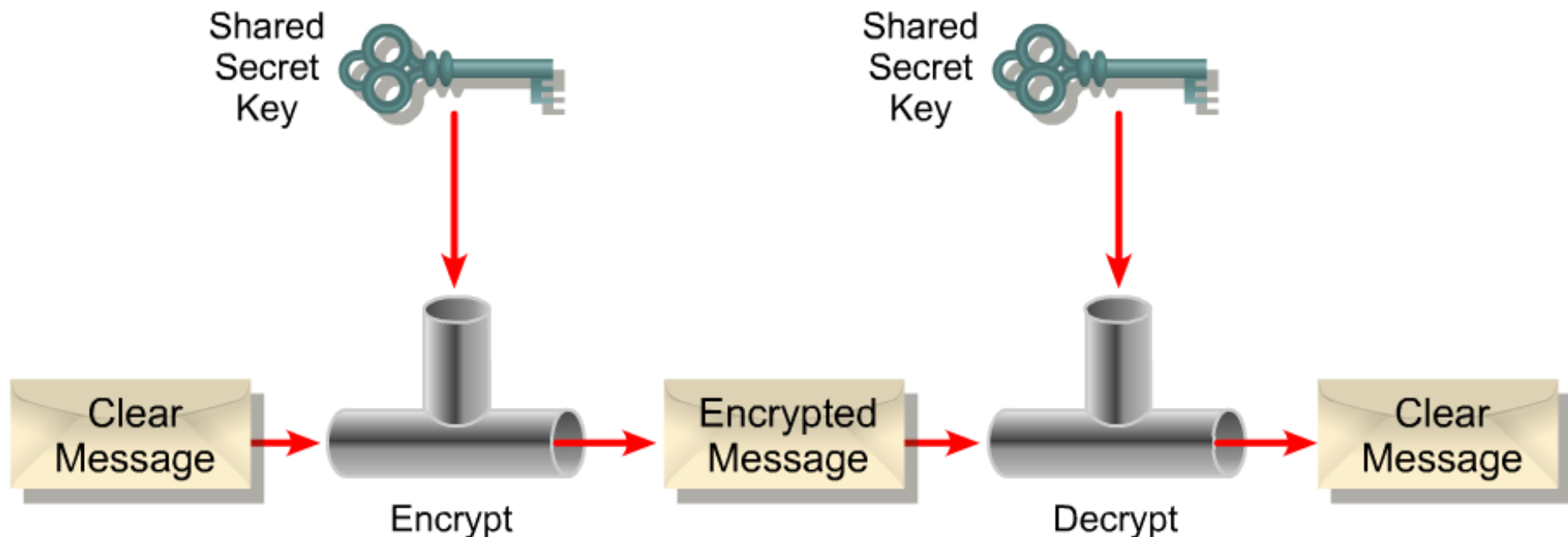


Secure Communications

Segurança em Redes de Comunicações
Mestrado em Cibersegurança
Mestrado em Engenharia de Computadores e
Telemática
DETI-UA

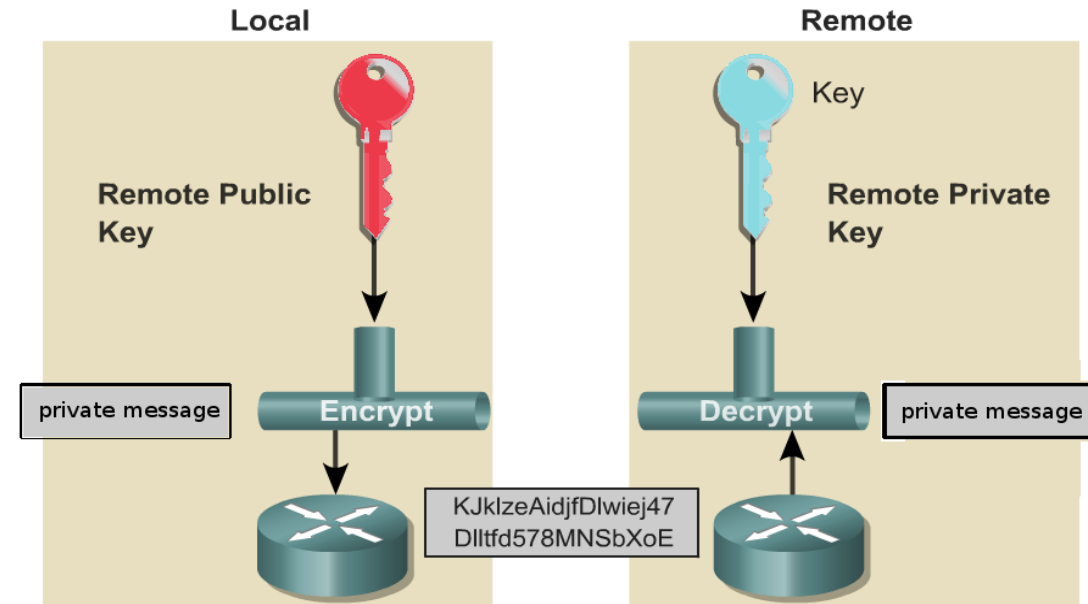
Symmetric Key Cryptography

- Two requirements for secure use of symmetric encryption:
 - Strong encryption algorithm
 - Secret key known only to sender / receiver
- Assume encryption algorithm is known
- Implies a secure channel to distribute key



Public Key Cryptography

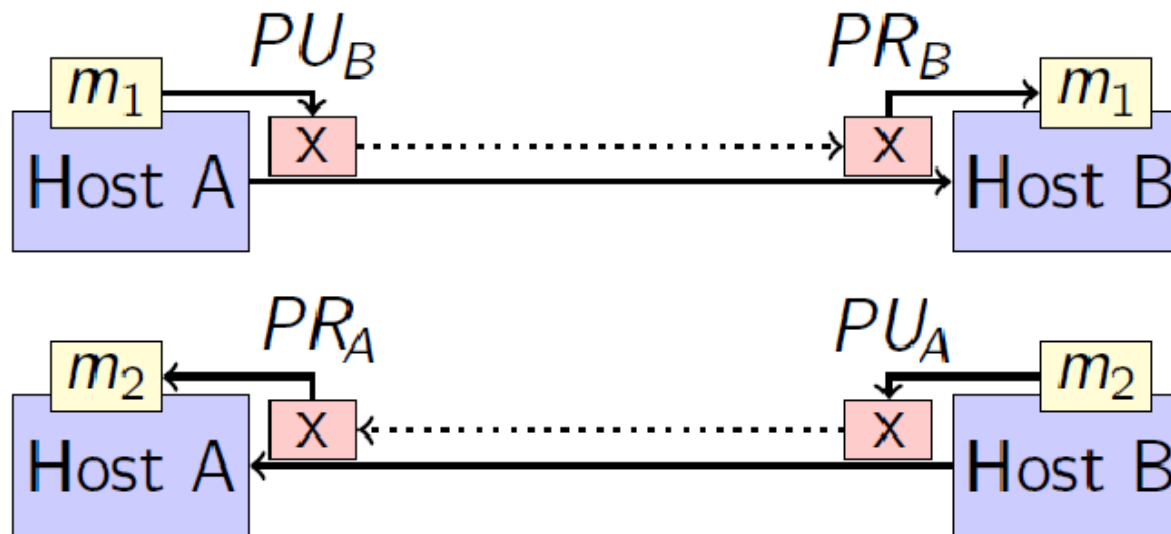
- Public Key Cryptography involves a **pair of keys**
- A **public** key
 - May be **known** by anybody, and can be used to **encrypt** messages, and **verify** signatures
- A **private** key
 - Known only to the recipient, used to **decrypt** messages, and **sign** (create) signatures
- Each public key is published, and the corresponding private key is kept secret
- Is **asymmetric** because those who encrypt messages or verify signatures cannot decrypt messages or create signatures



Public Key

Encryption for confidentiality

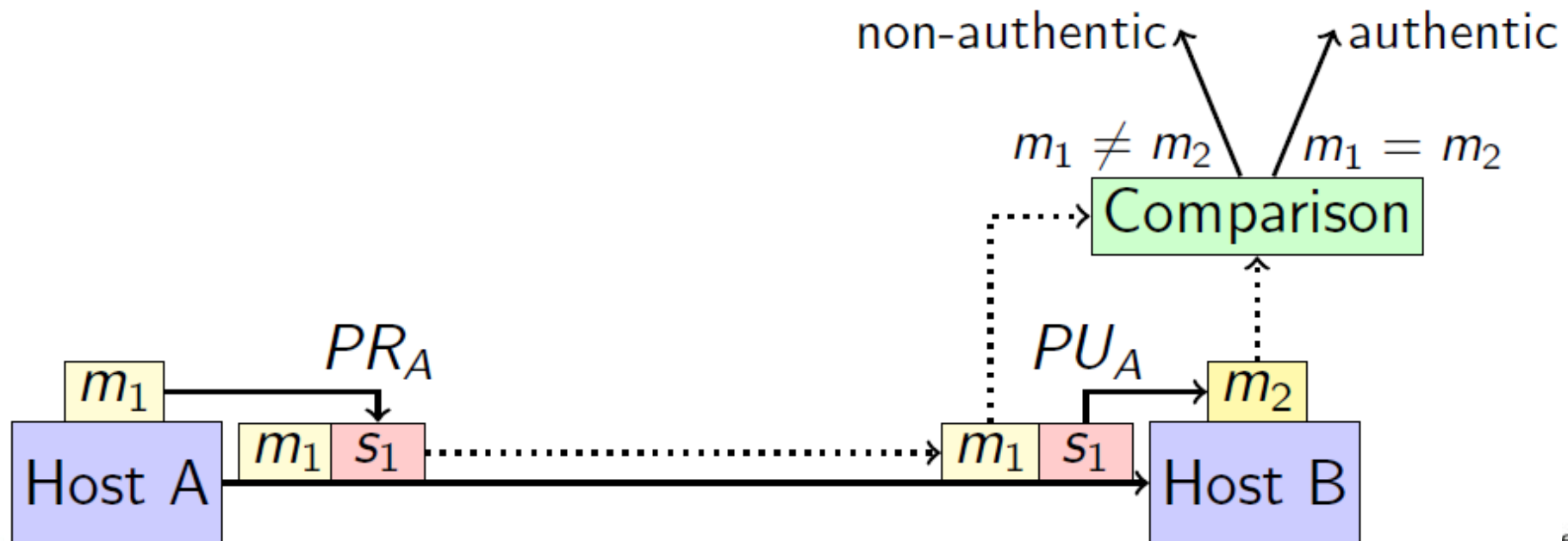
- To send an encrypted message from A to B
 - ♦ Host A encrypts data with Host B public key (PUB)
 - ♦ Host B decrypts data with Host B private key (PRB)
- To send an encrypted message from B to A
 - ♦ Host B encrypts data with Host A public key (PUA)
 - ♦ Host A decrypts data with Host A private key (PRA)
- This method is computational inefficient for encrypting large amounts of data.
- Commonly used to create secure communication channels where a temporary symmetric key can be negotiated and used to encrypt large amounts of data.



Public Key

Digital signatures for authentication

- To send an **authenticated message** from A to B
 - Host A creates a **signature** by encrypting data with Host A **private** key (PR_A)
 - Host A sends data and signature to host B
 - Host B **verifies** data by **decrypting** signature with Host A **public** key (PU_A) and compares with received message



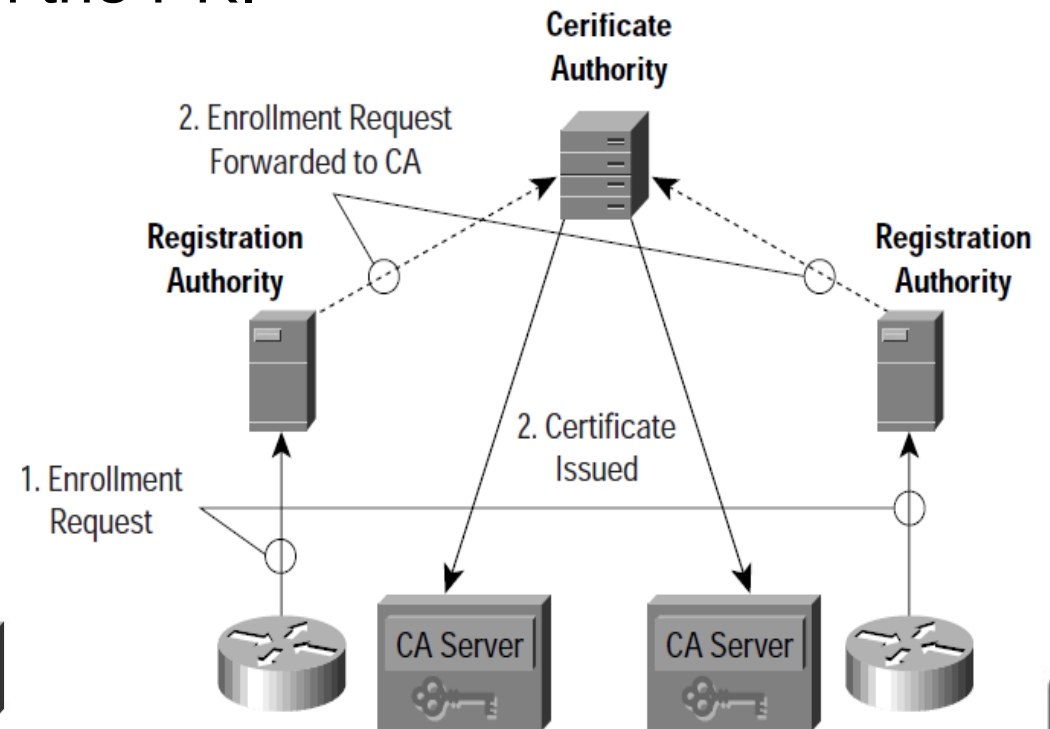
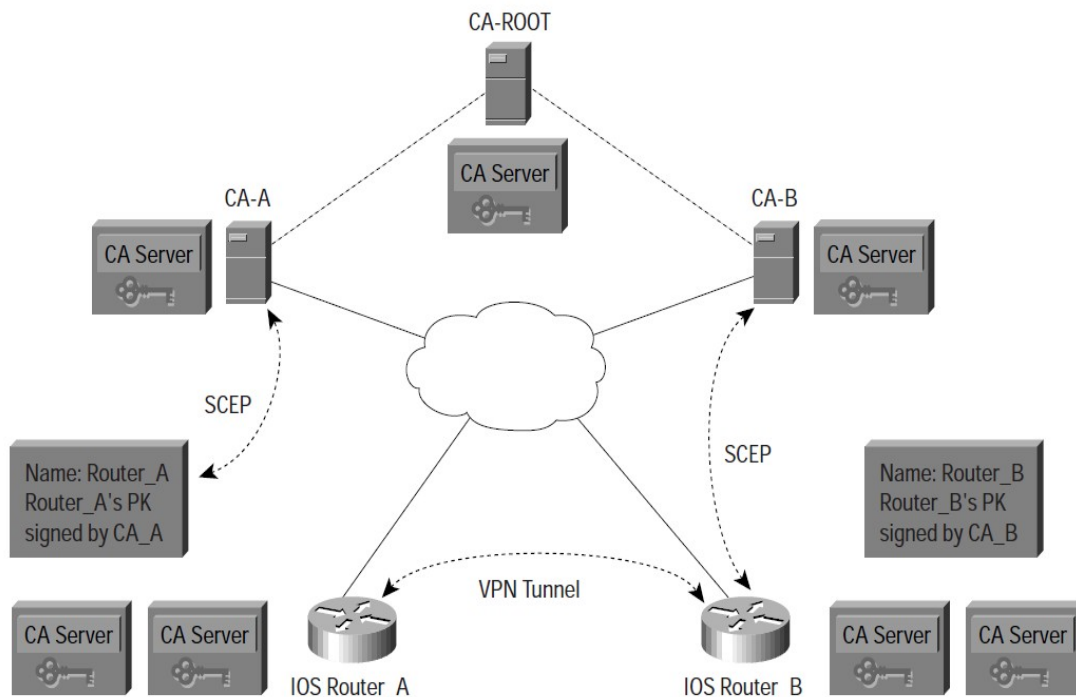
RSA (Rivest, Shamir and Adleman)

- Named after its inventors - Rivest, Shamir and Adleman
- It's a **public key** algorithm (encryption and decryption)
- Key length is variable
 - Common key lengths: 512, 1024 and 2048 bits
- Block size is variable but must be smaller than key length
- Ciphertext length will be the length of the key
- Slower than DES, AES and IDEA
 - Usually **not used** to encrypt **large** messages
 - Used to encrypt secret key and secret key used to encrypt messages



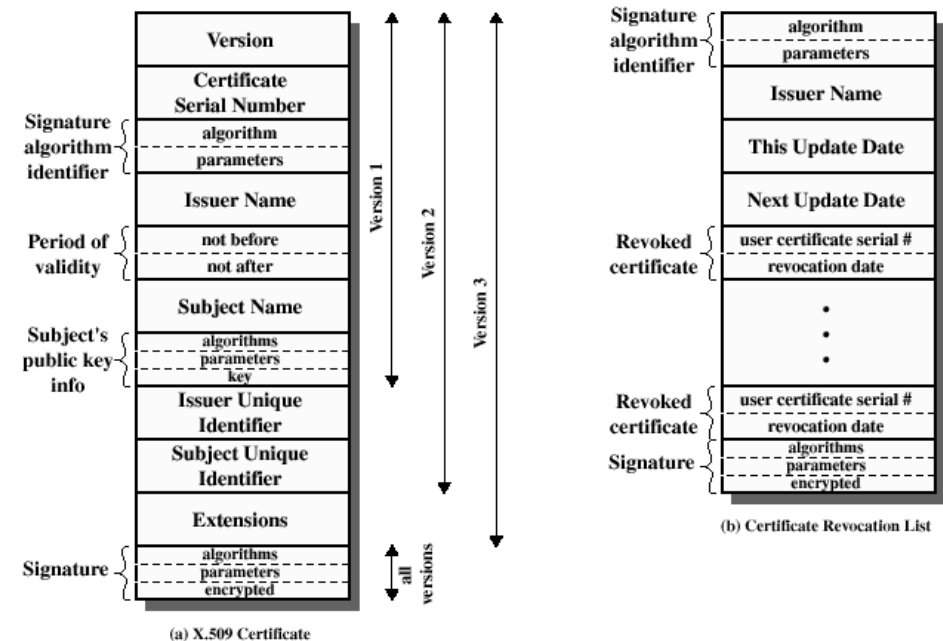
Public Key Infrastructure (PKI)

- PKIs are **hierarchical** in nature
- Each PKI participant holds a **digital certificate** that has been issued by a **Certificate Authority (CA)**
 - CA may be a root CA or a subordinate CA
 - ➔ **Trust chain.**
- PKI might use additional hosts called **Registration Authorities (RA)** to accept requests for enrollment in the PKI



X.509 certificate contents

- Version
- Serial Number
- Signature Algorithm
- Issuer Name
- Validity Period
- Subject Name

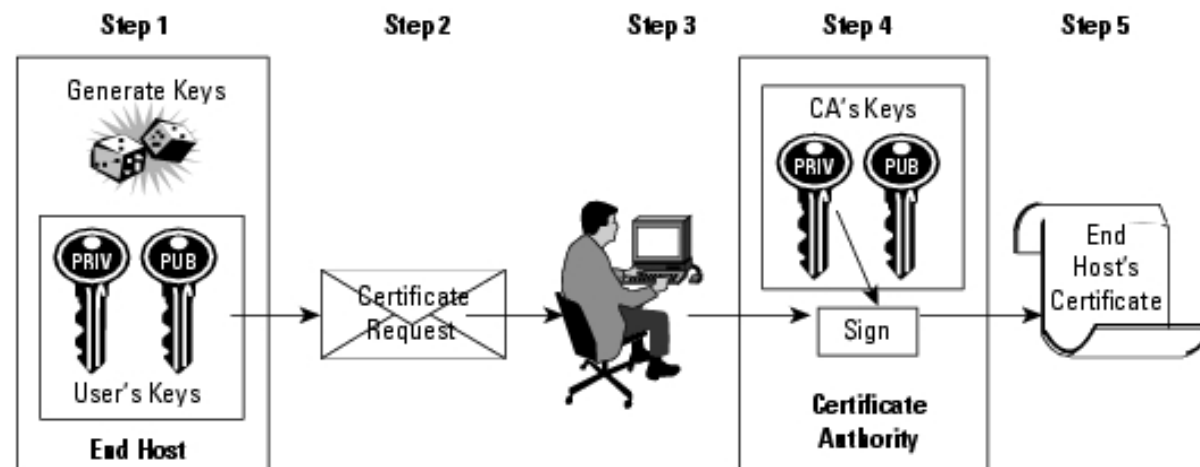


- Subject Name
 - Distinguished Name (DN) of the entity
 - CN=Java Duke, OU=Java Software Division, O=U.Aveiro, C=PT
- Subject Public Key Information
 - Public Key Algorithm
 - Subject Public Key
- Certificate Signature Algorithm
- Certificate Signature



Certificate Authority enrollment

- Simple Certificate Enrollment Protocol (SCEP) is used for **secure transportation** of **key** information and **certificates**
- Enrolling in a Certificate Authority
 1. End host generates a private-public key pair
 2. End host generates a certificate request, which it forwards to the CA
 3. Manual, human intervention is required to approve the enrollment request,
 4. After the approval, the CA **signs** the **certificate** with its **private** key and returns the completed certificate to the end host
 5. End host stores certificate

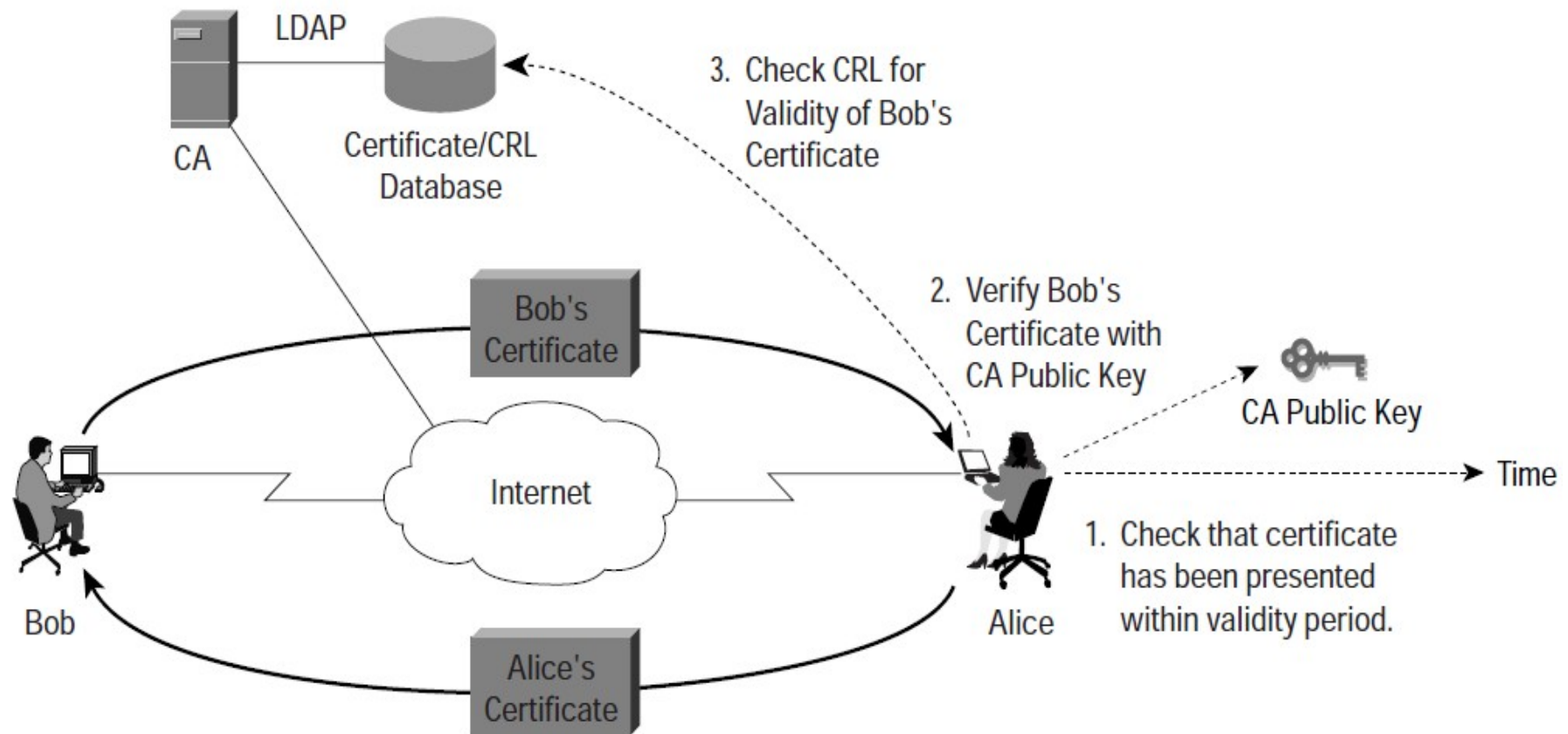


Certificate **Revocation Lists** (CRL)

- The CRL is another crucial PKI component
- Is a list of certificates that were formerly valid within the PKI, but have been **revoked** for some reason
- These reasons could include any of the following:
 - ♦ **Compromise** of keys within certificate
 - ♦ **Loss of access** privileges for user/device
 - ♦ Change of PKI structure requiring certificate re-issue



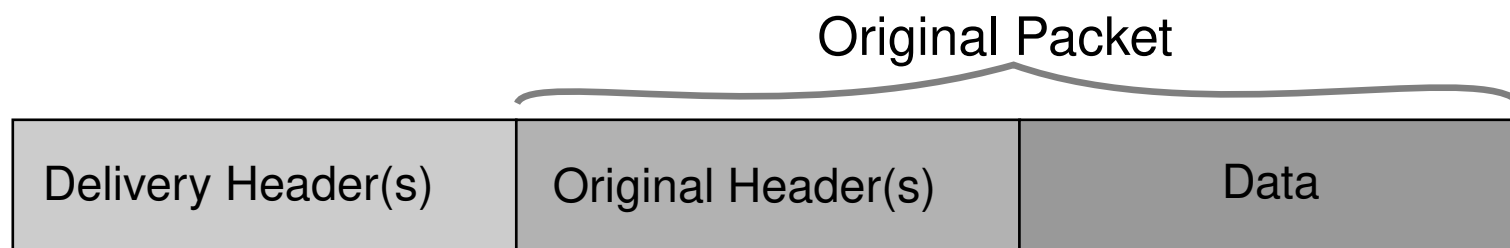
Certificate usage and validity check



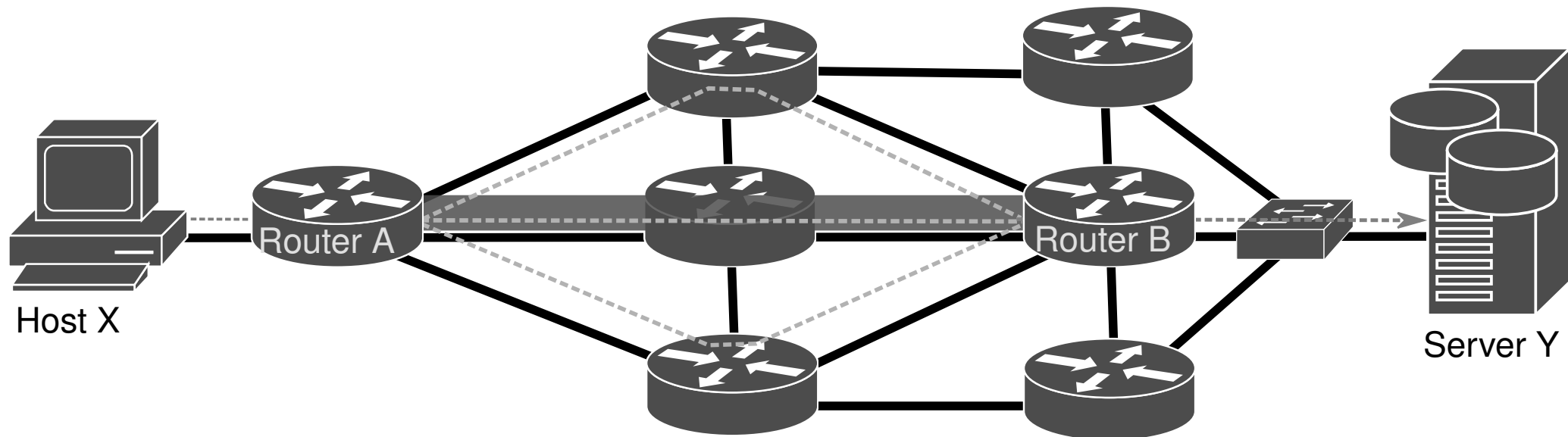
- The cert is being presented within its **validity period**
- The CA that signed the cert is **known** and **trustable**
- The certificate is **not on** a **revocation list** (optional in some scenarios)

Traffic Tunnel Concept

- Main purposes
 - Guarantee that a packet that reaches a network node will reach a specific secondary network node **independently** of the intermediary nodes routing processes,
 - Guarantee the delivery of a packet to a remote node when the intermediary nodes do not support the original packet network protocol, and,
 - Define a **virtual channel** that adds additional **data transport** features in order to provide differentiated **QoS**, **security requirements** and/or **optimized routing**.
- Achieved by adding, at the tunnel entry point, one or more protocol headers to the original packets to handle their delivery to the tunnel exit point.



Tunnel End-Points



Delivery protocol(s)	Original protocol(s)	Data
Source: A address Destination: B address	Source: X address Destination: Y address	

Virtual Tunnel Interface (VTI)

- Logical construction that creates a **virtual network interface** that can be handled as any other network interface within a network equipment.
- A tunnel does not require to have any network addresses other the ones already bound to the end-point router.
- However, most implementations impose that a network **address** must be **bound** to a **tunnel interface** in order to enable IP processing on the interface.
 - The tunnel interface may have a explicitly bound network address or reuse an address of another interface already configured on the router.

```
1 #interface Tunnel 1
2 #ip address 10.1.1.1 255.255.255.252
3 #ipv6 address 2001:A:A::1/64
4 #ip unnumbered FastEthernet0/0
5 #ipv6 unnumbered FastEthernet0/0
6 #ip ospf cost 10
7 #ipv6 ospf 1 area 0
8 #tunnel mode ipip
9 #tunnel source FastEthernet0/0
10 #tunnel destination 200.2.2.2
```



VTI Requirements

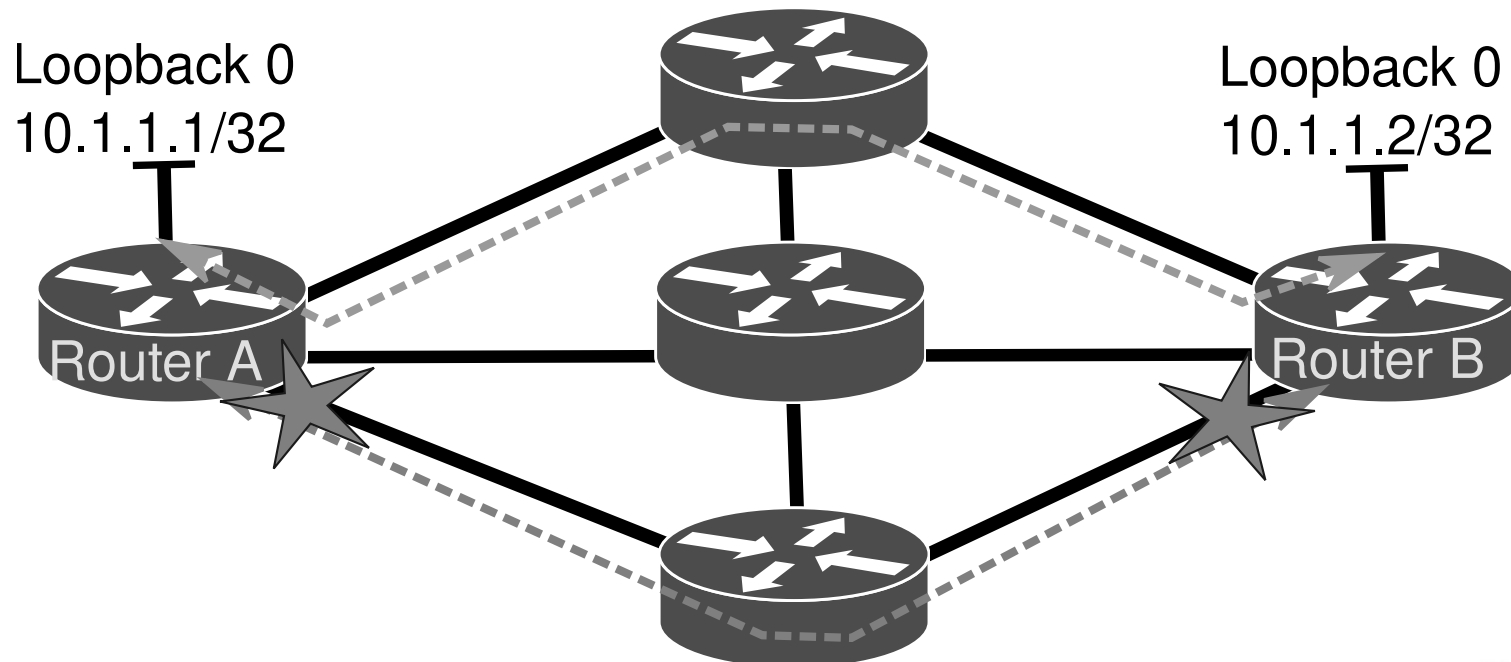
- A numeric identifier,
- A bounded IP address, this will enable IP processing,
 - ♦ Add the tunnel interface to the routing table and allow routing via the interface,
- A defined mode or type of tunnel,
 - ♦ Availability of tunnel models depends on the Router model, operating software and licenses.
- Tunnel **source**,
 - ♦ Defined as the name of the local interface or IPv4/IPv6 address depending on the type of the tunnel.
- Tunnel **destination**,
 - ♦ Defined as a domain name or IPv4/IPv6 address depending on the type of the tunnel.
 - ♦ This definition is not mandatory for all types of tunnels because in some cases the tunnel end-point is determined dynamically.
- May optionally have additional configurations for routing, security and QoS purposes.

```
1 #interface Tunnel 1
2 #ip address 10.1.1.1 255.255.255.252
3 #ipv6 address 2001:A:A::1/64
4 #ip unnumbered FastEthernet0/0
5 #ipv6 unnumbered FastEthernet0/0
6 #ip ospf cost 10
7 #ipv6 ospf 1 area 0
8 #tunnel mode ipip
9 #tunnel source FastEthernet0/0
10 #tunnel destination 200.2.2.2
```



Loopback Interfaces as End-Points

- Loopback interface is another **logical construction** that creates a virtual network interface completely independent from the remaining physical and logical router network interfaces.
- The main propose of a loopback interface is to provide a network address to serve as router identifier in remote network configurations and distribute algorithms.
- The main advantage of using loopback interfaces as tunnel end-points, is the creation of a tunnel not bounded to any individual network card/link that may fail.



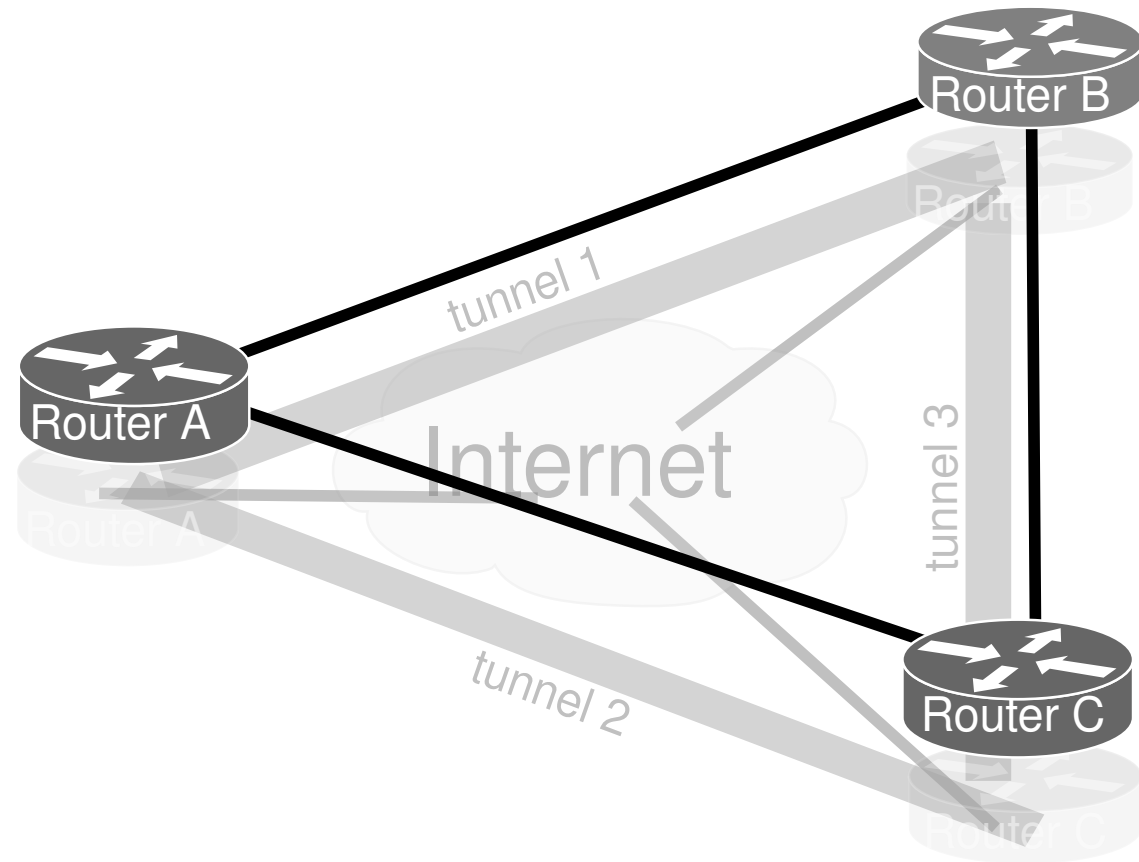
IP Tunnel Types

- **IPv4-IPv4**
 - ♦ Original **IPv4** packets are **delivered** using **IPv4** as network protocol.
- **GRE IPv4**
 - ♦ Original packets protocol (any network protocol) is defined by **GRE header** and delivered using **IPv4** as network protocol.
- **IPv6-IPv6**
 - ♦ Original **IPv6** packets are delivered using **IPv6** as network protocol.
- **GRE IPv6**
 - ♦ Original packets **protocol** (any network protocol) is defined by a **GRE header** and delivered using **IPv6** as network protocol.
- **IPv6-IPv4**
 - ♦ Original **IPv6** packets are delivered using **IPv4** as network protocol.
- **IPv4-IPv6**
 - ♦ Original **IPv4** packets are delivered using **IPv6** as network protocol.

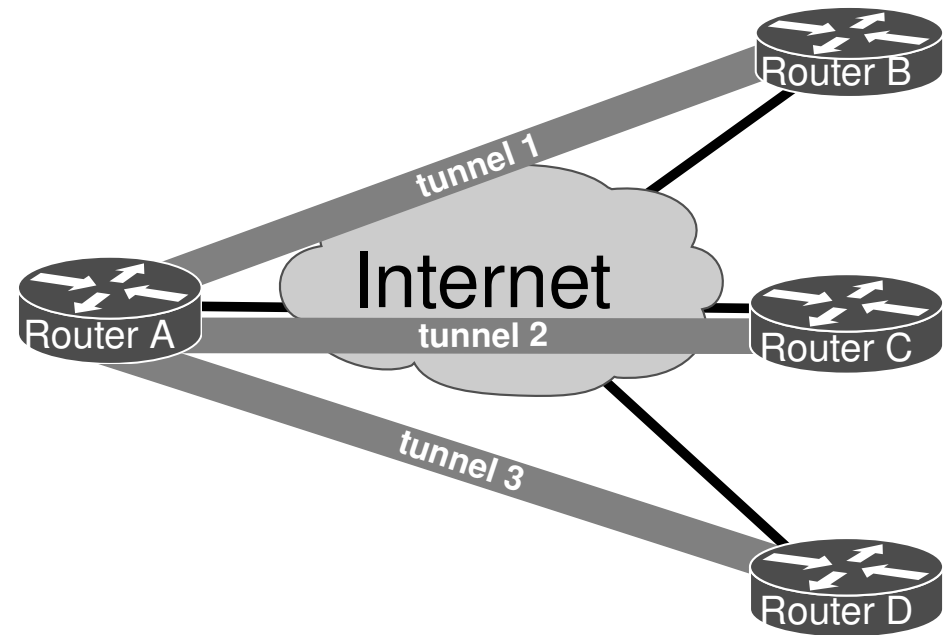
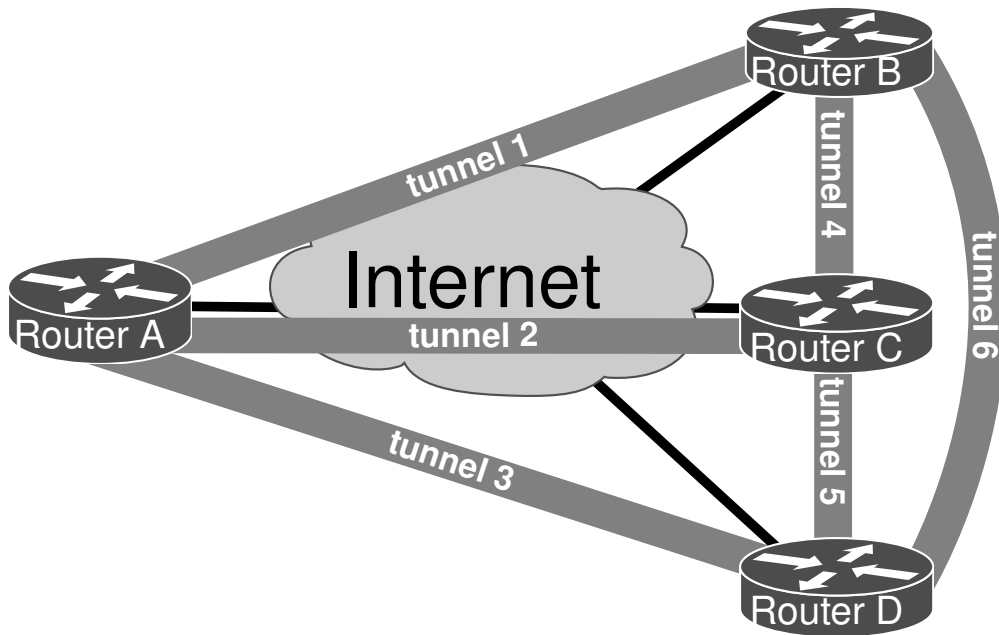


Overlay Network

- An overlay network can be defined as a **virtual network** defined **over another** network.
 - ♦ For a specific purpose like private transport/routing policies, QoS, security.
- The underlying network can be physical or also virtual.
 - ♦ May result in multiple layers of overlay networks.
- When any level of privacy protocol is present on an overlay network is designated by Virtual Private Network (VPN).



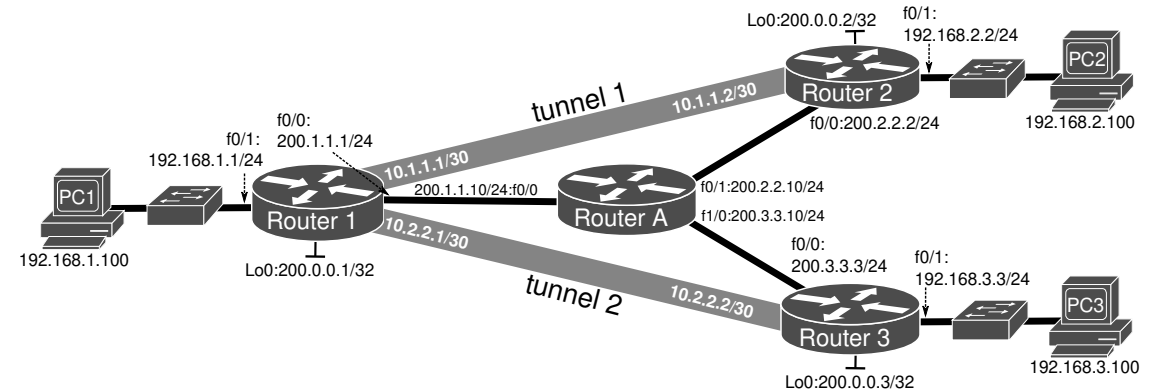
Full/Partial Overlay Mesh



Routing Through/Between Tunnels

• Static Routes

```
1 #ip route 192.168.2.0 255.255.255.0 Tunnel1
2 #ip route 192.168.2.0 255.255.255.0 10.1.1.2
3 #ipv6 route 2001:A:1::/64 Tunnel1
4 #ipv6 route 2001:A:1::/64 2001:0:0::2
5 #ip route 192.168.2.100 255.255.255.255 10.1.1.2
6 #ipv6 route 2001:A:1::100/128 2001:0:0::2
```



• Policy Based Routing (route-maps)

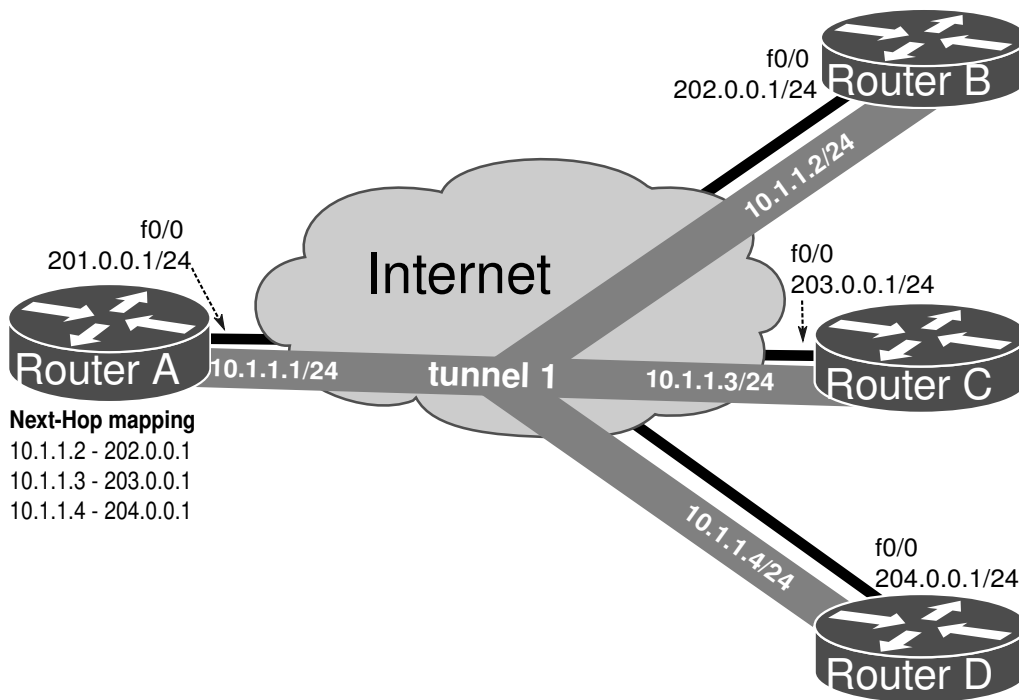
```
1 #access-list 100 permit ip host 192.168.1.100 192.168.2.0 255.255.255.0
2 #route-map routeT1
3   #match ip address 100
4   #set ip next-hop 10.1.1.2
5 #interface FastEthernet0/1
6   #ip policy route-map routeT1
```

• Dynamic Routing

- Multiple (distinct) routing processes.
 - One per overlay network, and
 - One for the underlying network.

```
1 #router ospf 1
2   #network 200.1.1.0 0.0.0.255 area 0
3   #network 200.0.0.1 0.0.0.0 area 0
4   !
5 #router ospf 2
6   #network 10.0.0.0 0.255.255.255 area 0
7   #network 192.168.0.0 0.0.255.255 area 1
```

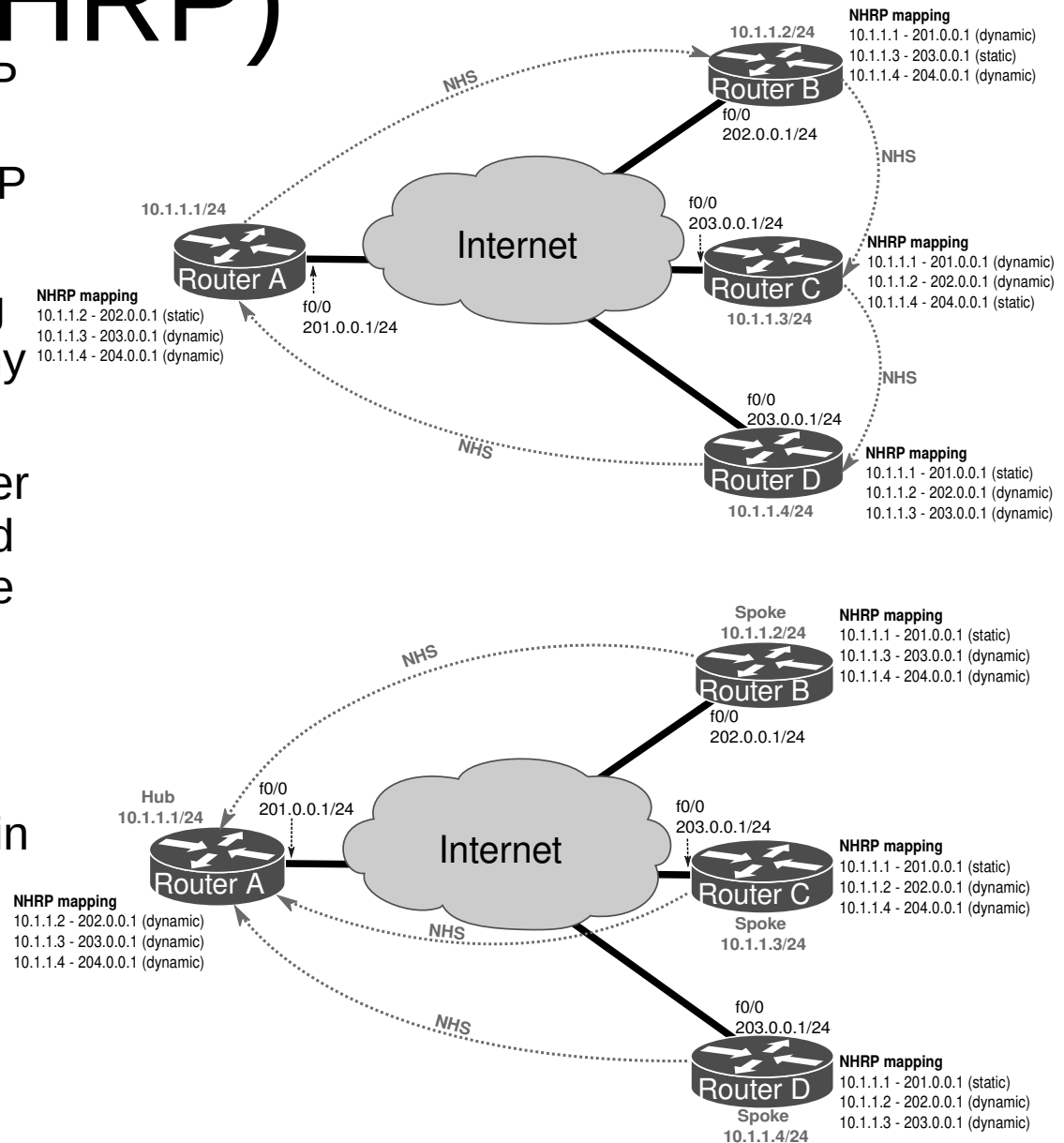

Multipoint Tunnels



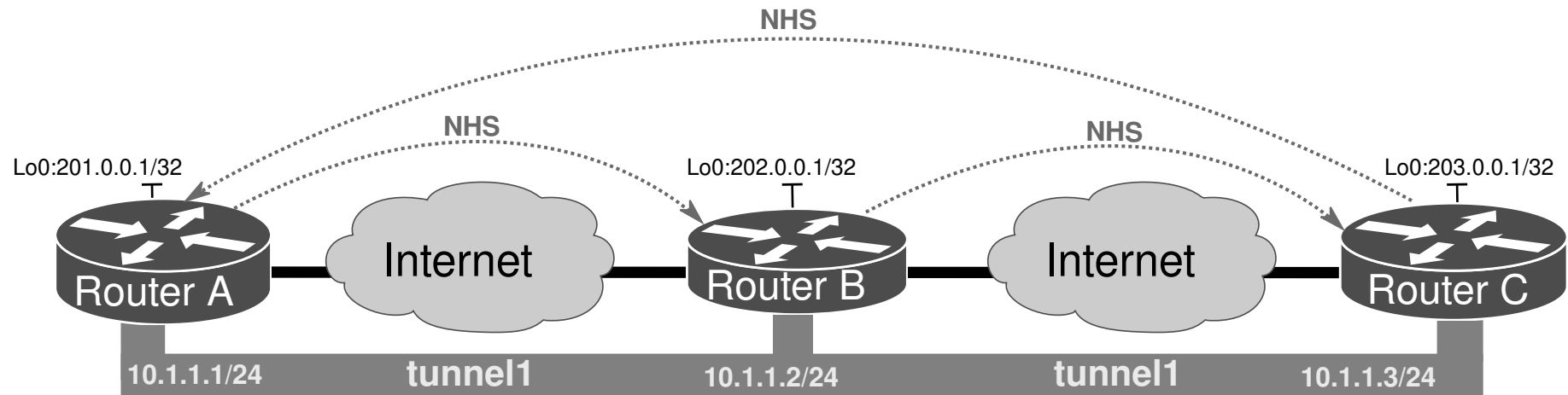
- In a scenario with many nodes to interconnect, the simpler and more efficient approach is to have a single **tunnel** that interconnect **multiple nodes** - a multipoint tunnel.
- Directly connect using a single virtual overlay IP network, defined within a multipoint tunnel.
- In a multipoint tunnel scenario, the delivery **header** address is determined based on the address of the **next hop** within the overlay network.
- Address mapping between overlay and underlying network addresses may be statically defined or dynamically obtained.

Next Hop Resolution Protocol (NHRP)

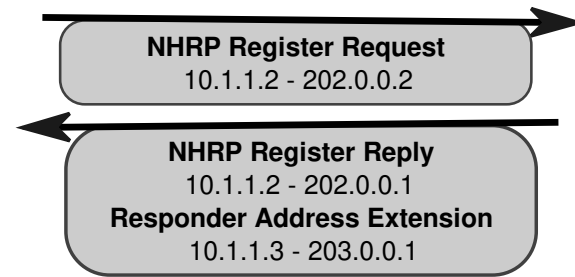
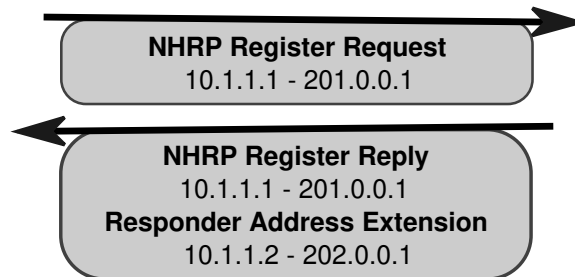
- NHRP allows to map a tunnel interface IP address (overlay network) to the respective underlying network interface IP address.
- NHRP tunnel requires that all intervening nodes should be able to find a path to any of the other nodes.
- Each node should at least know one other overlay node (and respective overlay and underlying addresses) through which he will try to find the other nodes address mappings.
 - Next Hop Server (NHS).
- Moreover, all nodes must be configured in a way that all nodes have at least one valid path to all other nodes - forming a partial mesh.



NHRP Information Exchange



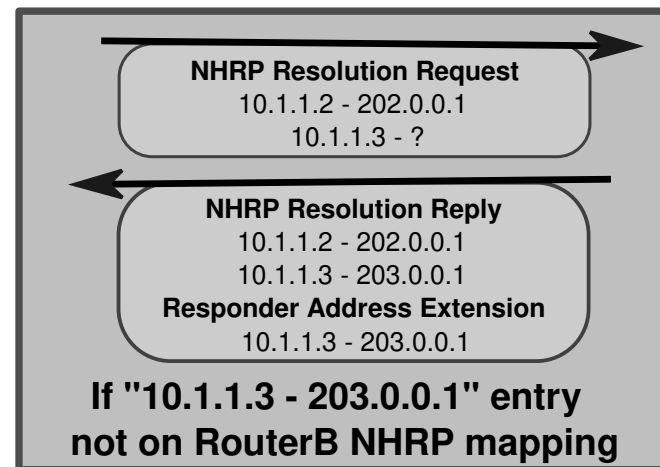
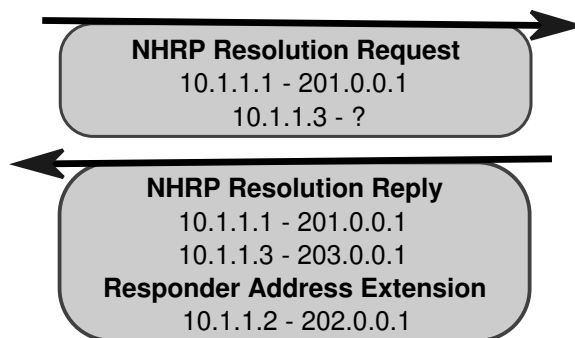
NHRP mapping
10.1.1.2 - 202.0.0.1 (static)



NHRP mapping
10.1.1.1 - 201.0.0.1 (static)

NHRP mapping
10.1.1.1 - 201.0.0.1 (static)
10.1.1.2 - 202.0.0.1 (dynamic)

NHRP mapping
10.1.1.2 - 202.0.0.1 (static)
10.1.1.3 - 203.0.0.1 (dynamic)



Hub-Spoke vs. Spoke-Spoke

- Hub-Spoke

- Each remote site is connected with a point-to-point GRE tunnel to a pre-defined central node (Hub).
- Hub accepts new tunnel connections from Spokes (branches nodes).
- Data communication (over the overlay network) between Spokes is relayed via the Hub.
- Multiple Hubs may exist to provide redundancy.

- Spoke-Spoke

- Individual branch office nodes can dynamically initiate tunnel connections between each other, bypassing the Hub node.
- Data communication (over the overlay network) can be direct between Spokes.
- Dynamic IGP routing protocols may operate between Spoke and Hubs, but not between Spokes.
- No interoperability with non-Cisco IOS routers. (?)



IPSec

- Framework of security protocols and algorithms used to secure data at the network layer
- Authentication Header (AH)
 - ♦ Ensures data integrity
 - ♦ Does not provide confidentiality
 - ♦ Provides origin authentication
 - ♦ Uses Keyed-hash mechanisms
- Encapsulating Security Payload (ESP)
 - ♦ Provides data confidentiality (encryption)
 - ♦ Data Integrity
 - ♦ Does not protect IP header
- AH and ESP use symmetric secret key algorithms, although public key algorithms are feasible



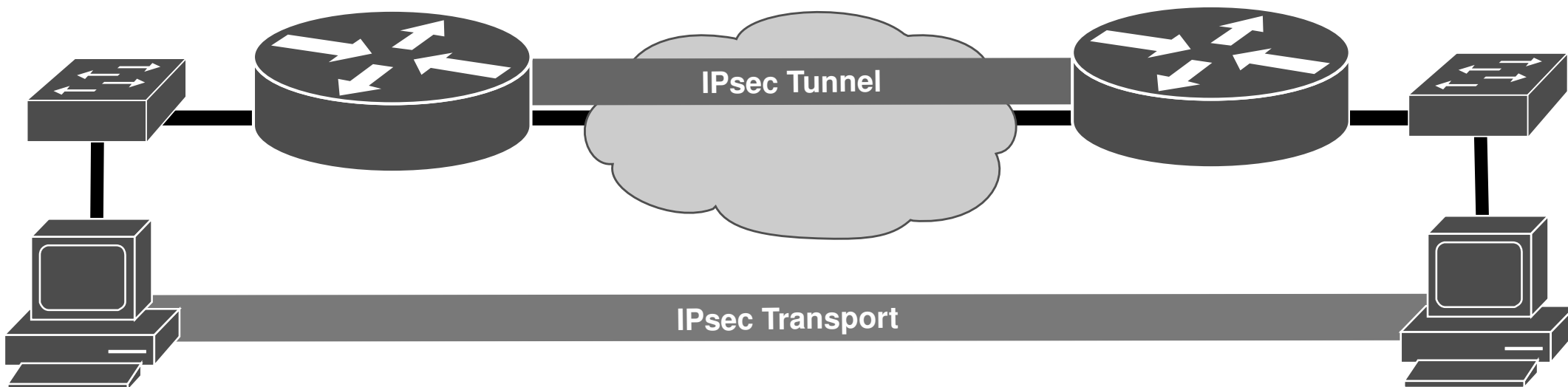
IPSec Modes

- Tunnel

- IPSec gateways provide IPSec services to other hosts in peer-to-peer tunnels
- End-hosts are not aware of IPSec being used to protect their traffic
- IPSec gateways provide transparent protection over untrusted networks

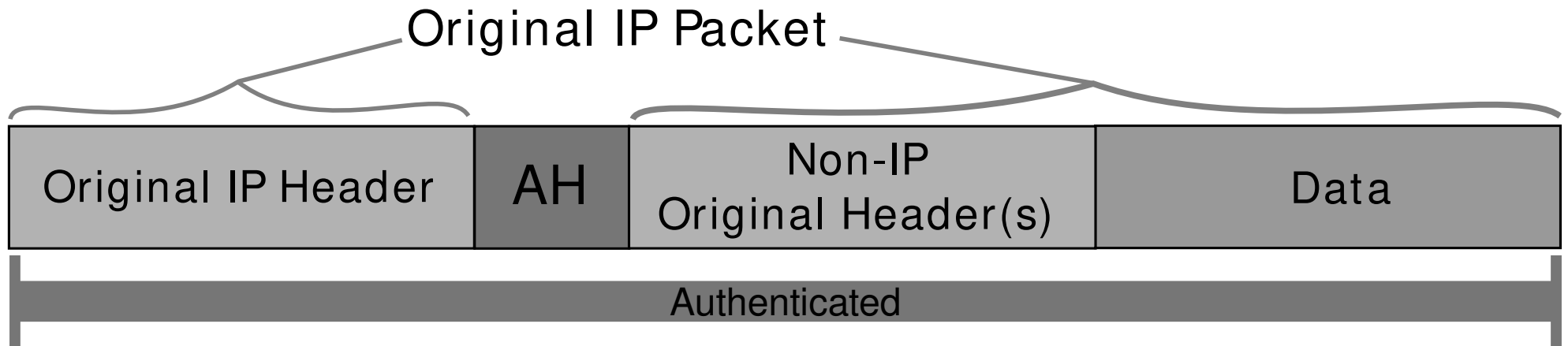
- Transport

- Each end host does IPSec encapsulation of its own data, host-to-host.
- IPSec has to be implemented on end-hosts
- The application endpoint must also be the IPSec endpoint

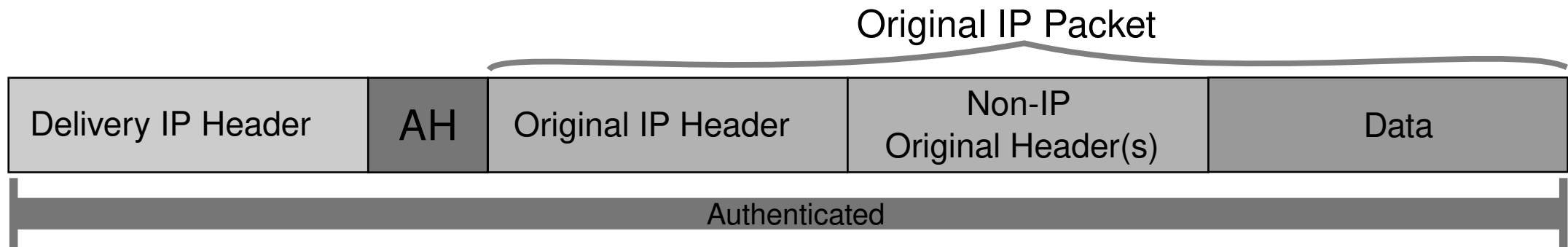


IPSec - AH header placement

- Transport mode

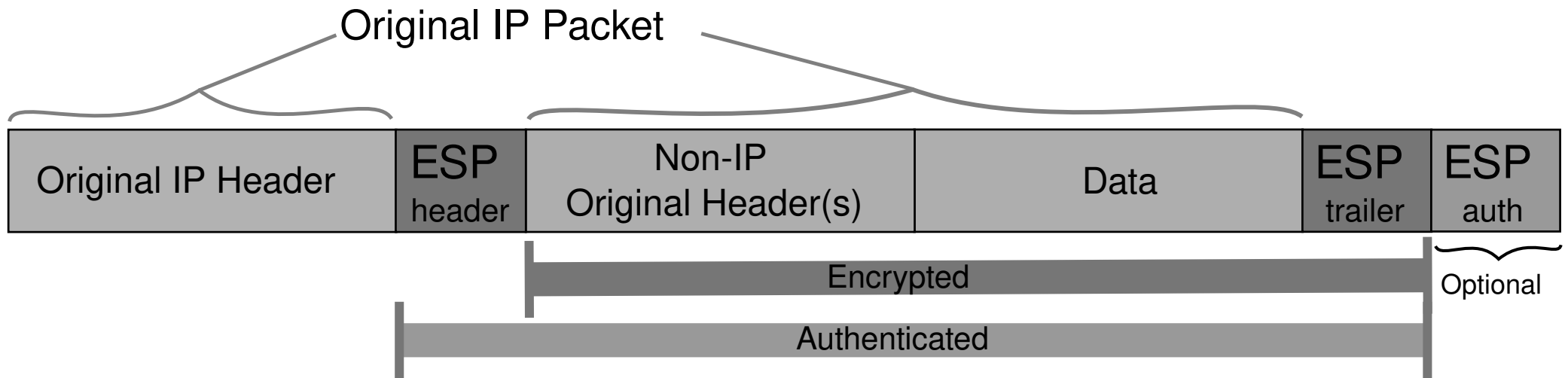


- Tunnel mode

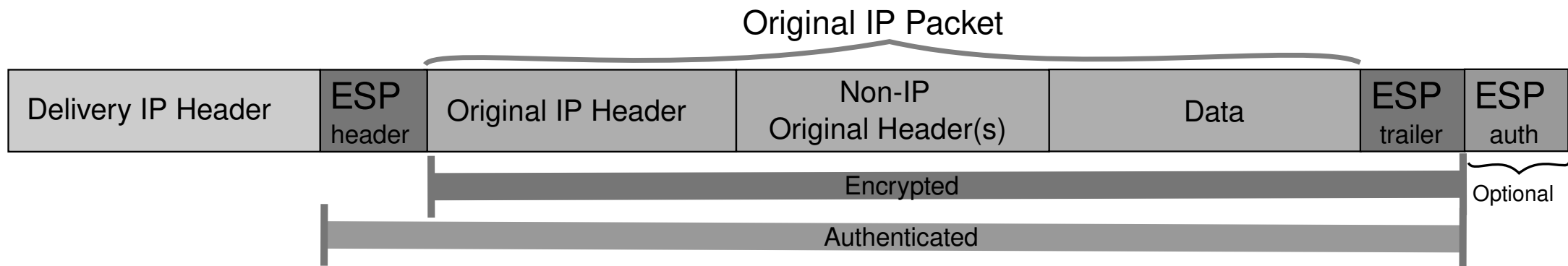


IPSec - ESP header placement

- Transport mode



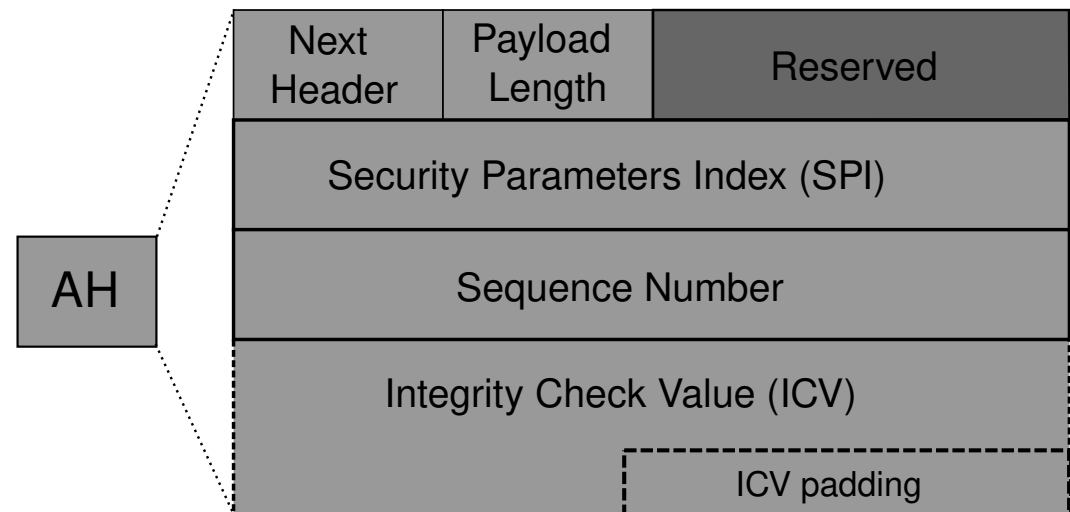
- Tunnel mode



IPsec AH Header

- Contains five mandatory fields:

- ◆ The Next Header field is an 8-bit field that identifies the type of the next payload after the AH.
- ◆ The Payload Length is an 8-bit field specifying the length of the header (excluding the first 8 bytes) in 4-byte units.
- ◆ The SPI field contains the negotiated outbound IPsec SPI and is used by the remote peer to identify the SA to which the packet belongs.
- ◆ The Sequence Number field is a 32-bit field that contains a counter value that increases by one for each sent packet (using the same outbound IPsec SA).
- ◆ The ICV field has a variable length (multiple of 32 bits) that contains the output of the authentication hash function (or HMAC based on symmetric encryption algorithms) applied to data/headers under protection.
 - May include padding to ensure that the overall length of the AH header is a multiple of 32 bits in IPv4 or 64 bits in IPv6.



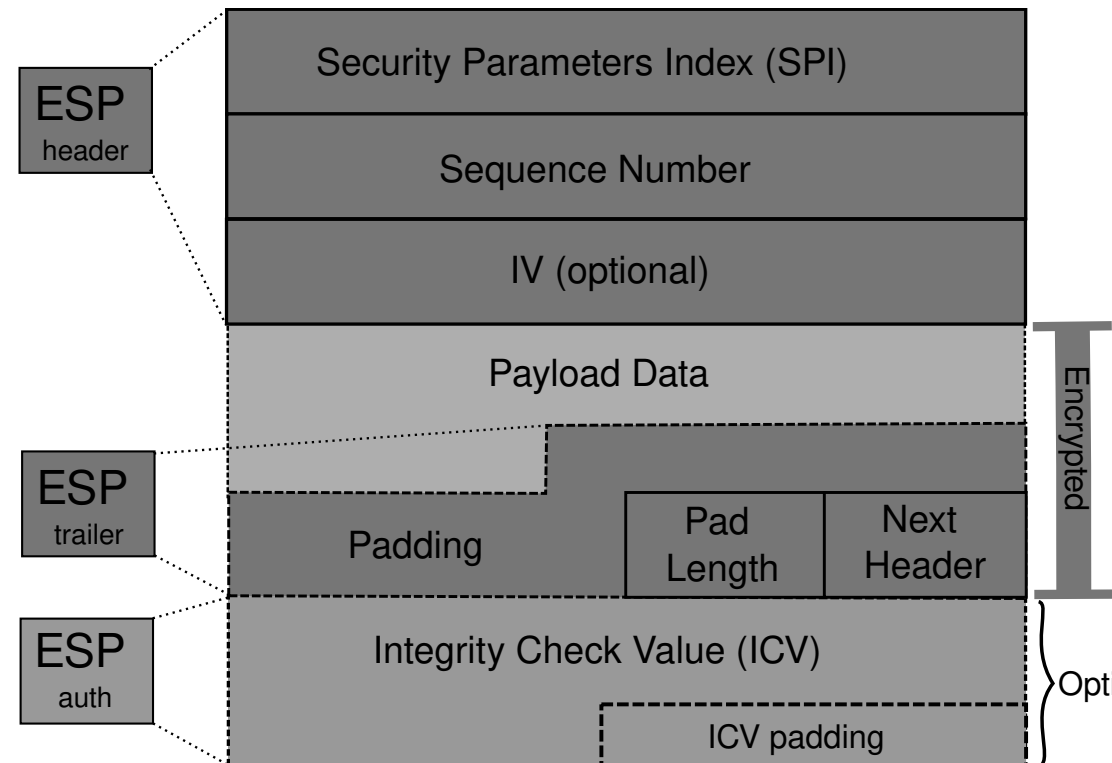
IPsec ESP Header and Trailer

- Contain five mandatory fields:

- ◆ The SPI field contains the negotiated outbound IPsec SPI and is used by the remote peer to identify the SA to which the packet belongs.
- ◆ The Sequence Number field is a 32-bit field that contains a counter value that increases by one for each sent packet (using the same outbound IPsec SA).
- ◆ The Padding field may contain 0 to 255 zero-bytes to guarantee: (i) a specific payload size imposed by the encryption algorithm (e.g., size multiple of the block cipher size), and (ii) that the Pad Length and Next header fields are right aligned within a 4-byte word.
- ◆ The Pad Length is an 8-bit field that indicates the number of padding bytes in the Padding field.
- ◆ The Next Header is an 8-bit field that identifies the type of data contained in the payload data.

- May contain two optional fields:

- ◆ When the encryption algorithm requires an explicit Initialization Vector (IV), this value is sent using the IV field.
 - Some algorithm modes combine encryption and integrity into a single operation.
- ◆ The ICV field has a variable length that contains the output of the authentication hash function (or HMAC based on symmetric encryption algorithms) applied to ESP header, Payload Data, and ESP trailer fields.
 - The ICV field may include padding.



IPSec - Security Associations

- SAs represent a policy contract between two peers or hosts
- Describe how the peers will use IPSec security services to protect network traffic
- An SA contains the following security parameters:
 - ♦ Authentication/encryption algorithm, key length and other encryption parameters (e.g. key lifetime, ...)
 - ♦ Session keys for authentication, or HMACs, and encryption, which can be entered manually or negotiated automatically
 - ♦ A specification of network traffic to which the SA will be applied (e.g. IP traffic or only TELNET sessions)
 - ♦ IPSec AH or ESP encapsulation protocol and tunnel or transport mode



Establishing SA and Cryptographic Keys

- ISAKMP - Internet Security Association and Key Management Protocol
 - Used to establishing Security Associations (SA) and cryptographic keys
 - Separate the details of security association management (and key management) from the details of key exchange
 - Provides a framework for authentication and key exchange but does not define them
- Oakley Key Determination Protocol
 - Key-agreement protocol
 - Allows authenticated peers to exchange keying material across an insecure connection
 - Uses Diffie-Hellman
- SKEME
 - Key exchange protocol
- IKE - Internet Key Exchange
 - Is a hybrid protocol
 - Uses part of Oakley and part of SKEME in conjunction with ISAKMP



IKE/ISAKMP and IPsec

- Enhances IPsec by providing additional features and flexibility
- Provides authentication of the IPsec peers, negotiates IPsec keys, and negotiates IPsec security associations
- The IKE tunnel protects the SA negotiations. After the SAs are in place, IPsec protects data transference
- Advantages
 - Eliminates the need to manually specify IPsec security parameters at both peers
 - Allows administrators to specify a lifetime for the IPsec security association
 - Allows encryption keys to change during IPsec sessions
 - Allows IPsec to provide anti-replay services
 - Permits certification authority (CA) support for a manageable, scalable IPsec implementation
 - Allows dynamic authentication of peers
- IKE/ISAKMP provides three methods for two-way authentication:
 - Pre-shared key (PSK),
 - Digital signatures (RSA-SIG),
 - Public key encryption (RSA-ENC).



ISAKMP and IPsec – Phases/Modes

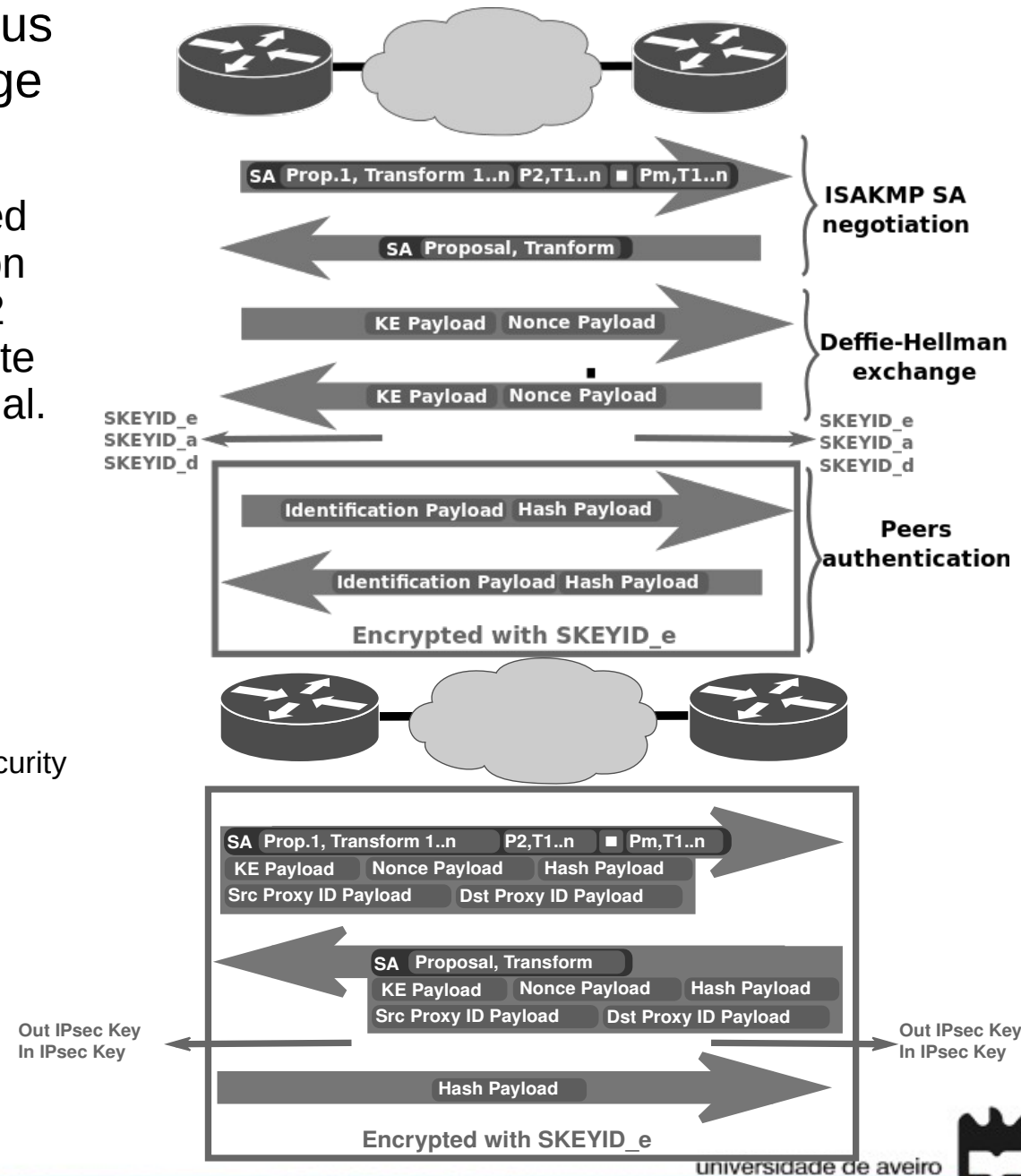
- ISAKMP modes control an efficiency versus security tradeoff during initial key exchange

Phase 1

- Peer agree on a set of parameters to be used to authenticate peers and to encrypt a portion of the phase 1 exchanges and all of phase 2 exchanges, authenticate peers, and generate keys to be used as generating keying material.
- Main mode
 - Requires six packets back and forth
 - Provides complete security during the establishment of an IPsec connection
 - Aggressive mode is an alternative to main mode
 - Uses half the exchanges, but provides less security because some information is transmitted in cleartext

Phase 2 - Quick mode

- Peers negotiate and agree on parameters required to establish a fully functional IPsec communication service.



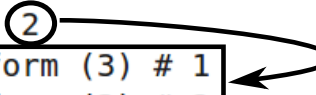
IPsec Packet Exchange

No.	Time	Source	Destination	Protocol	Length	Info
12	12.259744000	2001:a:a::2	2001:a:a::1	ISAKMP	146	Identity Protection (Main Mode)
13	12.293700000	2001:a:a::1	2001:a:a::2	ISAKMP	146	Identity Protection (Main Mode)
14	12.330320000	2001:a:a::2	2001:a:a::1	ISAKMP	298	Identity Protection (Main Mode)
15	12.364351000	2001:a:a::1	2001:a:a::2	ISAKMP	318	Identity Protection (Main Mode)
16	12.481540000	2001:a:a::2	2001:a:a::1	ISAKMP	170	Identity Protection (Main Mode)
17	12.496192000	2001:a:a::1	2001:a:a::2	ISAKMP	138	Identity Protection (Main Mode)
18	12.542122000	2001:a:a::2	2001:a:a::1	ISAKMP	250	Quick Mode
19	12.556571000	2001:a:a::1	2001:a:a::2	ISAKMP	250	Quick Mode
20	12.582568000	2001:a:a::2	2001:a:a::1	ISAKMP	114	Quick Mode
21	15.425134000	2001:a:a::2	2001:a:a::1	ESP	322	ESP (SPI=0xb26693bc)
22	15.440166000	2001:a:a::1	2001:a:a::2	ESP	202	ESP (SPI=0x328b3017)

▸ Frame 21: 322 bytes on wire (2576 bits), 322 bytes captured (2576 bits) on interface 0
 ▸ Ethernet II, Src: c2:04:62:06:00:00 (c2:04:62:06:00:00), Dst: ca:06:73:90:00:08 (ca:06:73:90:00:08)
 ▸ Internet Protocol Version 6, Src: 2001:a:a::2 (2001:a:a::2), Dst: 2001:a:a::1 (2001:a:a::1)
 ▾ Encapsulating Security Payload
 ESP SPI: 0xb26693bc (2993066940)
 ESP Sequence: 10

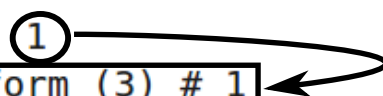
ISAKMP (phase 1) First Message

```
▷ Internet Protocol Version 4, Src: 200.2.2.2 (200.2.2.2), Dst: 200.1.1.1 (200.1.1.1)
▷ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▽ Internet Security Association and Key Management Protocol
  Initiator SPI: 06ba66b161c0b75d
  Responder SPI: 0000000000000000
  Next payload: Security Association (1)
▷ Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
▷ Flags: 0x00
  Message ID: 0x00000000
  Length: 204
▽ Type Payload: Security Association (1)
  Next payload: Vendor ID (13)
  Payload length: 96
  Domain of interpretation: IPSEC (1)
▷ Situation: 00000001
▽ Type Payload: Proposal (2) # 1
  Next payload: NONE / No Next Payload (0)
  Payload length: 84
  Proposal number: ①
  Protocol ID: ISAKMP (1)
  SPI Size: 0
  Proposal transforms: ②
    ▷ Type Payload: Transform (3) # 1
    ▷ Type Payload: Transform (3) # 2
▷ Type Payload: Vendor ID (13) : RFC 3947 Negotiation of NAT-Traversal in the IKE
▷ Type Payload: Vendor ID (13) : draft-ietf-ipsec-nat-t-ike-07
▷ Type Payload: Vendor ID (13) : draft-ietf-ipsec-nat-t-ike-03
▷ Type Payload: Vendor ID (13) : draft-ietf-ipsec-nat-t-ike-02\n
```



ISAKMP (phase 1) Second Message

```
▸ Internet Protocol Version 4, Src: 200.1.1.1 (200.1.1.1), Dst: 200.2.2.2 (200.2.2.2)
▸ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▾ Internet Security Association and Key Management Protocol
    Initiator SPI: 06ba66b161c0b75d
    Responder SPI: 48aa62bdc19e9e3
    Next payload: Security Association (1)
▸ Version: 1.0
    Exchange type: Identity Protection (Main Mode) (2)
▸ Flags: 0x00
    Message ID: 0x00000000
    Length: 104
▾ Type Payload: Security Association (1)
    Next payload: Vendor ID (13)
    Payload length: 56
    Domain of interpretation: IPSEC (1)
▸ Situation: 00000001
▾ Type Payload: Proposal (2) # 1
    Next payload: NONE / No Next Payload (0)
    Payload length: 44
    Proposal number: ①
    Protocol ID: ISAKMP (1)
    SPI Size: 0
    Proposal transforms: ①
    ▸ Type Payload: Transform (3) # 1
▸ Type Payload: Vendor ID (13) : RFC 3947 Negotiation of NAT-Traversal in the IKE
```



ISAKMP (phase 1) Third and Fourth Messages

▸ Internet Protocol Version 4, Src: 200.2.2.2 (200.2.2.2), Dst: 200.1.1.1
▸ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▼ Internet Security Association and Key Management Protocol

Initiator SPI: 06ba66b161c0b75d

Responder SPI: 48aa62bdc19e9e3

Next payload: Key Exchange (4)

▸ Version: 1.0

Exchange type: Identity Protection (Main Mode) (2)

▸ Flags: 0x00

Message ID: 0x00000000

Length: 276

▼ Type Payload: Key Exchange (4)

Next payload: Nonce (10)

Payload length: 132

Key Exchange Data: 6b90894c1593b8ddda8d321a05af8075

▼ Type Payload: Nonce (10)

Next payload: Vendor ID (13)

Payload length: 24

Nonce DATA: 21edc1d7ee9a9a51d9d8a0fccc1012ff9d58a348

▸ Type Payload: NAT-D (RFC 3947) (20)

▸ Type Payload: NAT-D (RFC 3947) (20)

▸ Internet Protocol Version 4, Src: 200.1.1.1 (200.1.1.1), Dst: 200.2.2.2
▸ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▼ Internet Security Association and Key Management Protocol

Initiator SPI: 06ba66b161c0b75d

Responder SPI: 48aa62bdc19e9e3

Next payload: Key Exchange (4)

▸ Version: 1.0

Exchange type: Identity Protection (Main Mode) (2)

▸ Flags: 0x00

Message ID: 0x00000000

Length: 296

▼ Type Payload: Key Exchange (4)

Next payload: Nonce (10)

Payload length: 132

Key Exchange Data: 820d0eafec6260bc958a60d1d086e6ec823032774f16c316...

▼ Type Payload: Nonce (10)

Next payload: Vendor ID (13)

Payload length: 24

Nonce DATA: 0f37423fb10f422983fcf0d9dcab26a5b8be59aa

▸ Type Payload: NAT-D (RFC 3947) (20)

▸ Type Payload: NAT-D (RFC 3947) (20)

ISAKMP (phase 1) Fifth and Sixth Messages

▷ Internet Protocol Version 4, Src: 200.2.2.2 (200.2.2.2), Dst: 200.1.1.1
▷ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▽ Internet Security Association and Key Management Protocol

Initiator SPI: 06ba66b161c0b75d

Responder SPI: 48aa62bdcbl9e9e3

Next payload: Identification (5)

▷ Version: 1.0

Exchange type: Identity Protection (Main Mode) (2)

▷ Flags: 0x01

Message ID: 0x00000000

Length: 92

Encrypted Data (64 bytes)

▷ Internet Protocol Version 4, Src: 200.1.1.1 (200.1.1.1), Dst: 200.2.2.2
▷ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▽ Internet Security Association and Key Management Protocol

Initiator SPI: 06ba66b161c0b75d

Responder SPI: 48aa62bdcbl9e9e3

Next payload: Identification (5)

▷ Version: 1.0

Exchange type: Identity Protection (Main Mode) (2)

▷ Flags: 0x01

Message ID: 0x00000000

Length: 68

Encrypted Data (40 bytes)



ISAKMP (phase 2) Message

```
▷ User Datagram Protocol, Src Port: 500 (500), Dst Port: 500 (500)
▽ Internet Security Association and Key Management Protocol
  Initiator SPI: 06ba66b161c0b75d
  Responder SPI: 48aa62bdc19e9e3
  Next payload: Hash (8)
▷ Version: 1.0
  Exchange type: Quick Mode (32)
▷ Flags: 0x01
  Message ID: 0x5277ae21
  Length: 220
  Encrypted Data
```

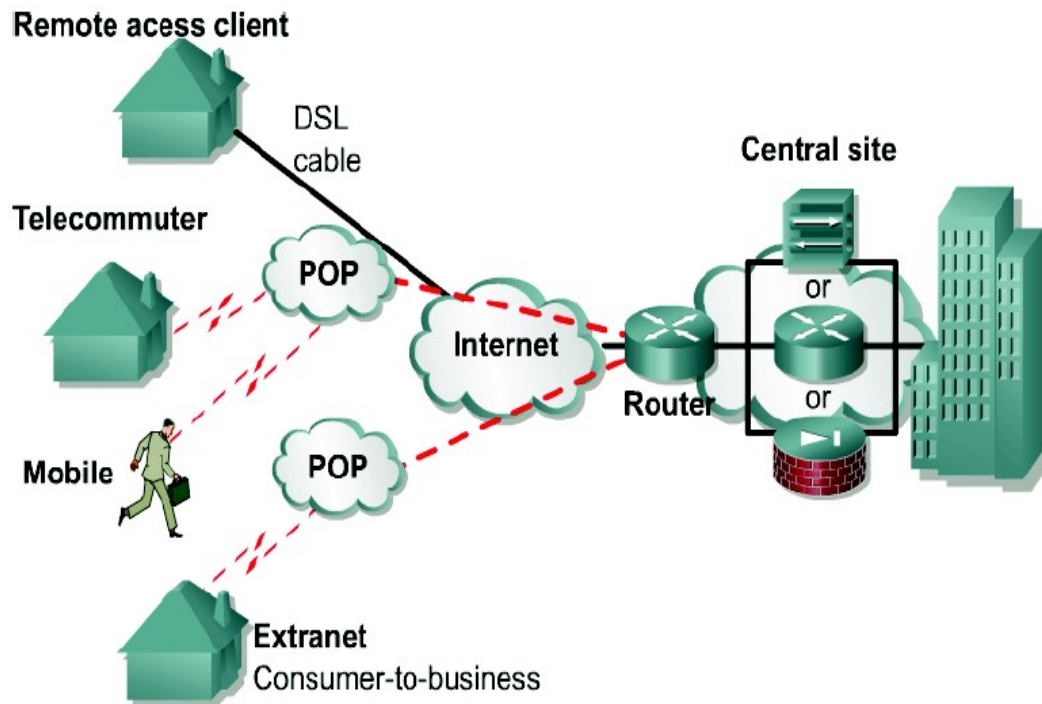


Virtual Private Networks (VPN)

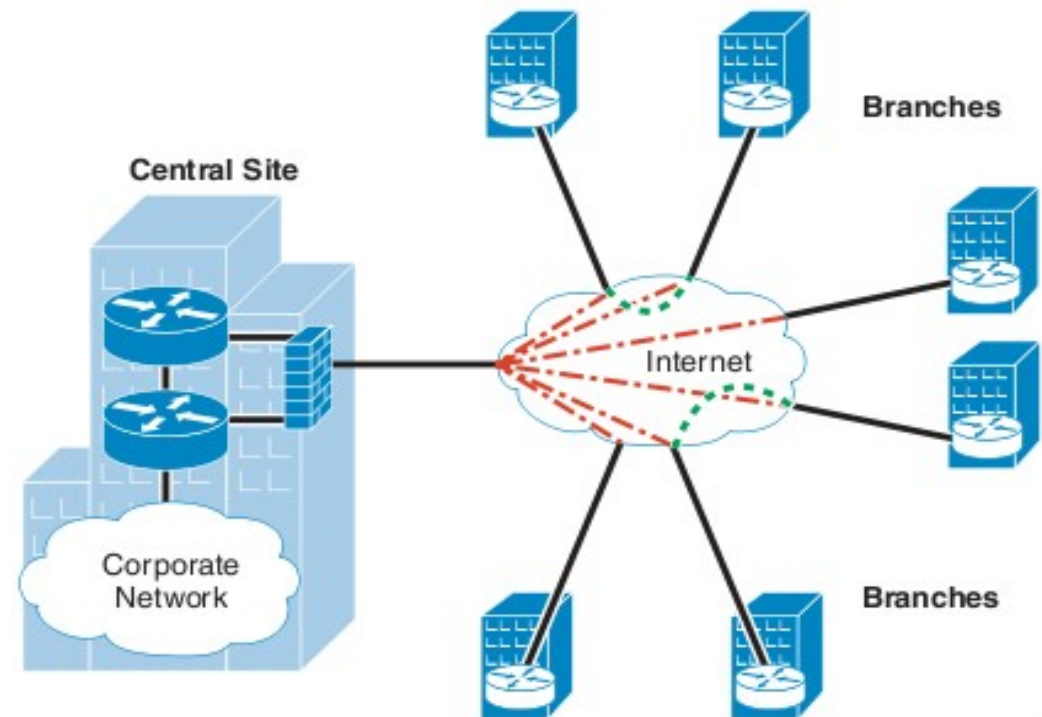


VPN - Virtual Private Networks

- Is an encrypted connection between private networks over a public network



- Remote Access VPN



- Site-to-Site VPN

VPN types

- Remote Access VPN

- ♦ PPTP
- ♦ L2TP/IPsec
- ♦ SSL/TLS VPN
 - Web VPN (client-less SSL VPN) – VPN client can be a standard browser
- ♦ SSH VPN
- ♦ Open VPN

- Site-to-Site VPN

- ♦ IPsec VPN
 - With static or dynamic configuration
- ♦ IPsec + GRE VPN
 - Dynamic Multipoint VPN



Remote Access VPN - PPTP VPN

- Based on PPTP
 - ♦ PPTP packages data within PPP packets
 - ♦ Encapsulates the PPP packets within IP packets
- Uses a form of General Routing Encapsulation (GRE) to get data to and from its final destination
- Supports authentication based on protocols PAP, EAP, CHAP, MS-CHAPv1 and MS-CHAPv2
- Uses MPPE as cipher
 - ♦ Has two different keys (one for each direction)
 - ♦ Requires MS-CHAPv2 authentication
 - ♦ Keys derived from the MS-CHAPv2's password hash and challenges
- PPTP creates a TCP control connection between the VPN client and VPN server to establish a tunnel
 - ♦ Uses TCP port 1723 for these connections
- PPTP can support only one tunnel at a time for each user



Remote Access VPN - L2TP/IPSec VPN

- Authentication can be performed with Digital Certificates (RSA) or with the same PPP authentication mechanisms as PPTP
- Provides data integrity, authentication of origin and replay protection
- Encryption provided by IPSec (ESP protocol)
- Can support multiple, simultaneous tunnels for each user
- Slower performance than PPTP



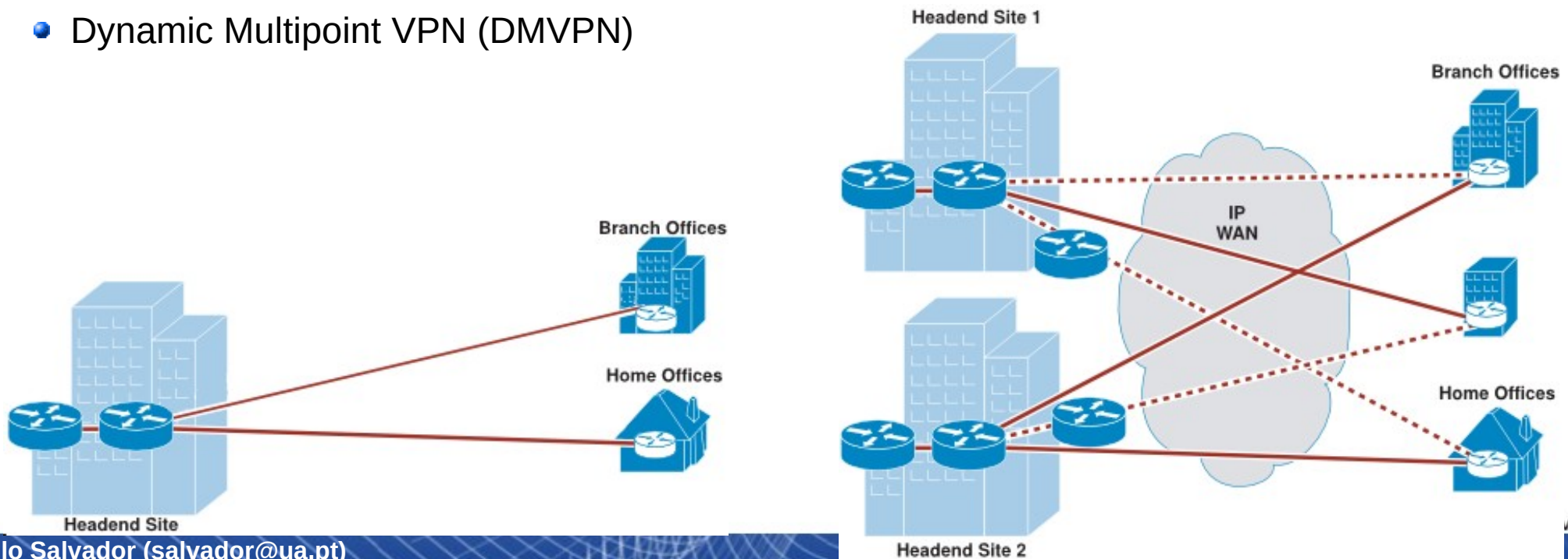
Other Remote Access VPN types

- SSL/TLS VPN
 - ♦ SSL/TLS protocol handles the VPN tunnel creation
 - ♦ SSL/TLS is much easier to implement than IPSec and provides a simple and well-tested platform
 - ♦ RSA handshake (or DH) is used exactly as IKE in IPSec
- SSH VPN
 - ♦ VPN over a SSH connection
 - ♦ SSH tunneling - port forwarding
- OpenVPN
 - ♦ Implements a SSL/TLS VPN
 - ♦ Allows PSK, certificate, and login/password based authentication
 - ♦ Encryption provided by OpenSSL (can use all ciphers available)
 - ♦ Compatible with dynamic and NAT addresses



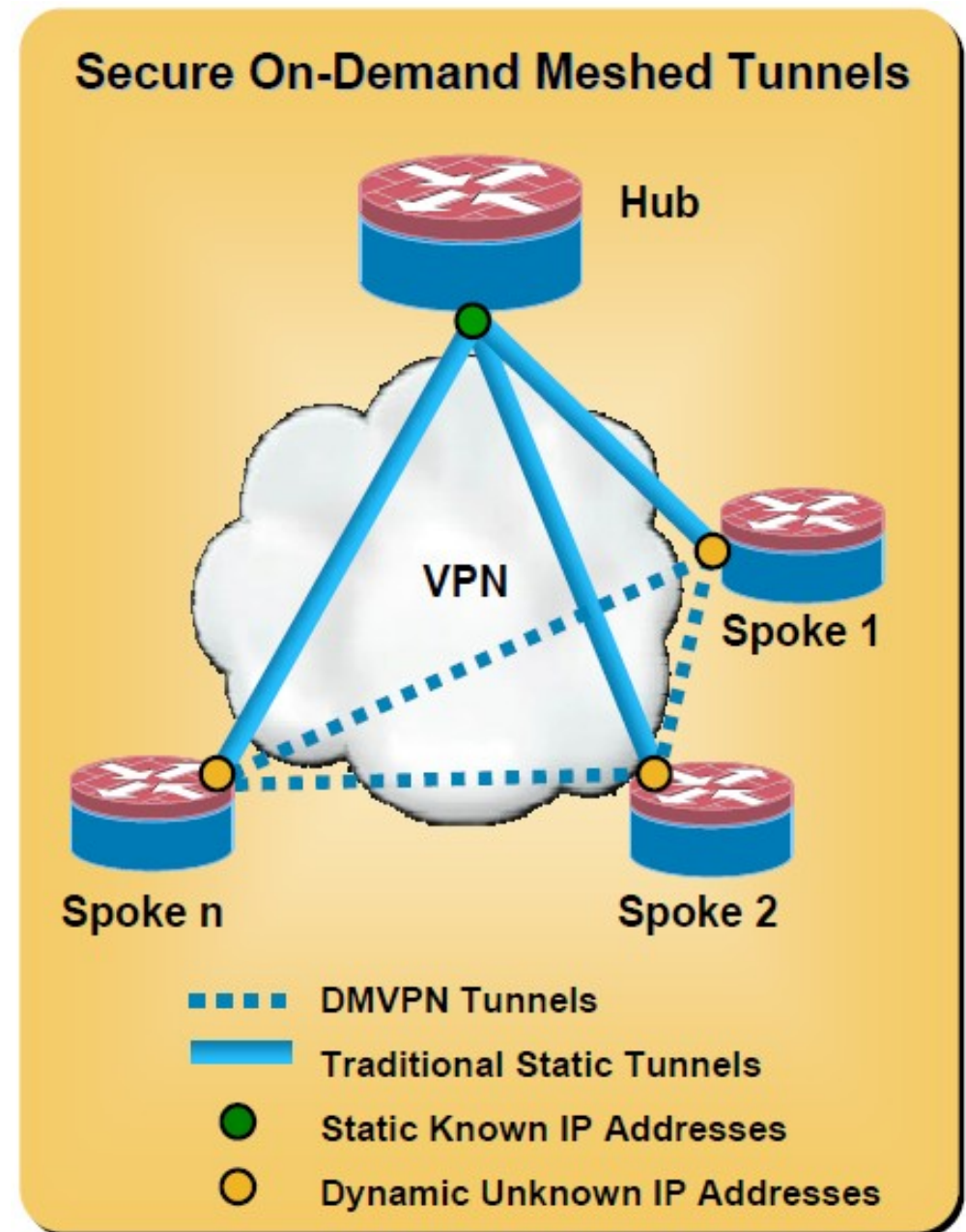
Variants of Site-to-Site IPsec VPN

- IPsec tunnels with static configuration
 - ◆ Requires the knowledge of all peers (IP addresses and security parameters)
 - ◆ High configuration overhead
- IPsec tunnels with dynamic configuration (at the headend/hub)
 - ◆ Hub + spokes configuration
 - ◆ Generic configuration at the headend/hub
 - ◆ Easy to add new spokes
- A basic IPsec tunnel can't protect multicast traffic.
- IPsec + GRE tunnels
 - ◆ Generic Routing Encapsulation (GRE) allows the protection of multicast traffic over IPsec
- Dynamic Multipoint VPN (DMVPN)

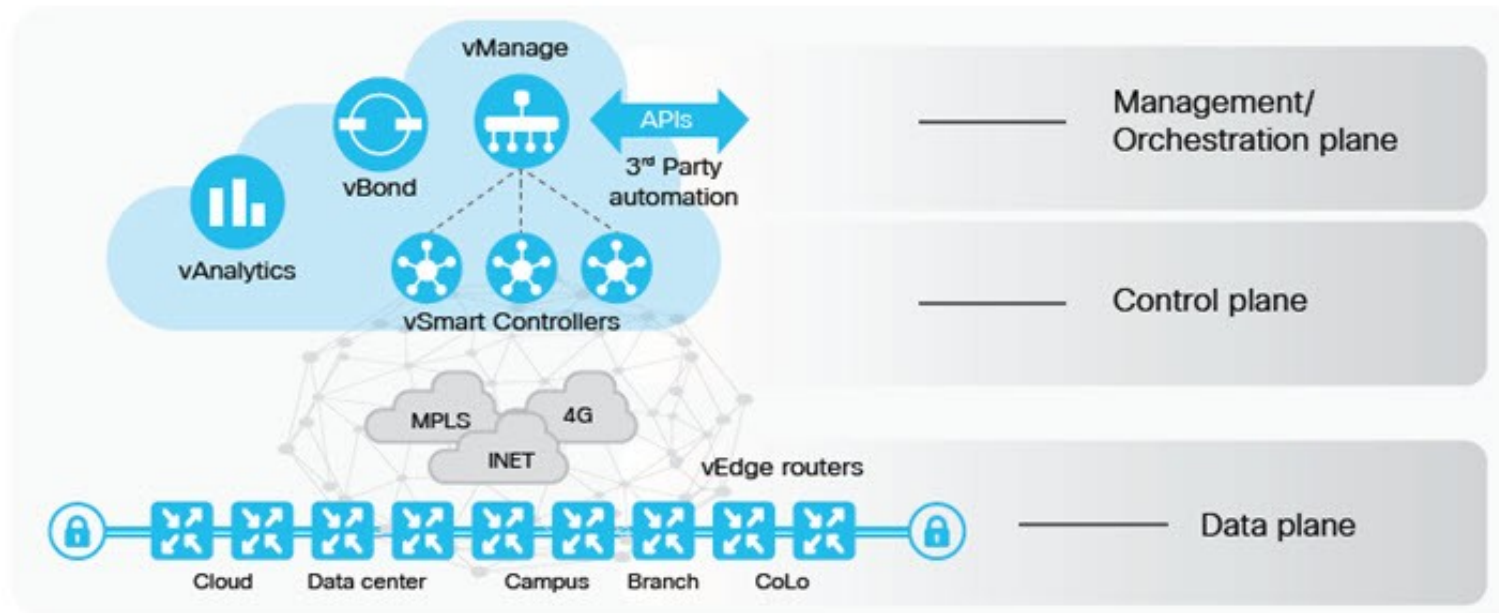


Dynamic Multipoint VPN

- Relies on NHRP to create overlay network
- Provides full meshed connectivity with simple configuration of hub and spoke
- Supports dynamically addressed spokes
- Facilitates zero-touch configuration for addition of new spokes
- Features automatic IPsec triggering for building an IPsec tunnel



SD-WAN



- **Software Defined WAN**

- ◆ Edge Connectivity Abstraction.
- ◆ WAN Virtualization.
- ◆ Policy-Driven, Centralized Management.
- ◆ Elastic Traffic Management.
- ◆ Advantages: Easy deployment and management.
- ◆ Disadvantages: Completely dependence (present and future) on external providers.