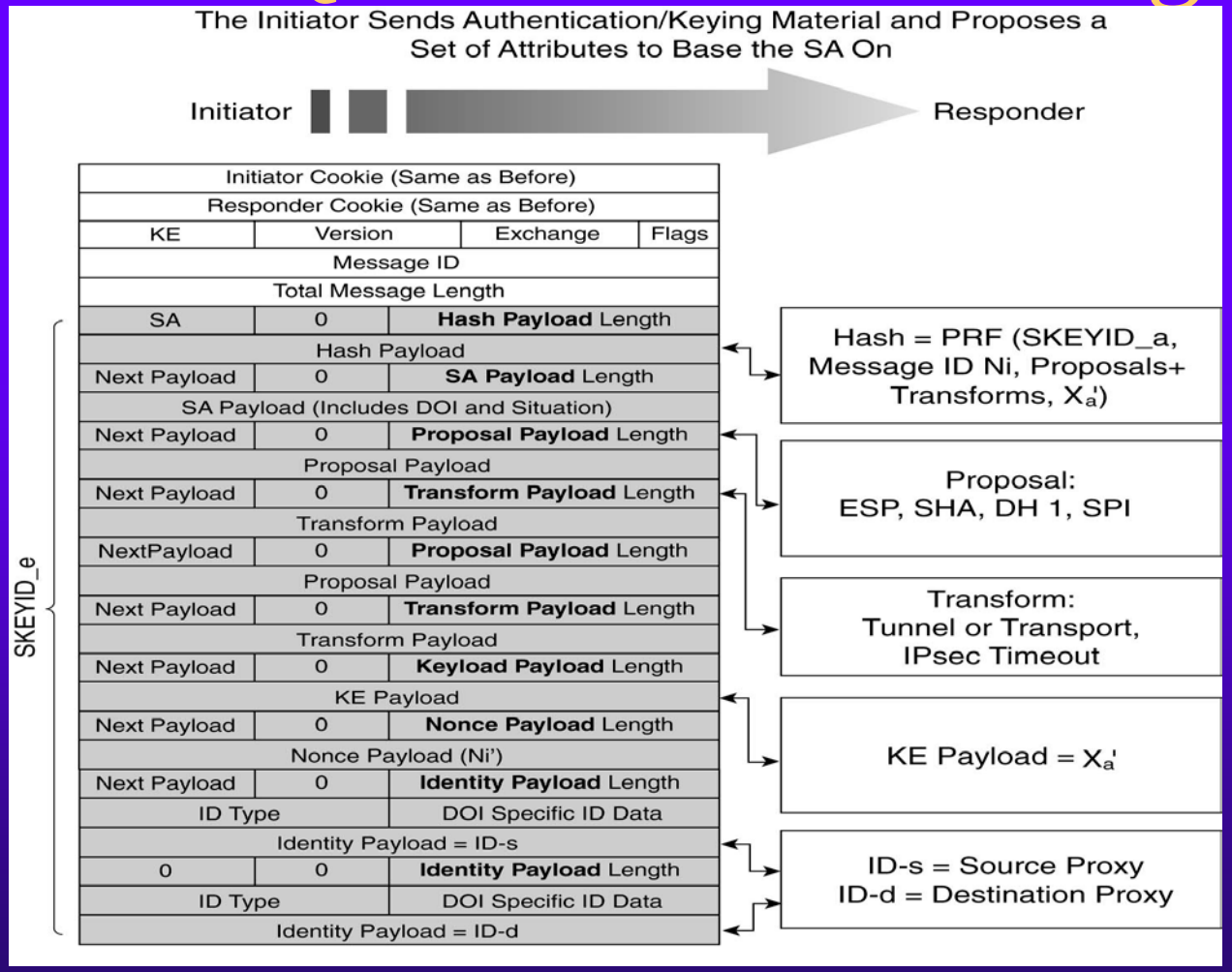
Completion of IKE Phase I (Main Mode) using Preshared Key

- IKE SA established
- Main mode using preshared key authentication completed
- Quick mode will be used to negotiate parameters of IPsec SA

IKE Phase 2 (Quick Mode)

- Negotiate parameters of IPsec SA
- Perfect Forward Secrecy (PFS) may be used by initiator to request that a new DH secret be generated over an encrypted channel
  - New nonces generated: N_i` and $N_r$`
  - New DH public values:
    - X_a`=g^a mod p
    - X_b`=g^b mod p

IKE Quick Mode Message 1

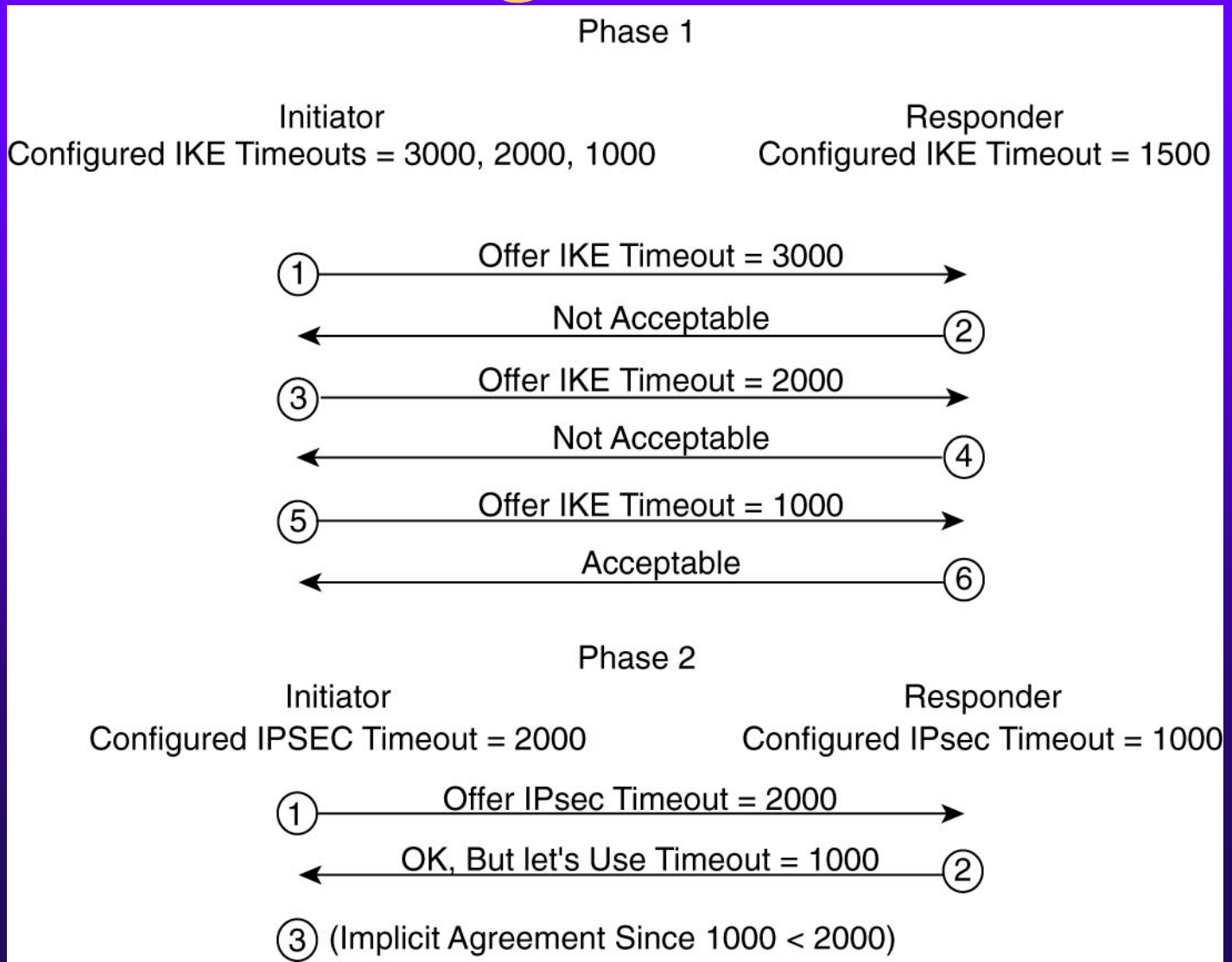


-Hash used for reauthentication

-proposal and transform suggests ESP or AH encapsulation type, SHA or MD5 integrity checking, DH group, and tunnel or transport mode

- Key exchange payload used for generating new DH secret

IKE and IPsec Lifetime Negotiation



IKE Quick Mode Message 2

The ResponderSends Authentication/Keying Material and Accepts a Set of Attributes to Base the SA On

Initiator ← Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
KE	Version	Exchange	Flags
Message ID			
Total Message Length			
SA	0	**Hash Payload** Length	
Hash Payload			
Proposal	0	**SA Payload** Length	
SA Payload (Includes DOI and Situation)			
Transform	0	**Transform Payload** Length	
Proposal Payload (Accepted Proposal)			
Proposal	0	**Proposal Payload** Length	
Transform Payload (Accepted Transform)			
Nonce	0	**KE Payload** Length	
KE Payload			
ID	0	**Nonce Payload** Length	
Nonce Payload (Nr')			
ID	0	**Identity Payload** Length	
ID Type		DOI Specific ID Data	
Identity Payload = ID-s (Generally similar to ID-d for intiator)			
0	0	**Identity Payload** Length	
ID Type		DOI Specific ID Data	
Identity Payload = ID-d (Generally similar to ID-s for initiator)			

SKEYID_e

Hash = PRF (SKEYID_a, Message ID Ni', Nr', Accepted Proposal + Transform, X_b'

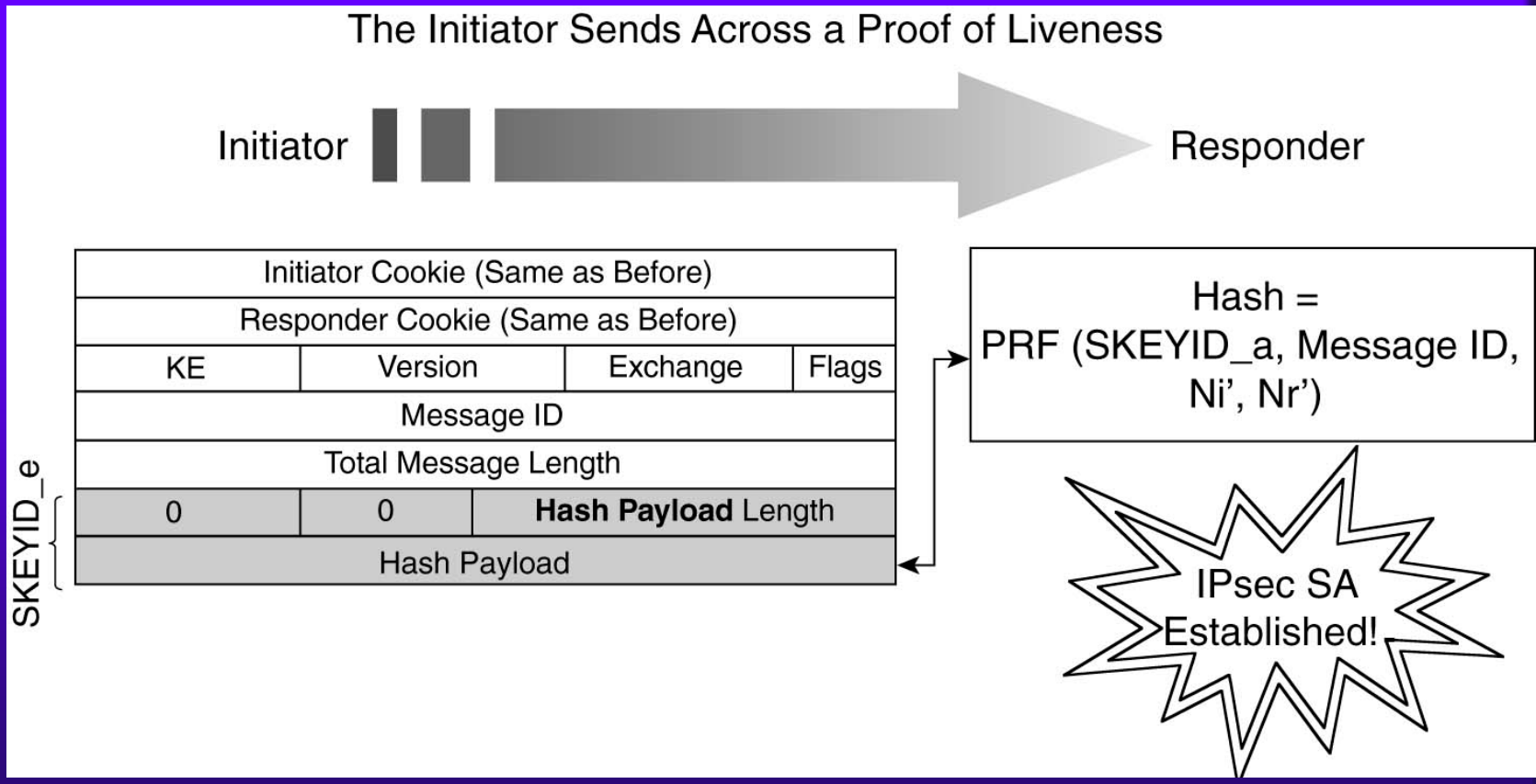
Proposal: With Responder's SPI

KE Payload = X_b'

Generation of IPsec Keying Material

- Both peers generate new DH shared secret = $(X_b`)^a \bmod p = (X_a`)^b \bmod p$
- Both peers generate shared session keys for incoming and outgoing IPsec SAs based on SKEYID_d, new DH shared secret, SPI, and $N_i`$ and $N_r`$

IKE Quick Mode Message 3



Main Mode Using Digital Signature Authentication followed by Quick Mode Negotiation

Session Keys Generated by the Initiator in Digital Signatures Method of Main Mode Negotiation

Calculation of Three Keys (Initiator)

SKEYID_d — Used to Calculate Subsequent IPSec Keying Material
SKEYID_a — Used to Provide Data Integrity and Authentication to IKE Messages
SKEYID_e — Used to Encrypt IKE Messages

PRF = A Pseudo Random Function Based on the Negotiated Hash

SKEYID = PRF (Pre-shared Key, N_i | N_r)

SKEYID_d = PRF (SKEYID, g^{ab} | CKY-I | CKY-R | 0)

SKEYID_a = PRF (SKEYID, SKEYID_d | g^{ab} | CKY-I | CKY-R | 1)

SKEYID_e = PRF (SKEYID, SKEYID_a | g^{ab} | CKY-I | CKY-R | 2)

Session Keys Generated by the Responder in Digital Signatures Method of Main Mode Negotiation

Calculation of Three Keys (Responder)

SKEYID_d — Used to Calculate Subsequent IPSec Keying Material
SKEYID_a — Used to Provide Data Integrity and Authentication to IKE Messages
SKEYID_e — Used to Encrypt IKE Messages

$$SKEYID = PRF\ (N_i \mid N_r \mid g^{ab})$$

$$SKEYID\_d = PRF\ (SKEYID, g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 0)$$

$$SKEYID\_a = PRF\ (SKEYID, SKEYID\_d \mid g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 1)$$

$$SKEYID\_e = PRF\ (SKEYID, SKEYID\_a \mid g^{ab} \mid CKY\text{-}I \mid CKY\text{-}R \mid 2)$$

IKE Main Mode Message 5 (using Digital Signatures)

The Initiator Sends Its Authentication Material and ID

Initiator → Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
KE	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	0	**Identity Payload** Length	
ID Type		DOI Specific ID Data	
Identity Payload			
Next Payload	0	**Signature Payload** Length	
Signature Data			
0	0	**Certificate Payload** Length	
Certificate Encoding	Certificate Data		
Certificate Data			

Identification of Responder (ID_I) Such as Host Name or IP Address

Signature = Hash_I encrypted with Priv_I = Priv_I {PRF (SKEYID, CKY–I, CKY–R, SA Payload, Proposals + Transforms, ID_I)}

IKE Main Mode Message 6 (using Digital Signatures)

The Responder Sends Its Authentication Material and ID

Initiator ← Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
KE	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	0	**Identity Payload** Length	
ID Type		DOI Specific ID Data	
Identity Payload			
Next Payload	0	**Signature Payload** Length	
Signature Data			
0	0	**Certificate Payload** Length	
Certificate Encoding	Certificate Data		
Certificate Data			

Identification of Responder (ID_R) Such as Host Name or IP Address

Signature = Hash_I encrypted with Priv_R = Priv_R {PRF (SKEYID, CKY–I, CKY–R, SA Payload, Proposals + Transforms, ID_R)}

Aggressive Mode of IKE Phase 1 using Preshared Key Authentication

IKE Aggressive Mode Message 1

The Initiator Proposes a Set of Attributes, ID, Nonce and DH Public Value

Initiator → Responder

Initiator Cookie (Same as Before)			
Responder Cookie (Same as Before)			
SA	Version	Exchange	Flags
Message ID			
Total Message Length			
Next Payload	1	**SA Payload** Length	
SA Payload (includes DOI and Situation			
Next Payload	1	**Proposal Payload** Length	
Proposal Payload			
Next Payload	1	**Transform Payload** Length	
Transform Payload			
Next Payload	1	**Proposal Payload** Length	
Proposal Payload			
Next Payload	1	**Transform Payload** Length	
Transform Payload			
Next Payload	0	**KE Payload** Length	
KE Payload (Includes DH Public Value)			
Next Payload	0	**Nonce Payload** Length	
Nonce Payload (Includes Nonce)			
0	0	**Identity Payload** Length	
ID Type		DOI Specific ID Data	
Identity Payload			

DOI Identifies the Exchange to be Occuring to Setup IPSec

SPI = 0 for All Phase 1 Messages, Includes Proposal #, Protocol ID, SPI Size, # of Transforms, SPI

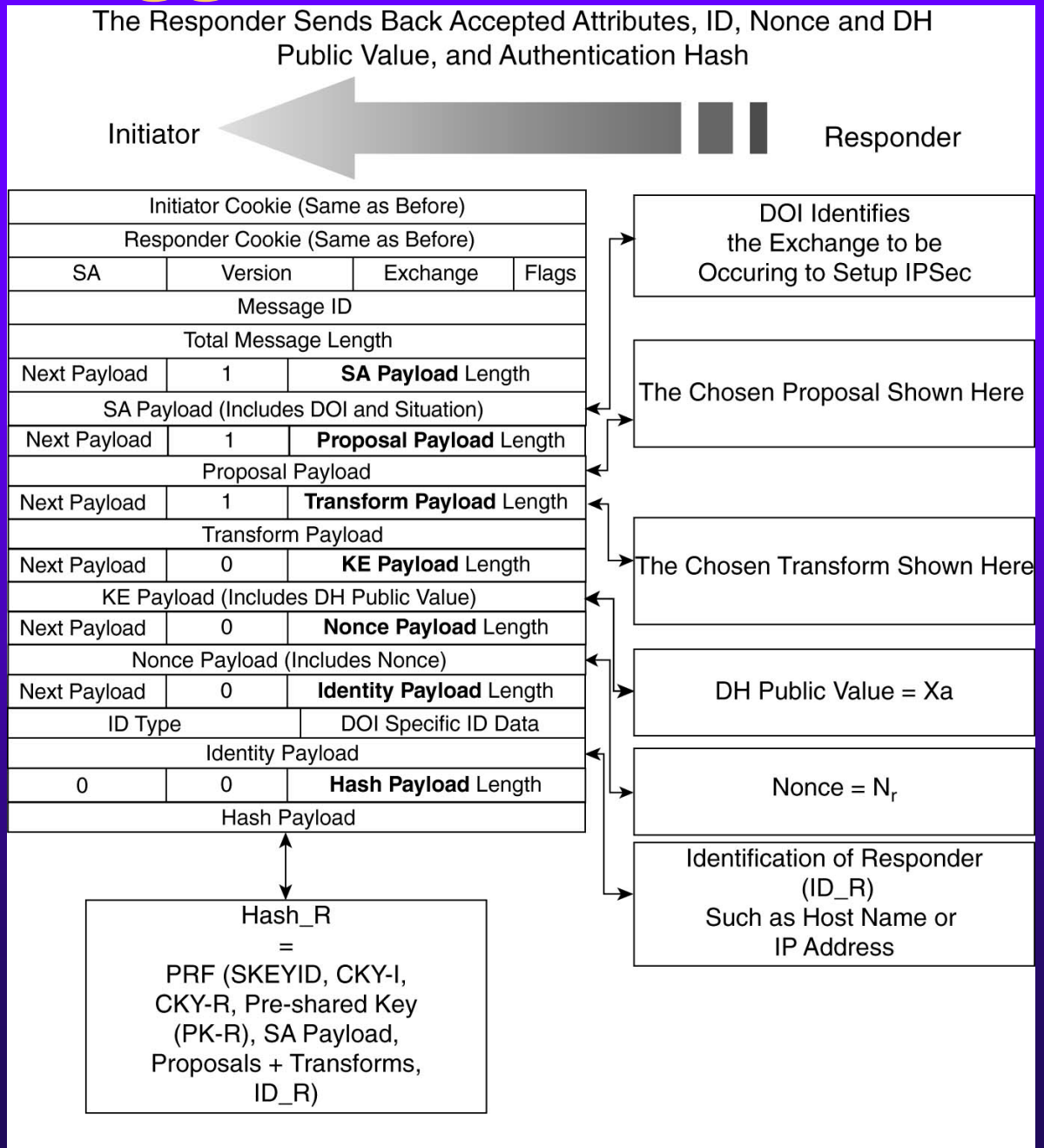
Includes Transform #, Transform ID, SA Attributes, For Example, DES, MD5, DH 1, Pre-share

DH Public Value = Xa

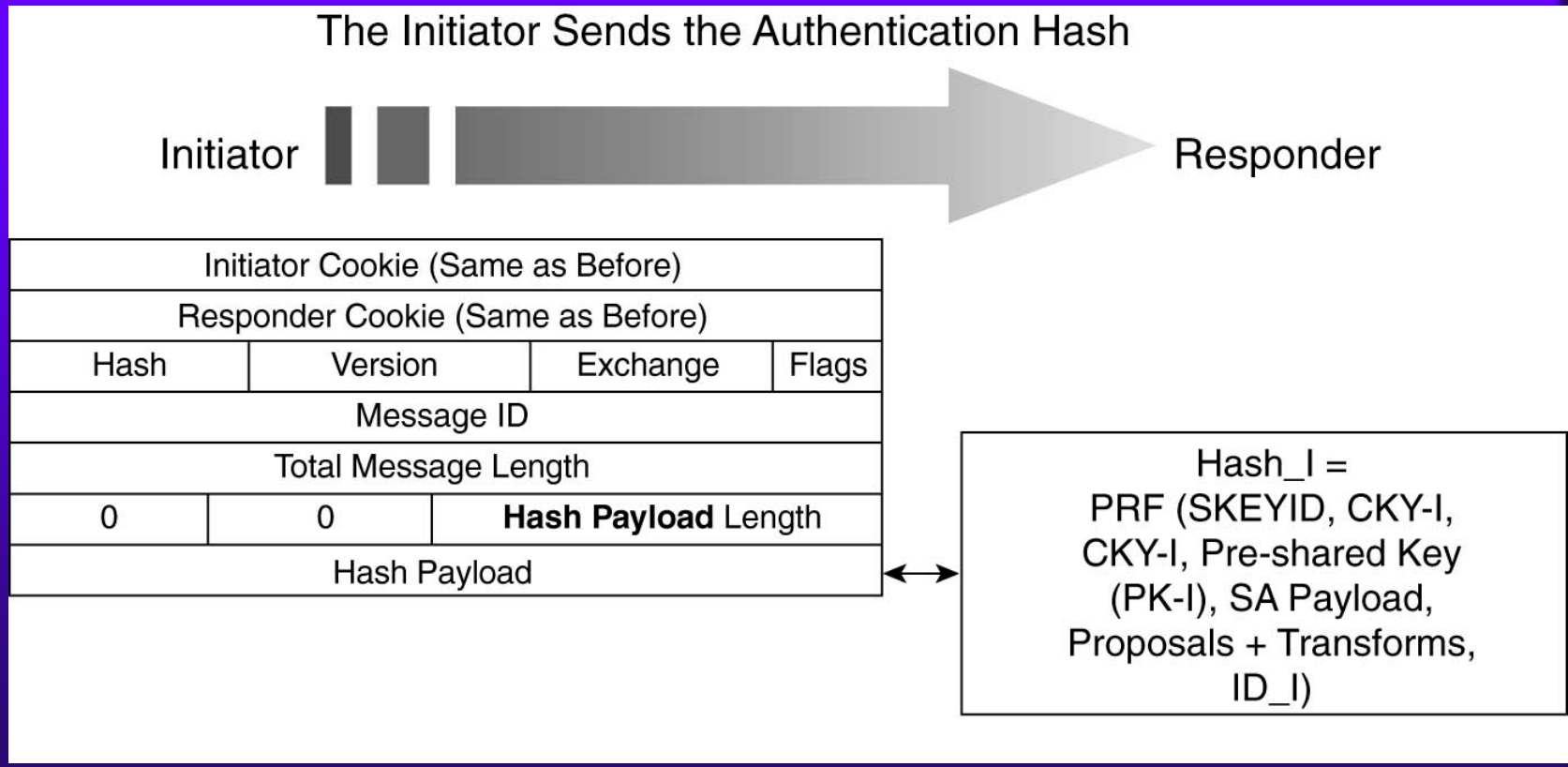
Nonce = N_i

Identification of Responder (ID_I) Such as Host Name or IP Address

IKE Aggressive Mode Message 2



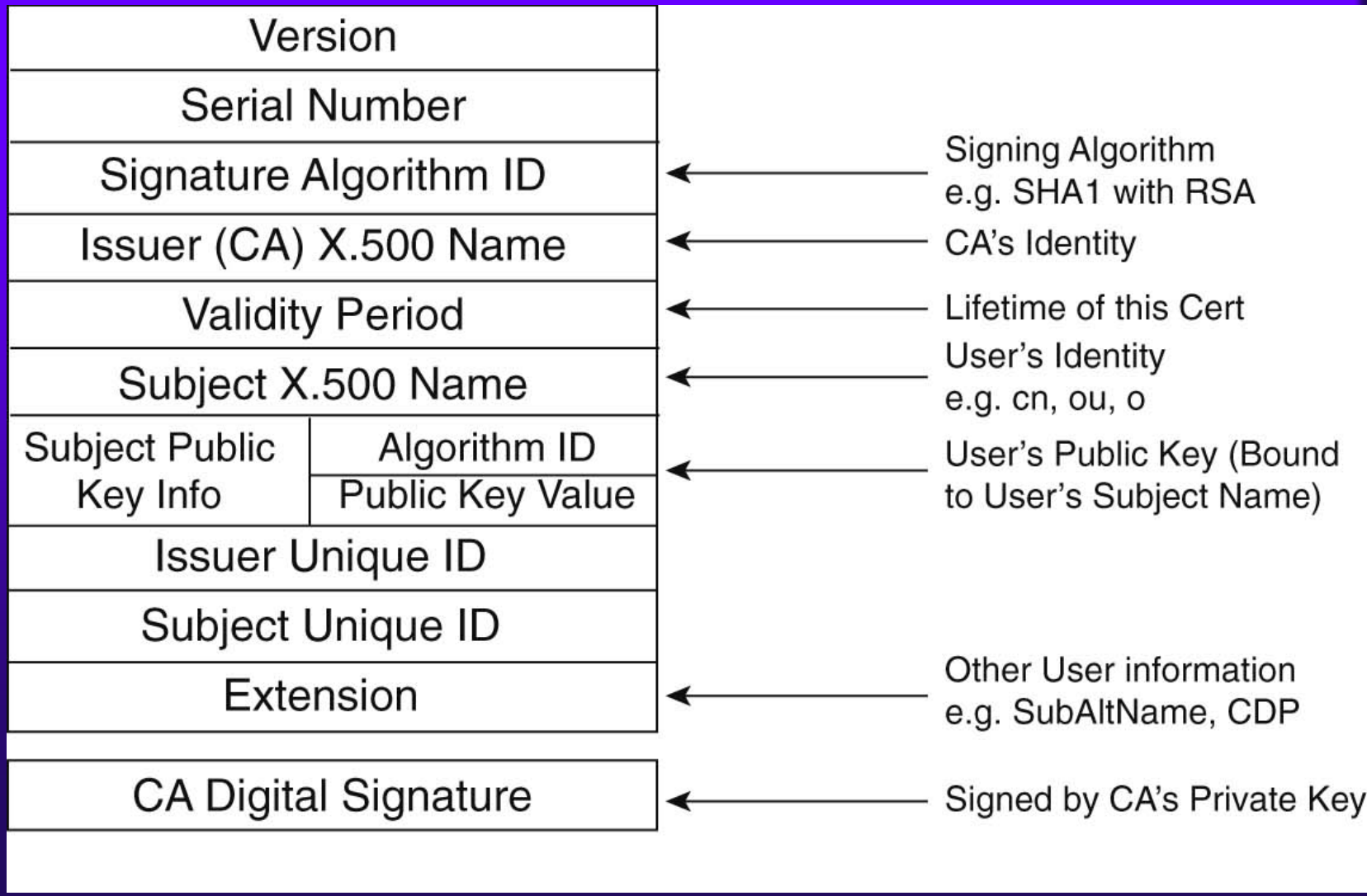
IKE Aggressive Mode Message 3



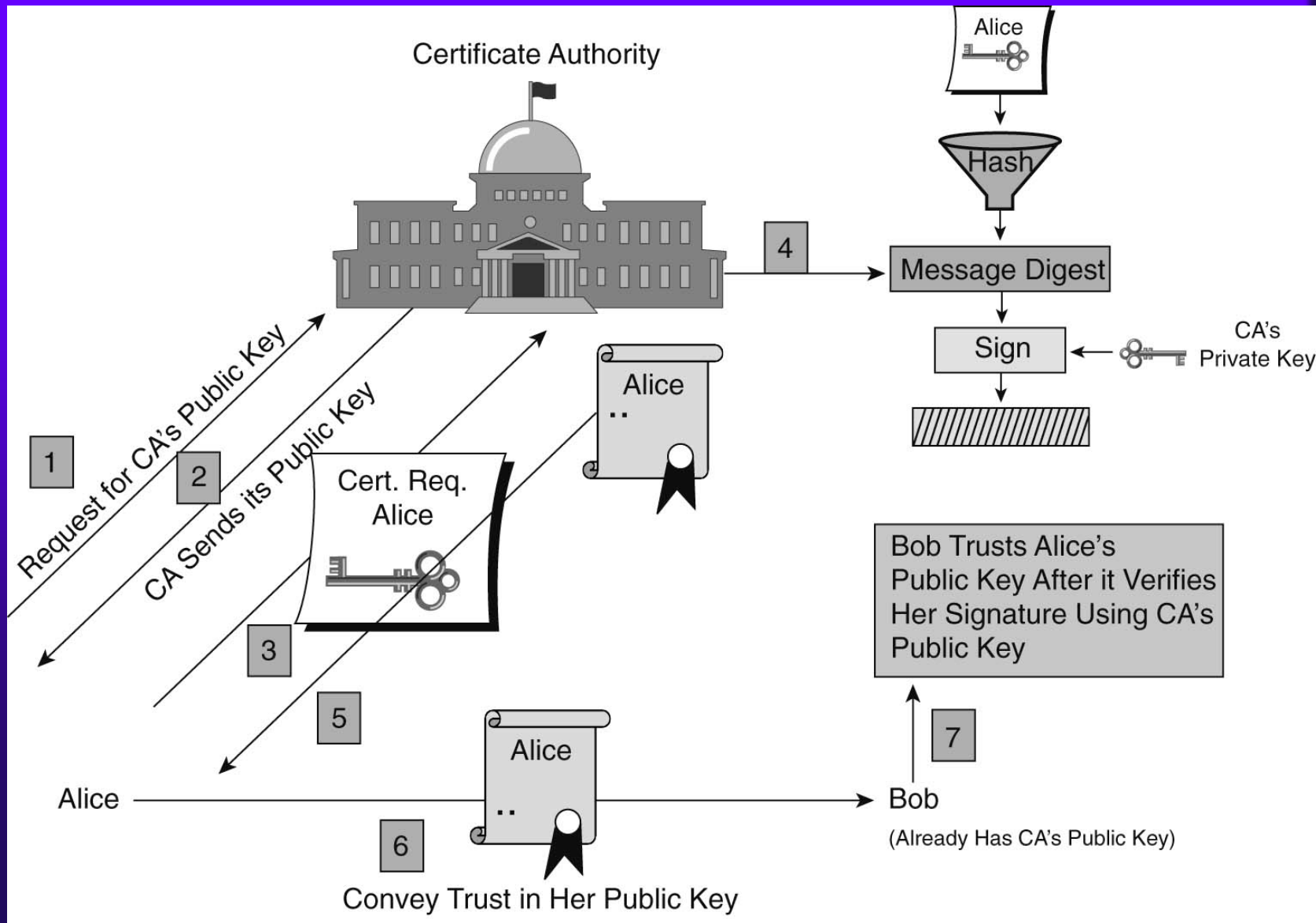
IKE Device Authentication Methods

- Preshared keys
- Digital signatures
- Encrypted nonces

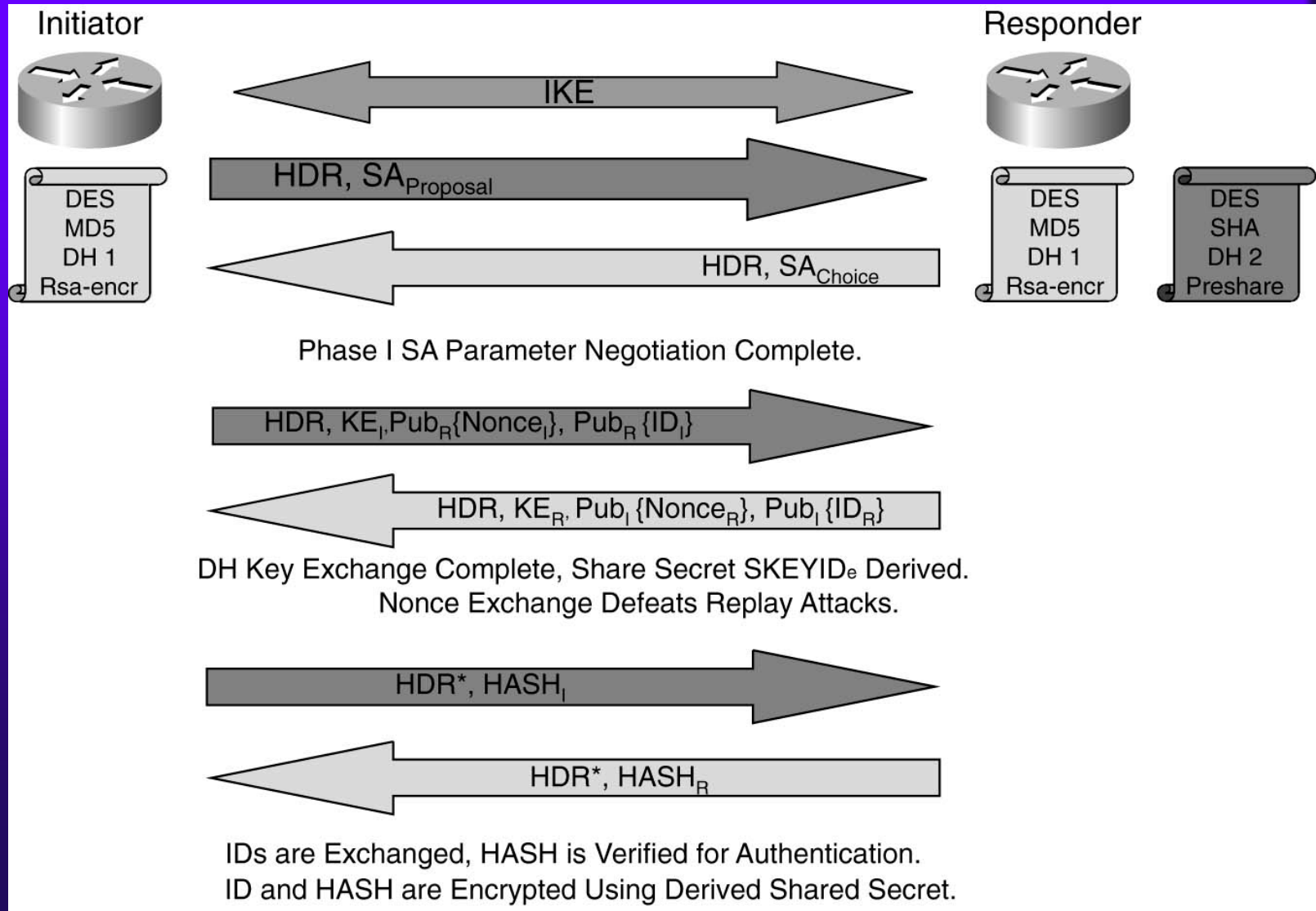
Contents of a Digital Certificate



Using Digital Certificates



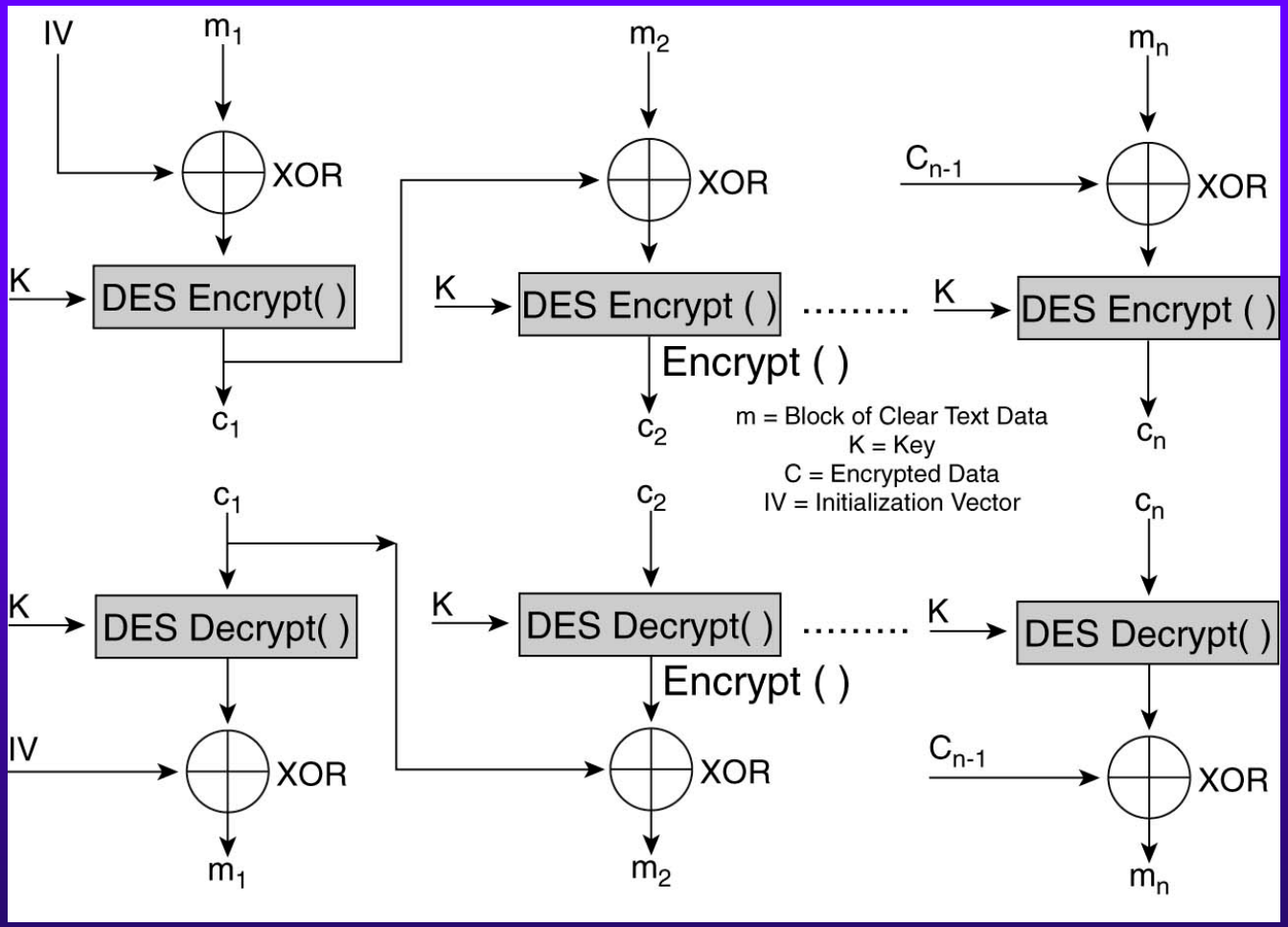
IKE Main Mode Using Encrypted Nonces



Encryption Methods in IPsec

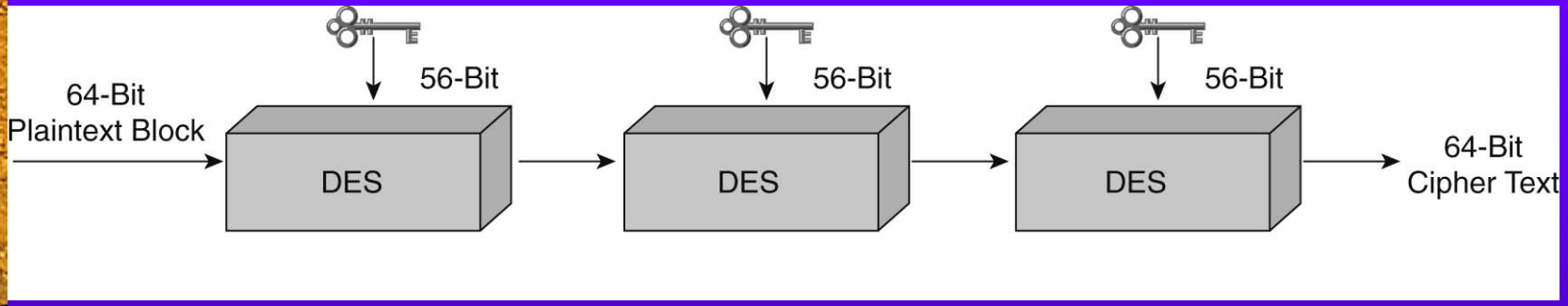
- Data Encryption Standard (DES)
- Triple DES (3DES)

DES Encryption using Cipher Block Chaining (CBC)



- Cipher block: DES encryption algorithm converting fixed-length message into cipher text of same length
-block size of DES is 64 bits while key length is 56 bits
-Initialization vector is sent in ESP header

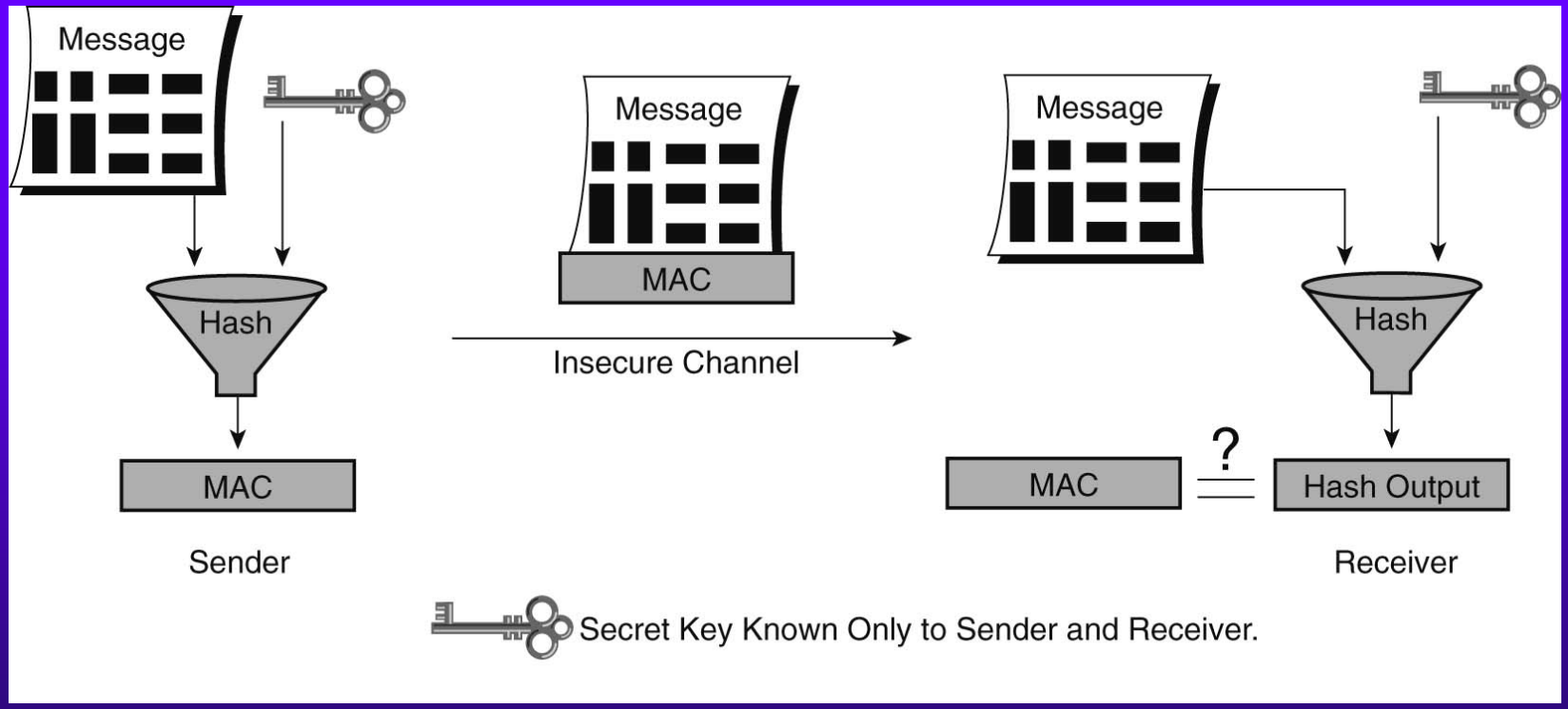
3DES Encryption



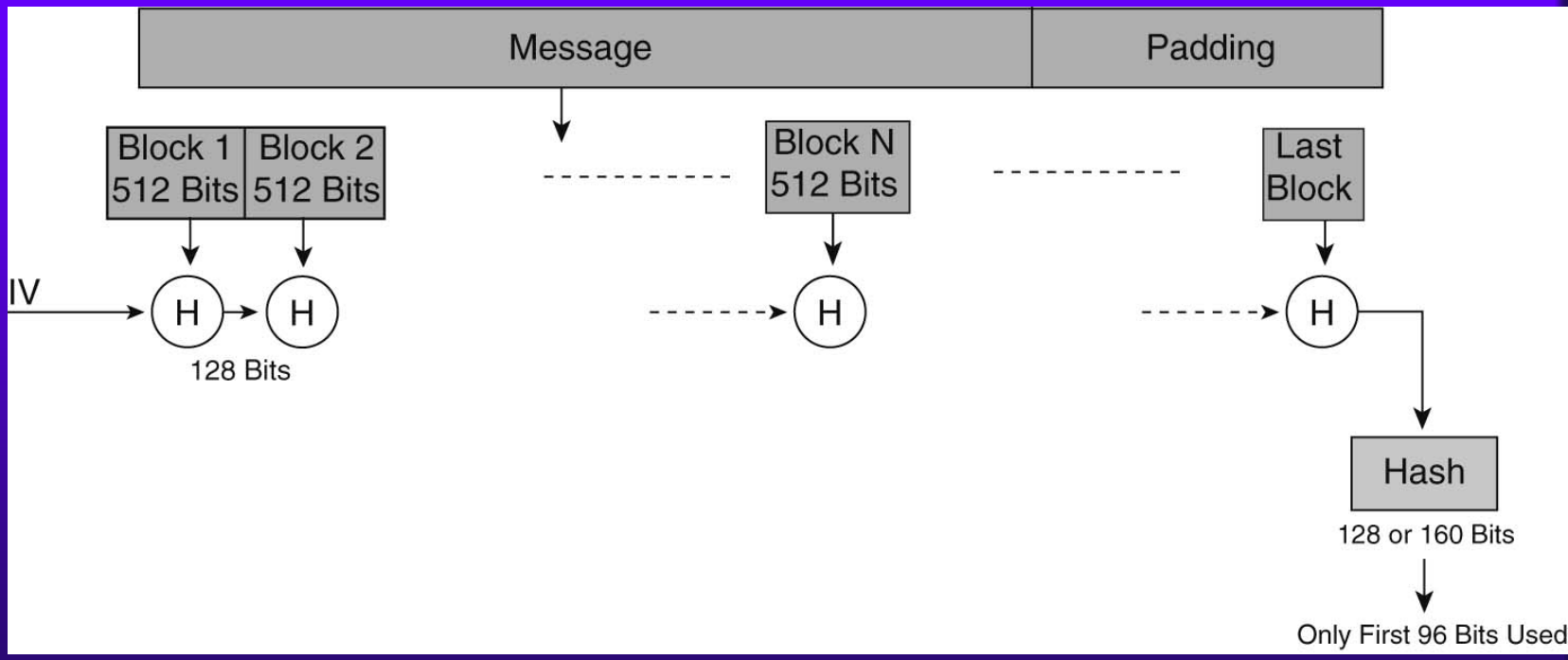
Overall key length is 168 bits

Integrity Checking Mechanism in IPsec

Integrity Checking Using Hashes



Use of Hashes in ESP and AH

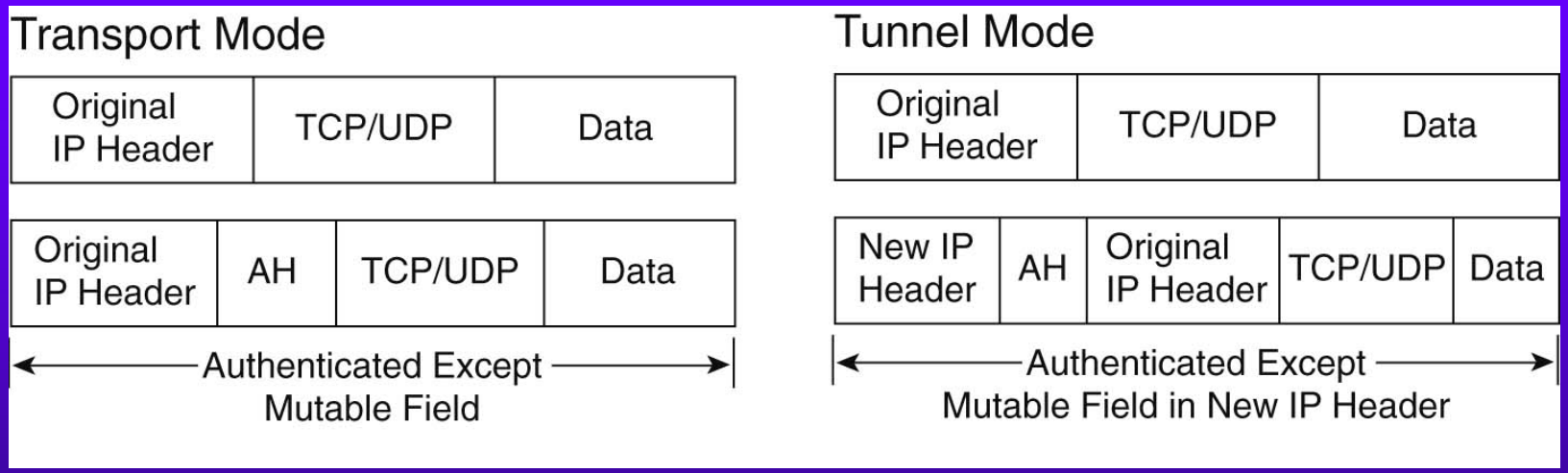


MD5 or SHA hashes are truncated to 96 bits

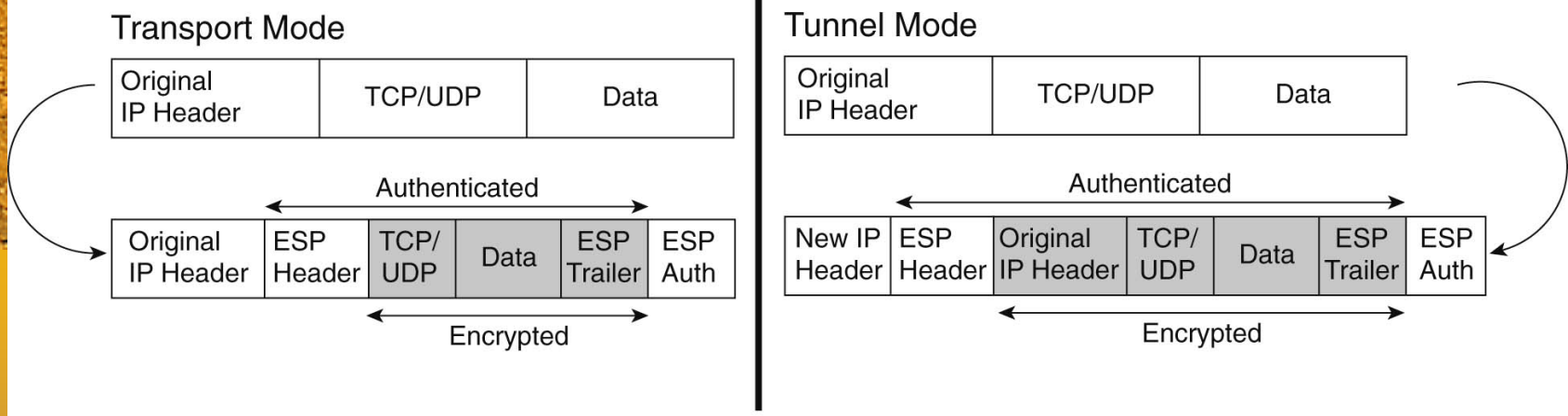
Packet Encapsulation in IPsec

- Transport mode
- Tunnel mode

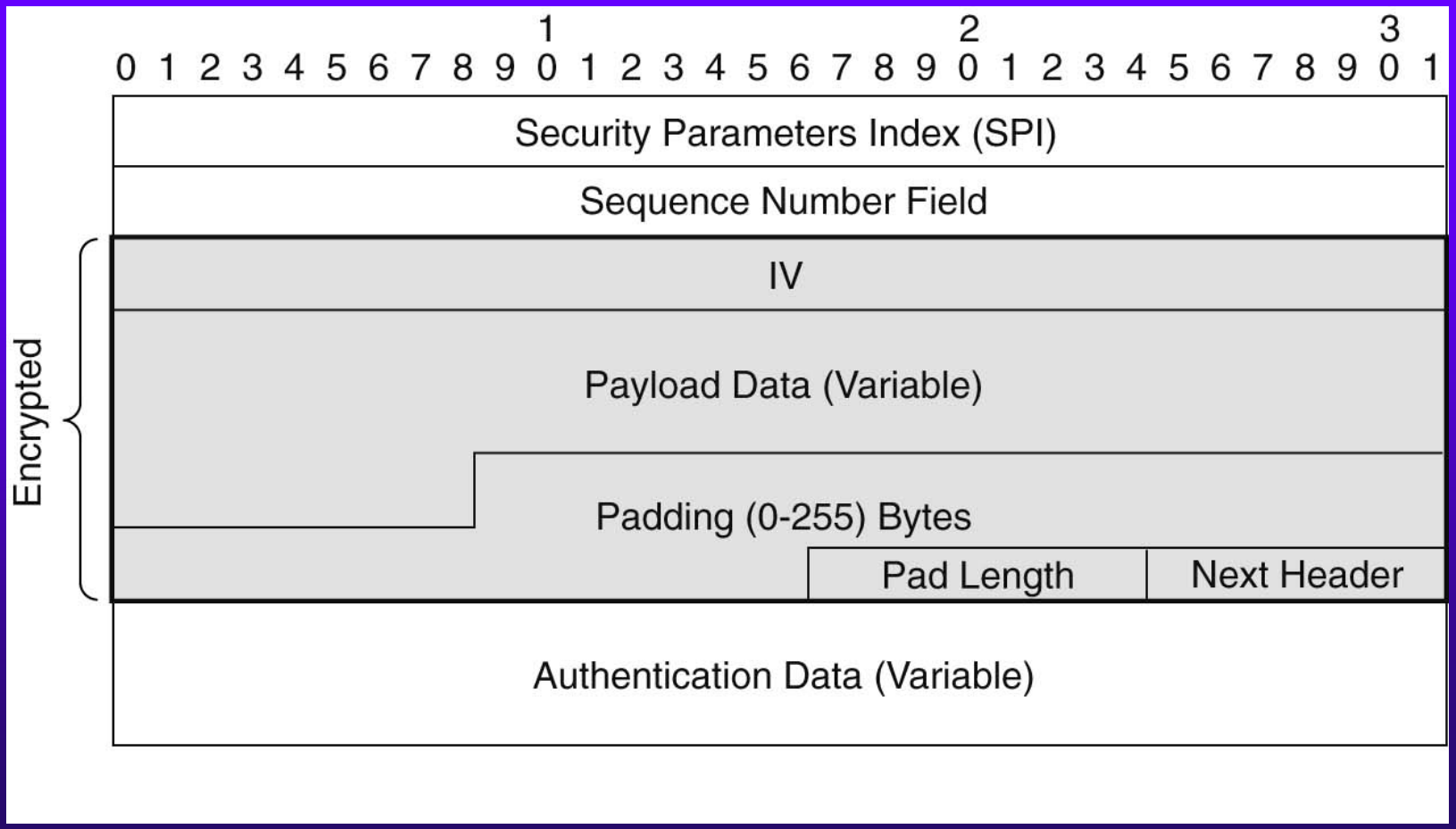
Packet Format Using AH in Tunnel and Transport Modes



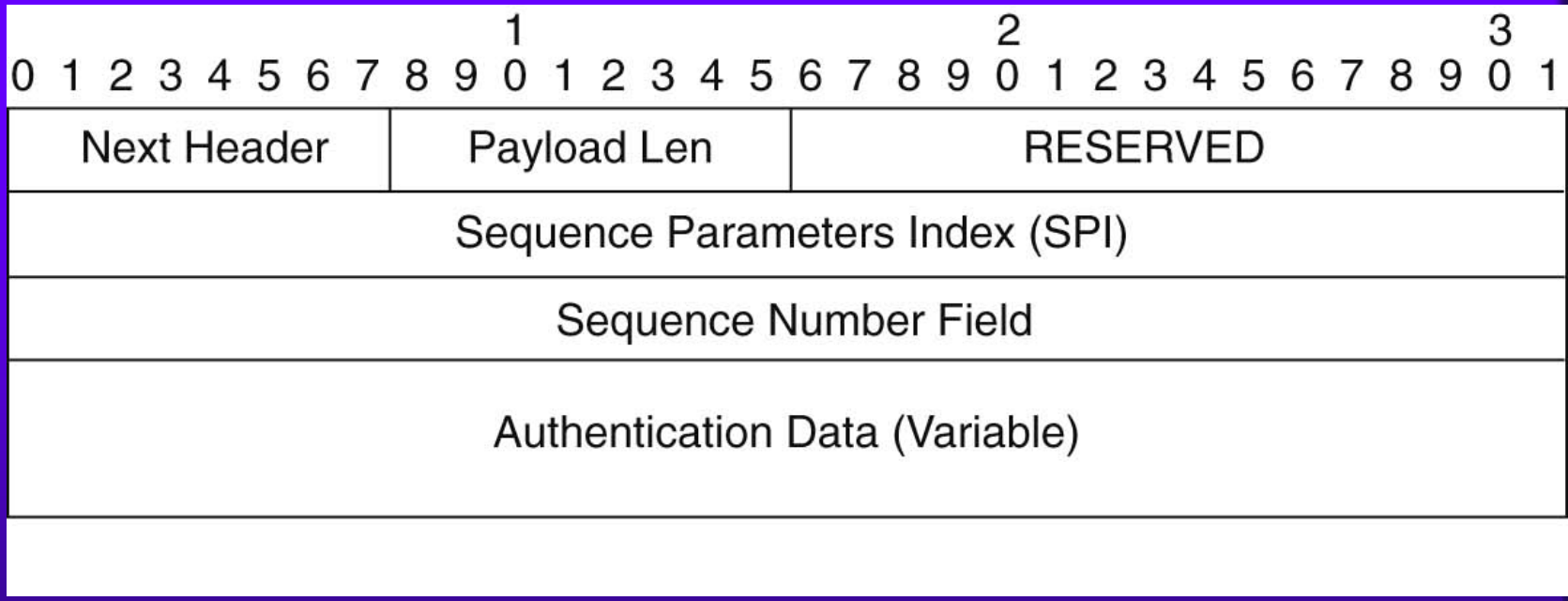
Packet Format Using ESP in Tunnel and Transport Modes



ESP Header Format

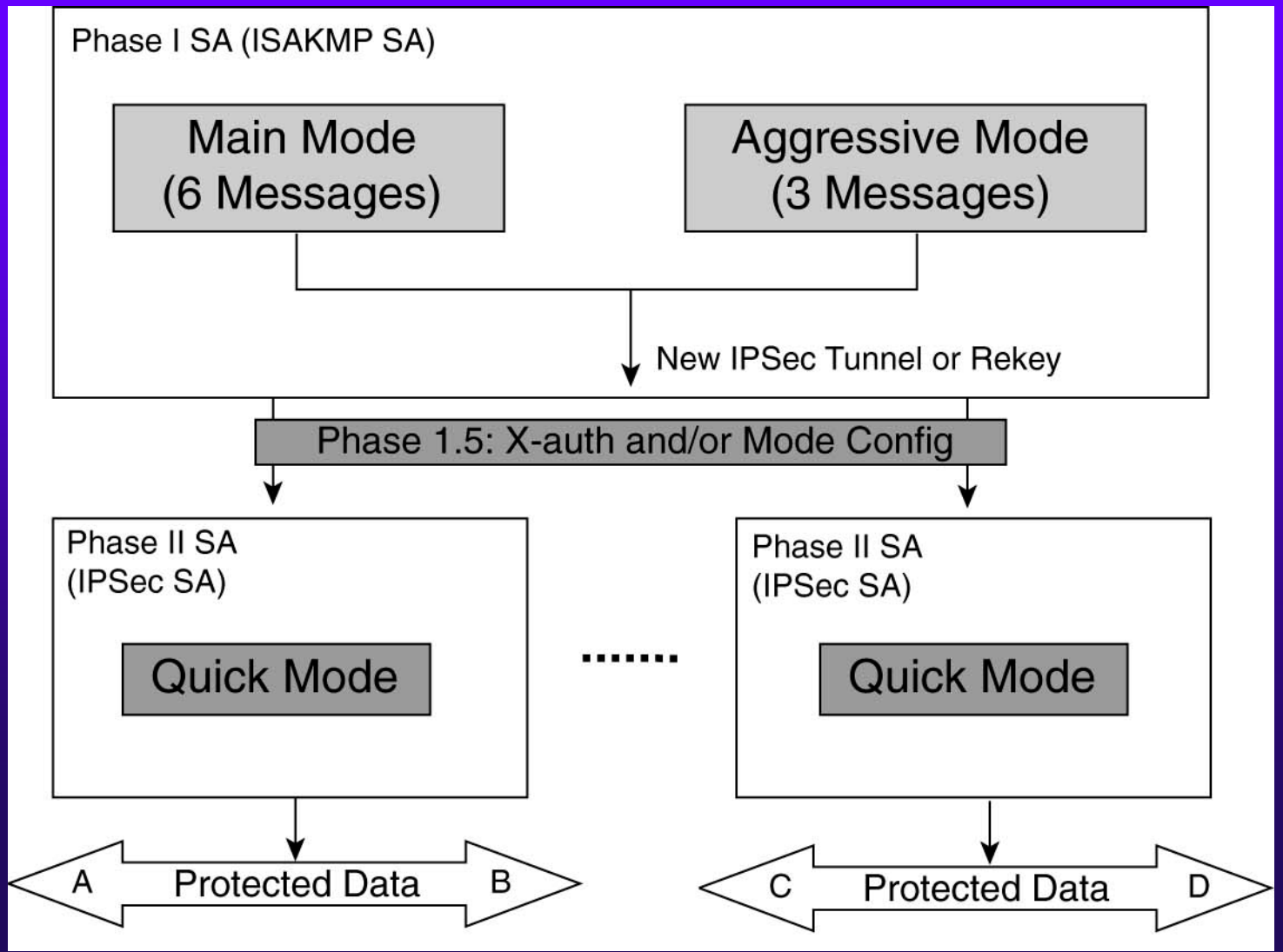


AH Header Format

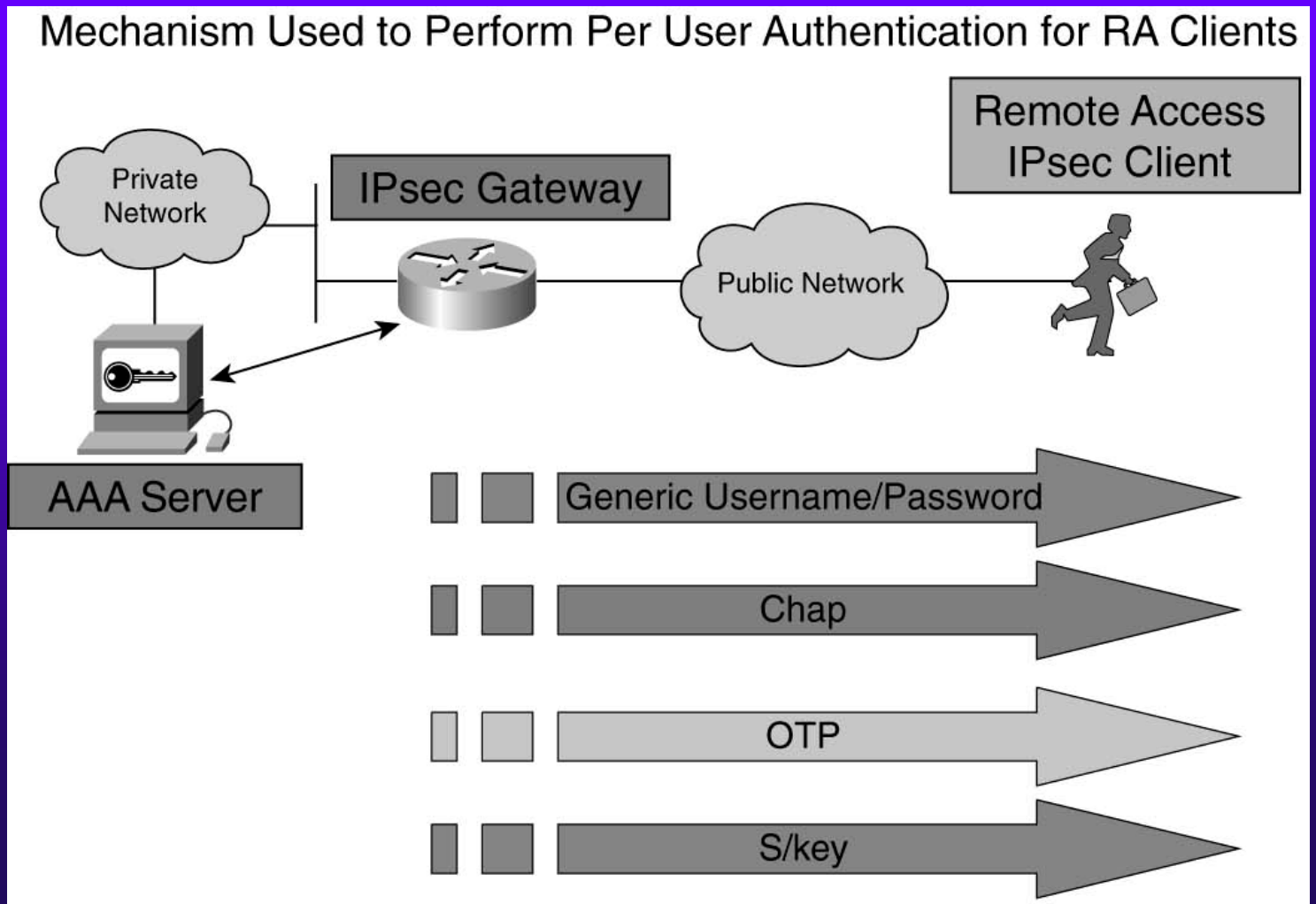


IKE Enhancements for Remote-Access Client IPsec

Extended Authentication and Mode Config in IKE



Negotiation of X-Auth During IKE Negotiation



Start of X-Auth with Exchange of Attribute Payloads Using ISAKMP Messages

Type =

ISAKMP_CFG_REQUEST = 1
or
ISAKMP_CFG_REPLY = 2
or
ISAKMP_CFG_SET = 3
or
ISAKMP_CFG_ACK = 4

ISAKMP HEADER		
Hash	Reserved = 0	**Attributes Payload** Length
Type = 1	Reserved = 0	Identifier
Attributes (0 Length Attributes in Request)		
0	0	**Hash Payload** Length
Hash		

Hash
=
Prf (SKEYID_a,
ISAKMP Header M-ID |
Attribute Payload)

Initiator
IPsec Gateway

Attribute Request

Attribute Reply

Responder
IPsec Client

ISAKM P HEADER		
Hash	Reserved = 0	**Attributes Payload** Length
Type = 2	Reserved = 0	Identifier
Attributes (Set to the Values Requested or XAUTH_STATUS = FAIL)		
0	0	**Hash Payload** Length
Hash		

Attributes =

XAUTH_TYPE	16520
XAUTH_USER_NA	16521
XAUTH_USER_PASSWORD	16522
XAUTH_PASSCODE	16523
XAUTH_MESSAGE	16524
XAUTH_CHALLENGE	16525
XAUTH_DOMAIN	16526
XAUTH_STATUS	16527

Completion of X-Auth with Exchange of Attribute Payloads Using ISAKMP Messages

Type =

ISAKMP_CFG_REQUEST = 1
or
ISAKMP_CFG_REPLY = 2
or
ISAKMP_CFG_SET = 3
or
ISAKMP_CFG_ACK = 4

ISAKMP HEADER		
Hash	Reserved = 0	**Attributes Payload** Length
Type = 3	Reserved = 0	Identifier
Attributes (X-auth_Status = OK or FAIL)		
0	0	**Hash Payload** Length
Hash		

Initiator
IPsec Gateway

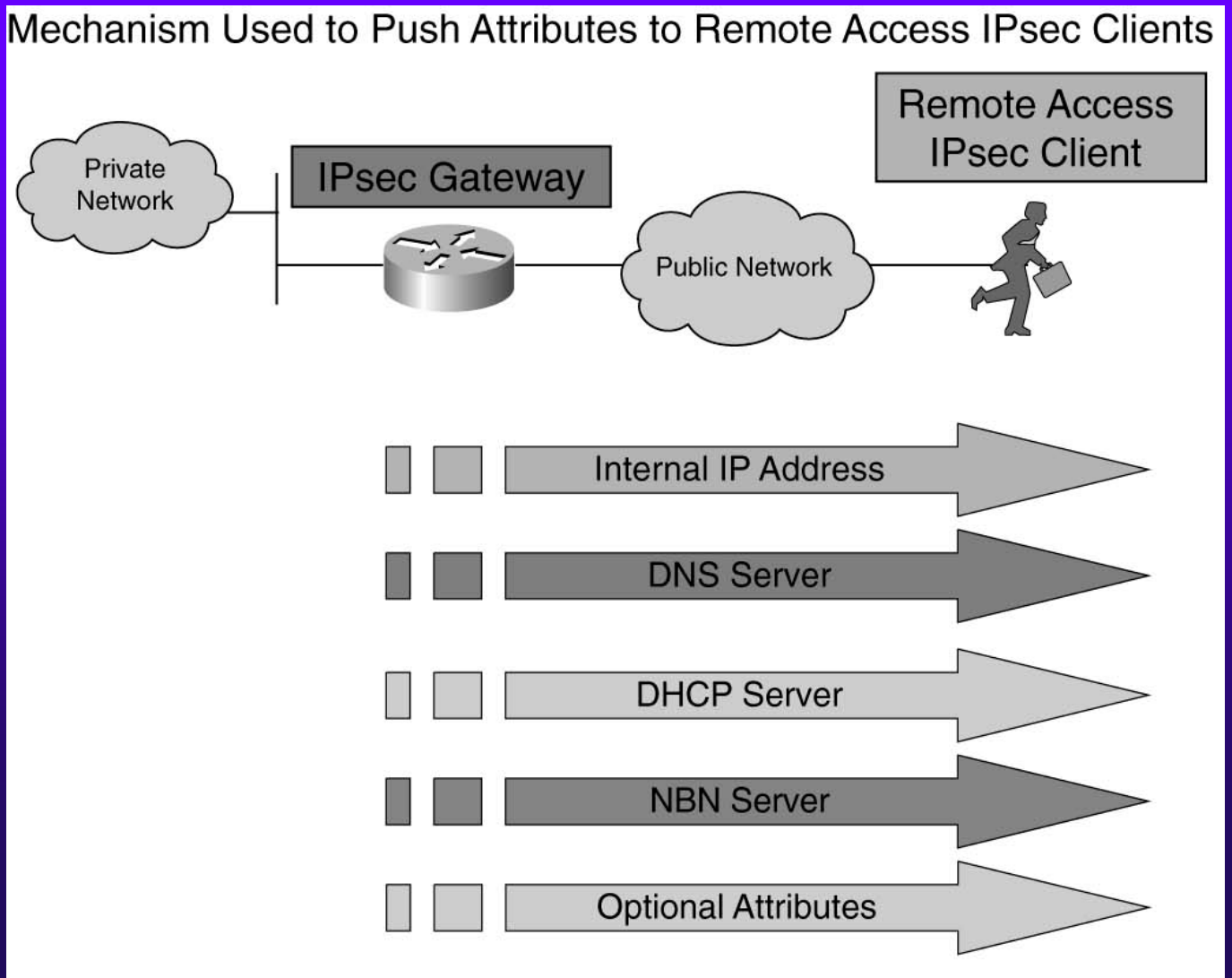
Attribute Set

Attribute ACK

Responder
IPsec Client

ISAKM P HEADER		
Hash	Reserved = 0	**Attributes Payload** Length
Type = 4	Reserved = 0	Identifier
Attributes (None Included)		
0	0	**Hash Payload** Length
Hash		

Mode Configuration During IKE Negotiation



Exchanging Attribute Payloads Between the Gateway and the IPsec Client

Type =

ISAKMP_CFG_REQUEST = 1
or
ISAKMP_CFG_REPLY = 2
or
ISAKMP_CFG_SET = 3
or
ISAKMP_CFG_ACK = 4

ISAKMP Header		
Next Payload	Reserved = 0	**Attributes Payload** Length
Type = 1	Reserved = 0	Identifier
Attributes (0 Length Attributes in Request)		
0	0	**Hash Payload** Length
Hash		

Hash
=
Prf (SKEYID_a,
ISAKMP Header M-ID |
Attribute Payload)

Initiator
Remote Client

Attribute Request

Attribute Reply

Responder
IPsec Gateway

ISAKMP Header		
Next Payload	Reserved = 0	**Attributes Payload** Length
Type = 2	Reserved = 0	Identifier
Attributes (Set to the Values for One or More of the Attributes to Pushed)		
0	0	**Hash Payload** Length
Hash		

Attributes =

INTERNAL_IP4_ADDRESS	1
INTERNAL_IP4_NETMASK	2
INTERNAL_IP4_DNS	3
INTERNAL_IP4_NBNS	4
INTERNAL_IP4_ADDRESS_EXPIRY	5
INTERNAL_IP4_DHCP	6

Other Allowed but not Mandatory

NAT Transparency

Tagging on a UDP Header to Traverse PAT