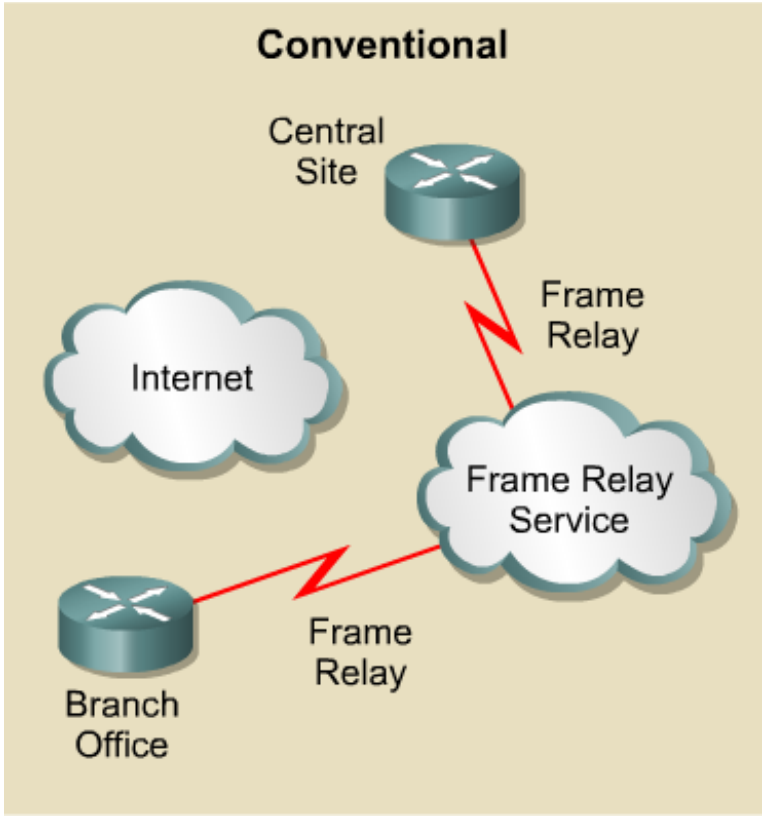
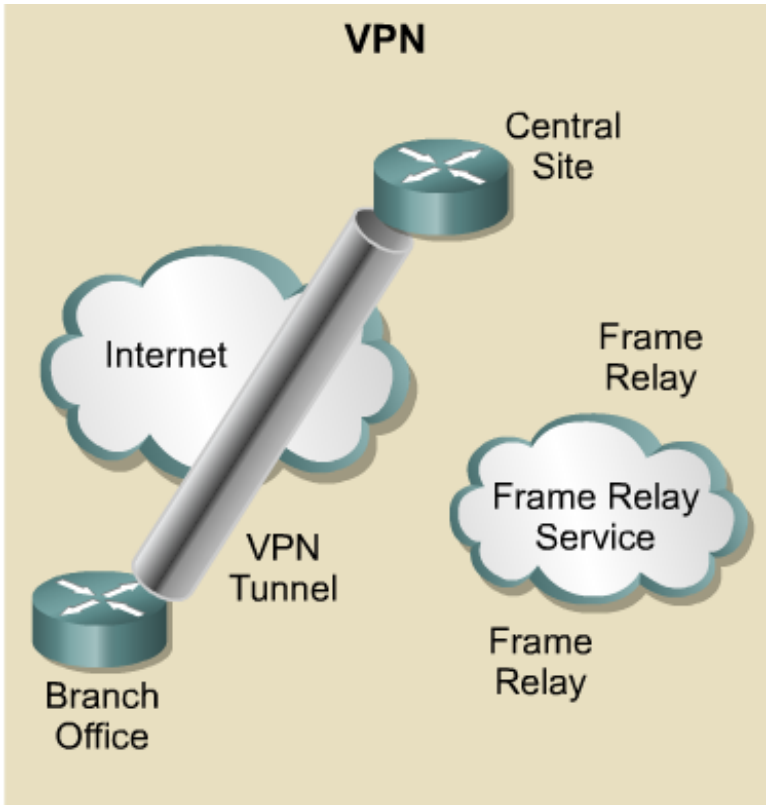
Implementing Virtual Private Networks

Conventional Private Networks



- Higher cost
- Less flexible
- WAN management
- Complex topologies

Virtual Private Networks



- Lower cost
- More flexible
- Simpler management
- Tunnel topology

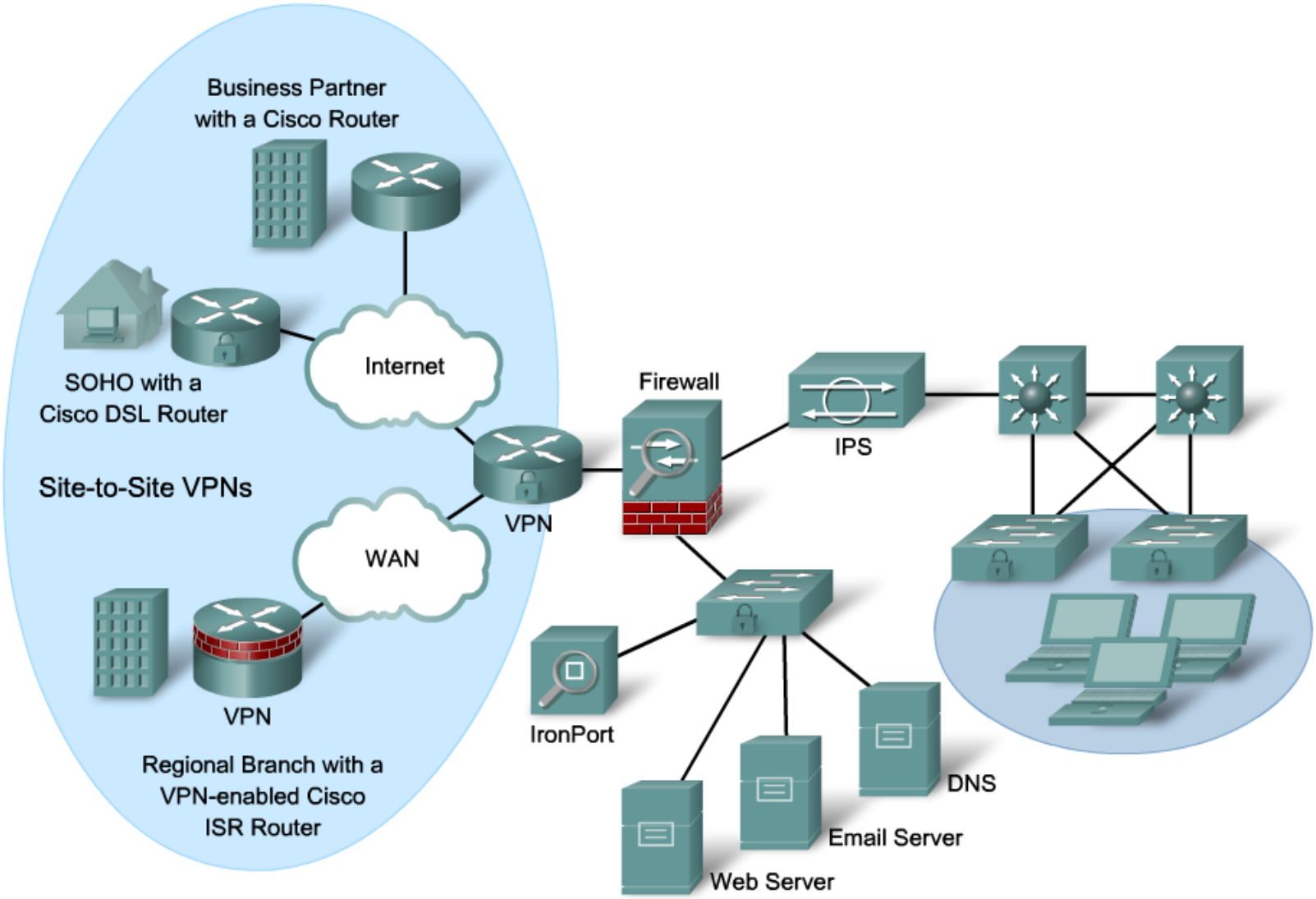
VPNs

- A Virtual Private Network (VPN) provides the same network connectivity for remote users over a public infrastructure as they would have over a private network.
- VPN services for network connectivity include:
  - Authentication
  - Data integrity
  - Confidentiality

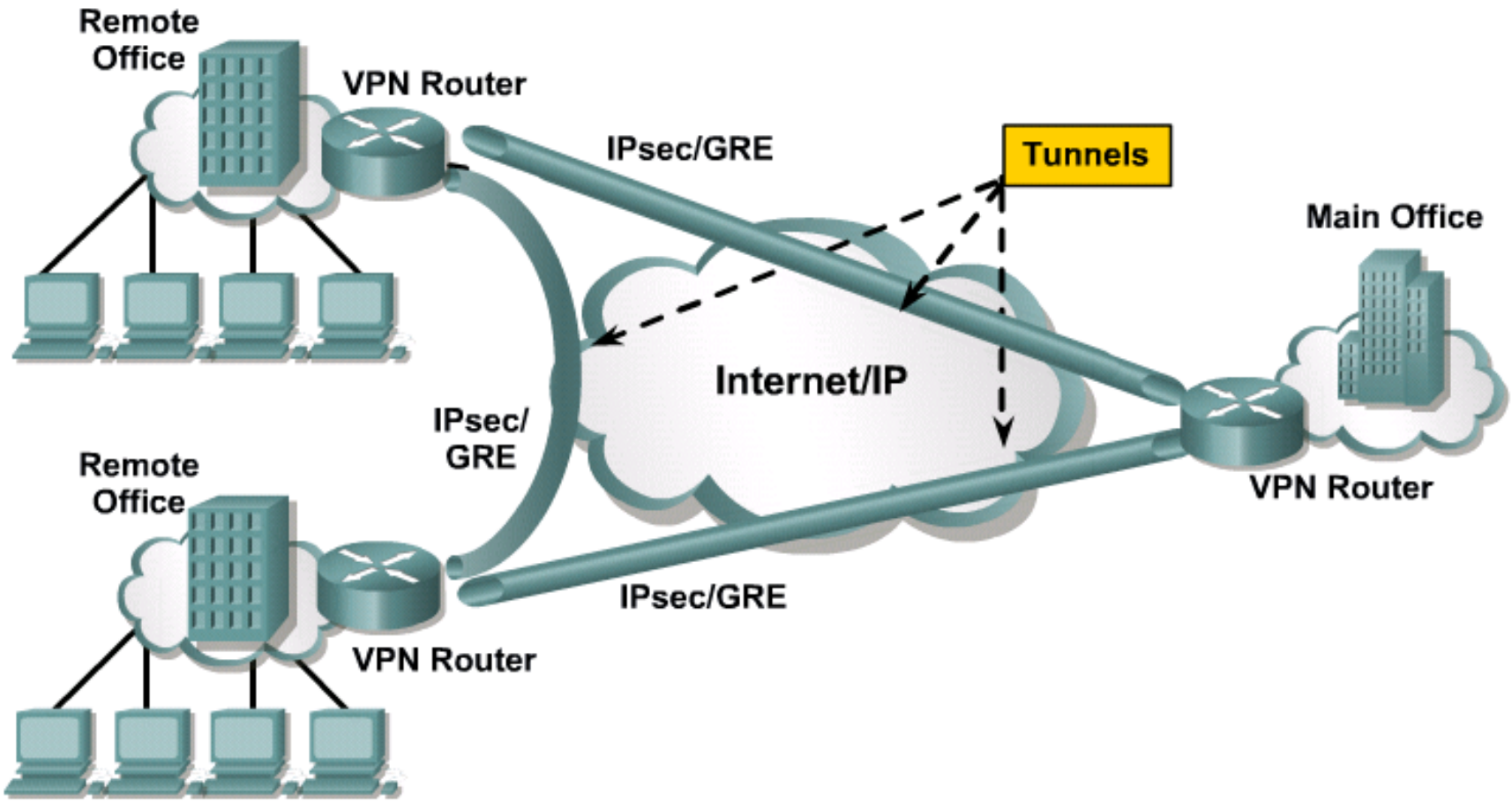
Two Types of VPNs

- Site-to-Site VPNs:
  - Intranet VPNs connect corporate headquarters, remote offices, and branch offices over a public infrastructure.
  - Extranet VPNs link customers, suppliers, partners, or communities of interest to a corporate Intranet over a public infrastructure.
- Remote Access VPNs:
  - Which securely connect remote users, such as mobile users and telecommuters, to the enterprise.

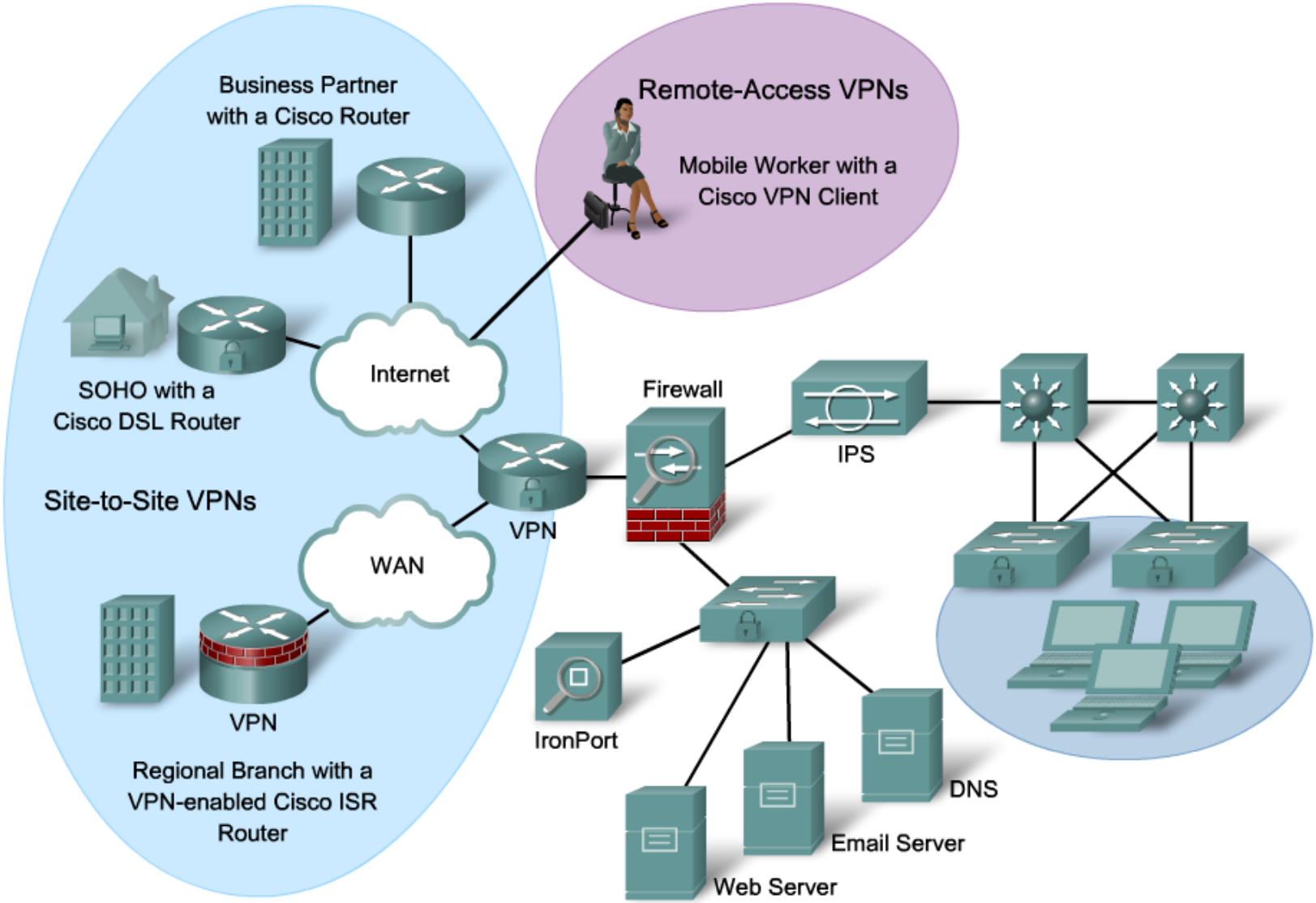
Site-to-Site VPNs



Site-to-Site VPNs



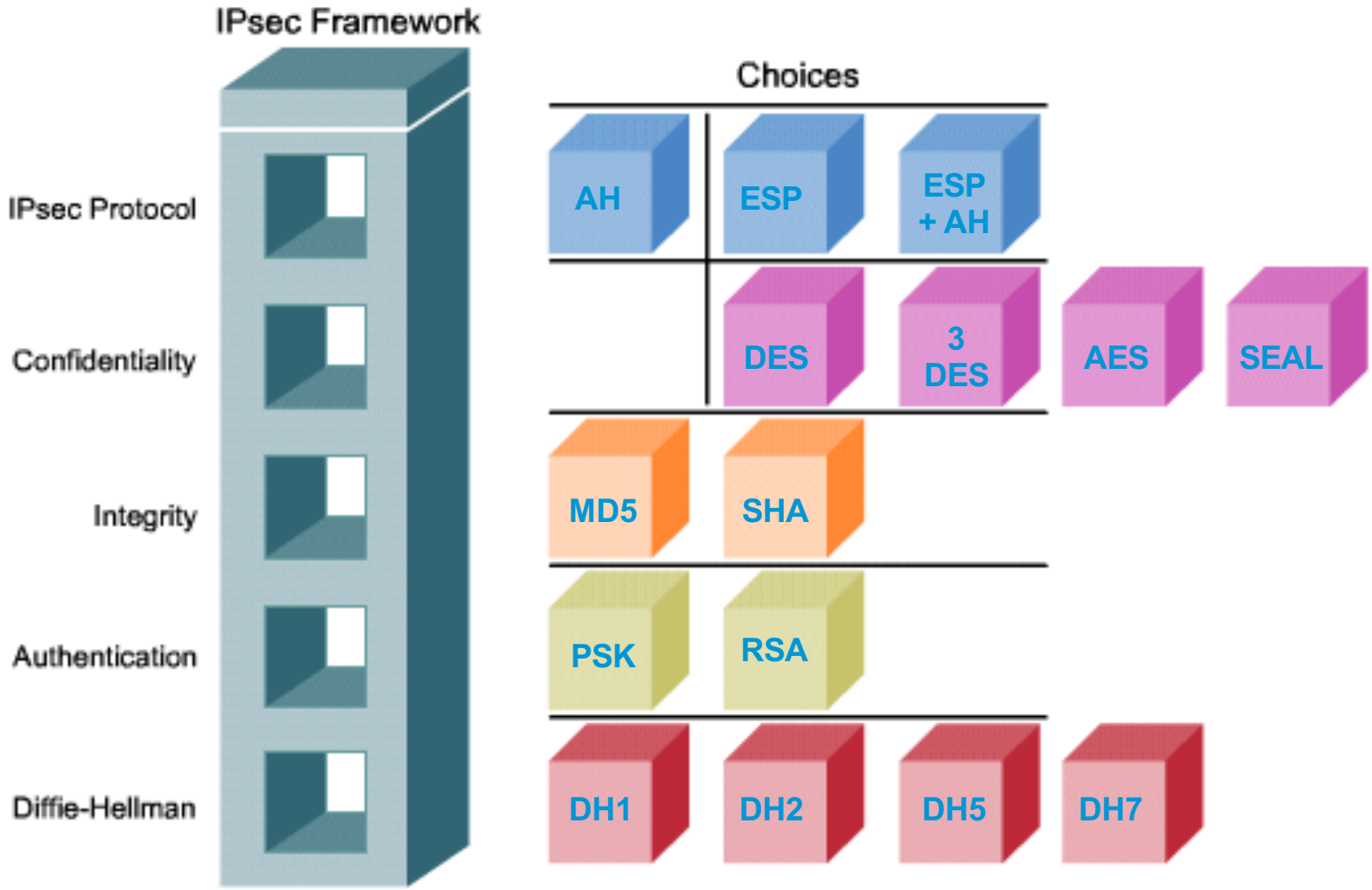
Remote Access VPNs



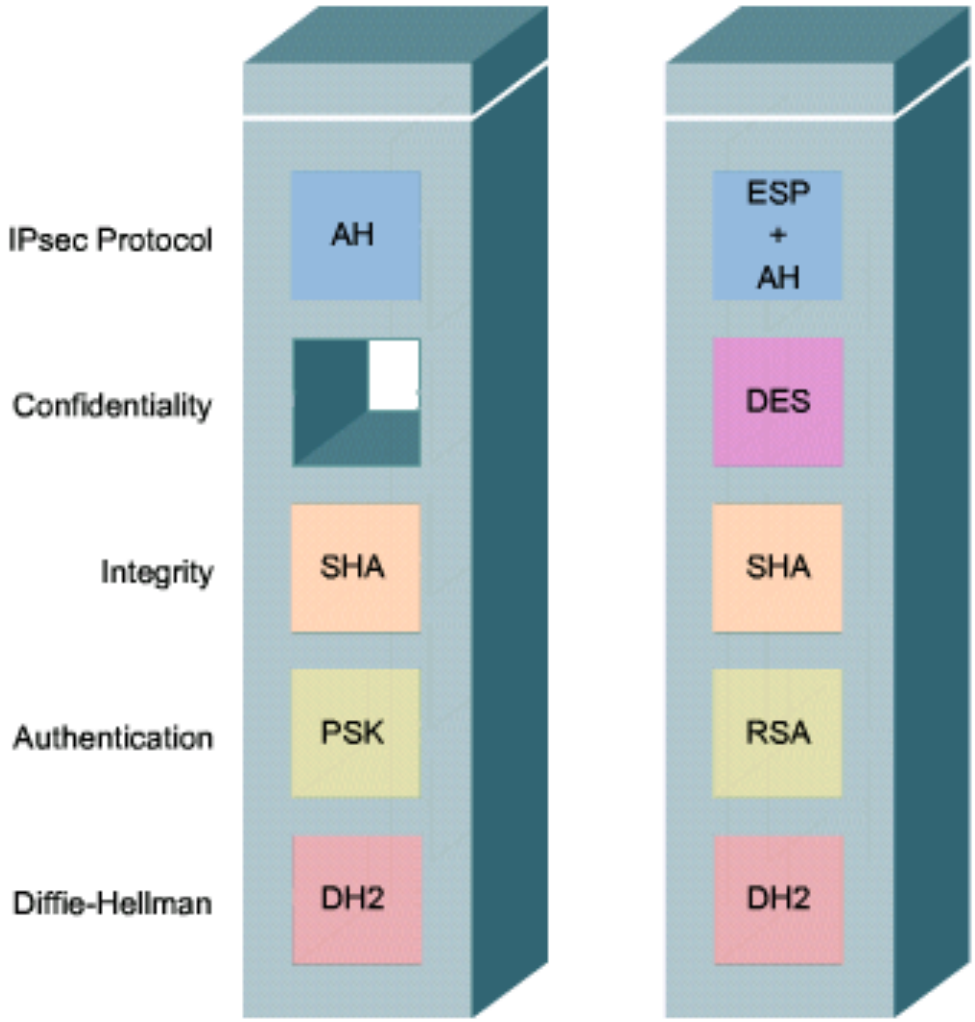
IPsec - Internet Protocol Security

- A “framework” of open standards developed by the IETF to create a secure tunnel at the network (IP) layer.
  - It spells out the rules for secure communications.
  - RFC 2401 - RFC 2412
- IPsec is not bound to any specific encryption or authentication algorithms, keying technology, or security algorithms.
- IPsec allows newer and better algorithms to be implemented without patching the existing IPsec standards.

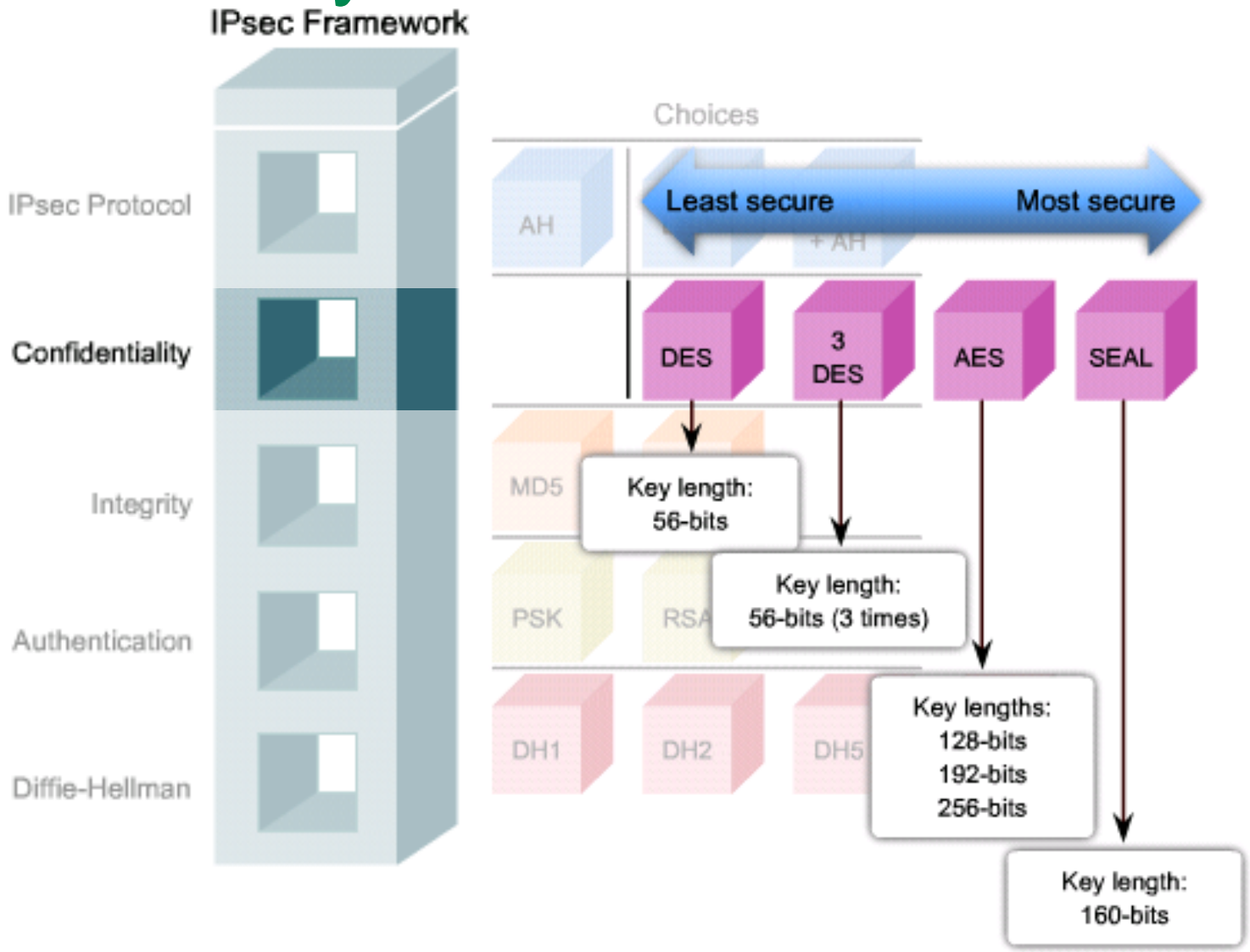
IPsec Protocol Framework



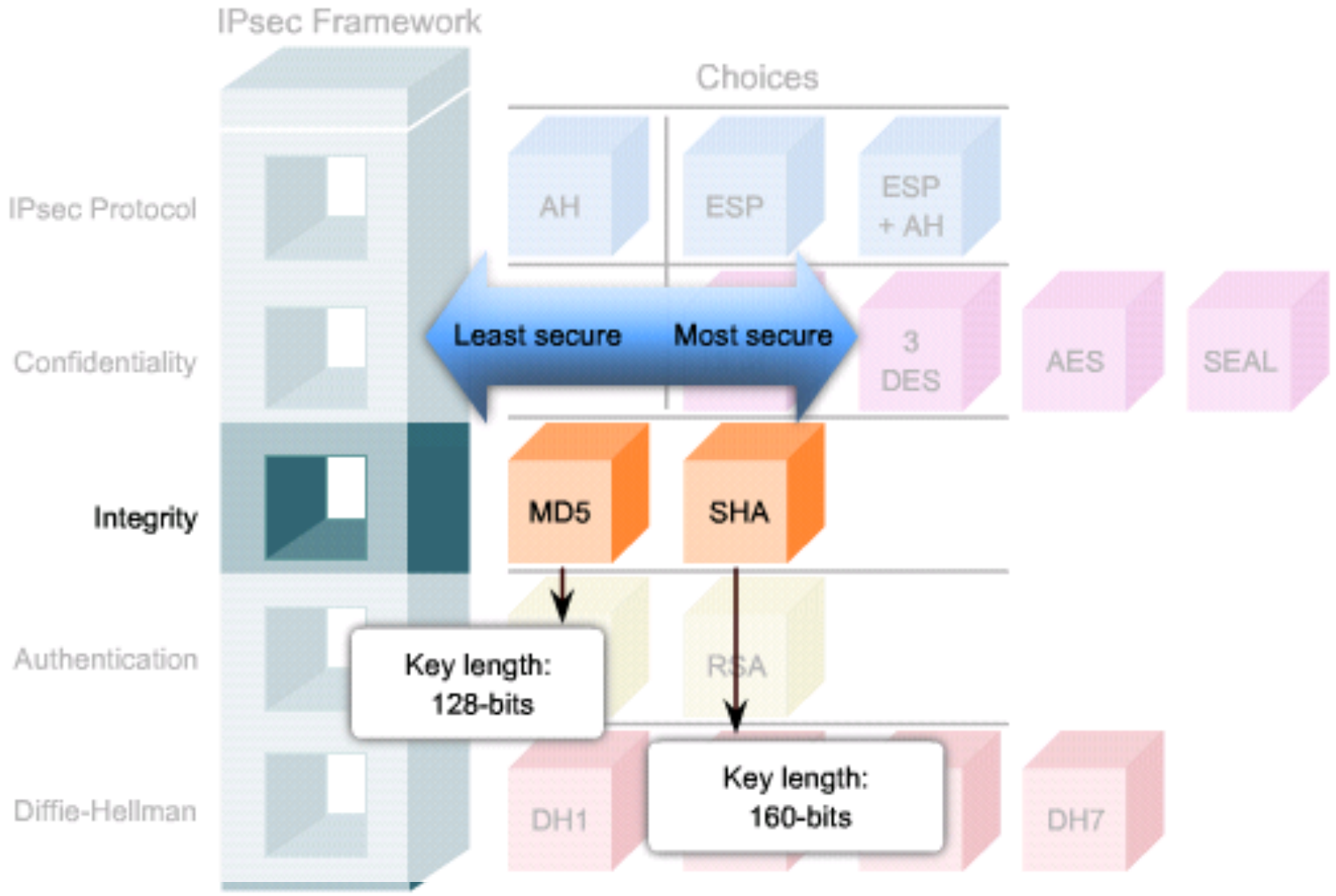
IPsec Protocol Framework



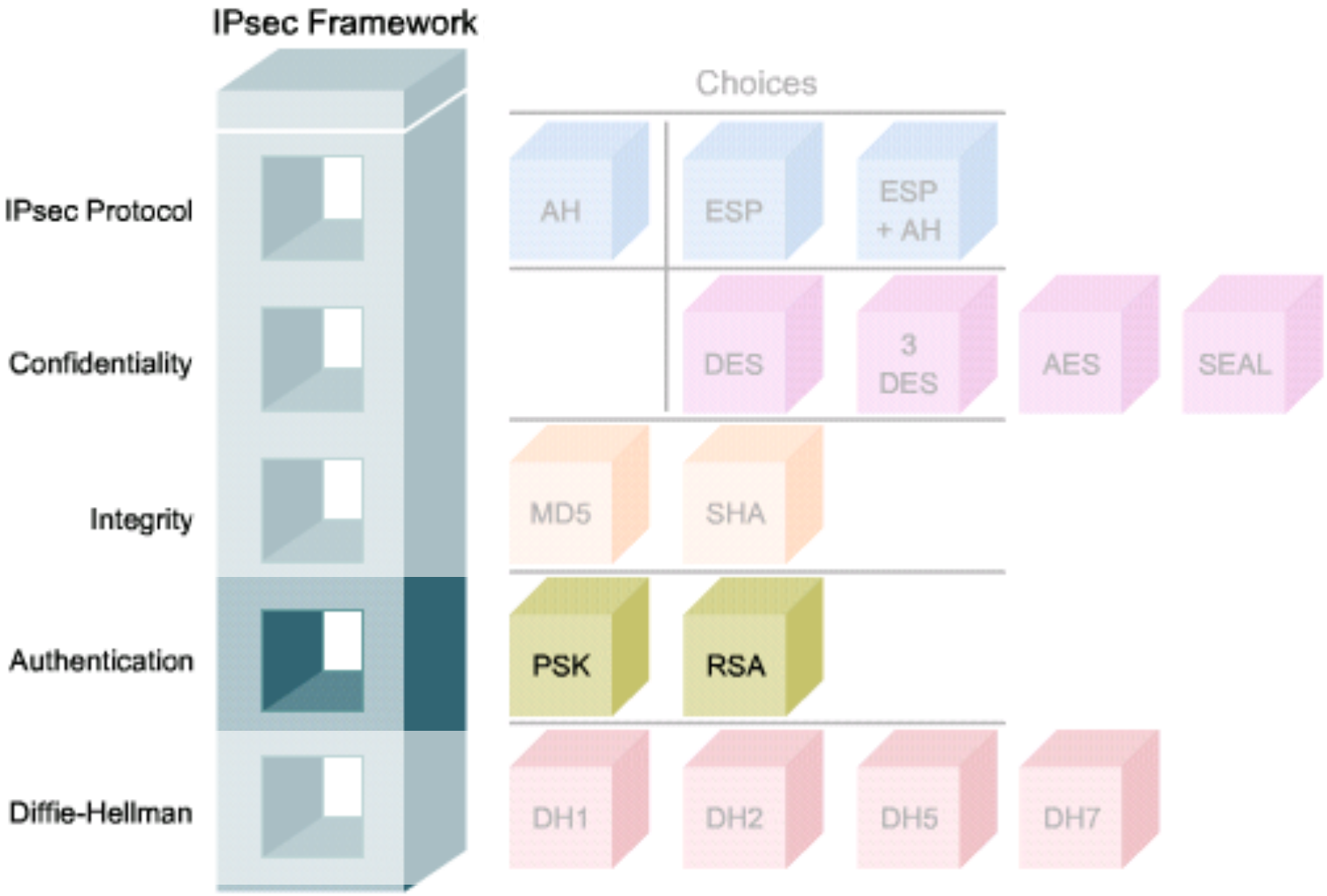
Confidentiality



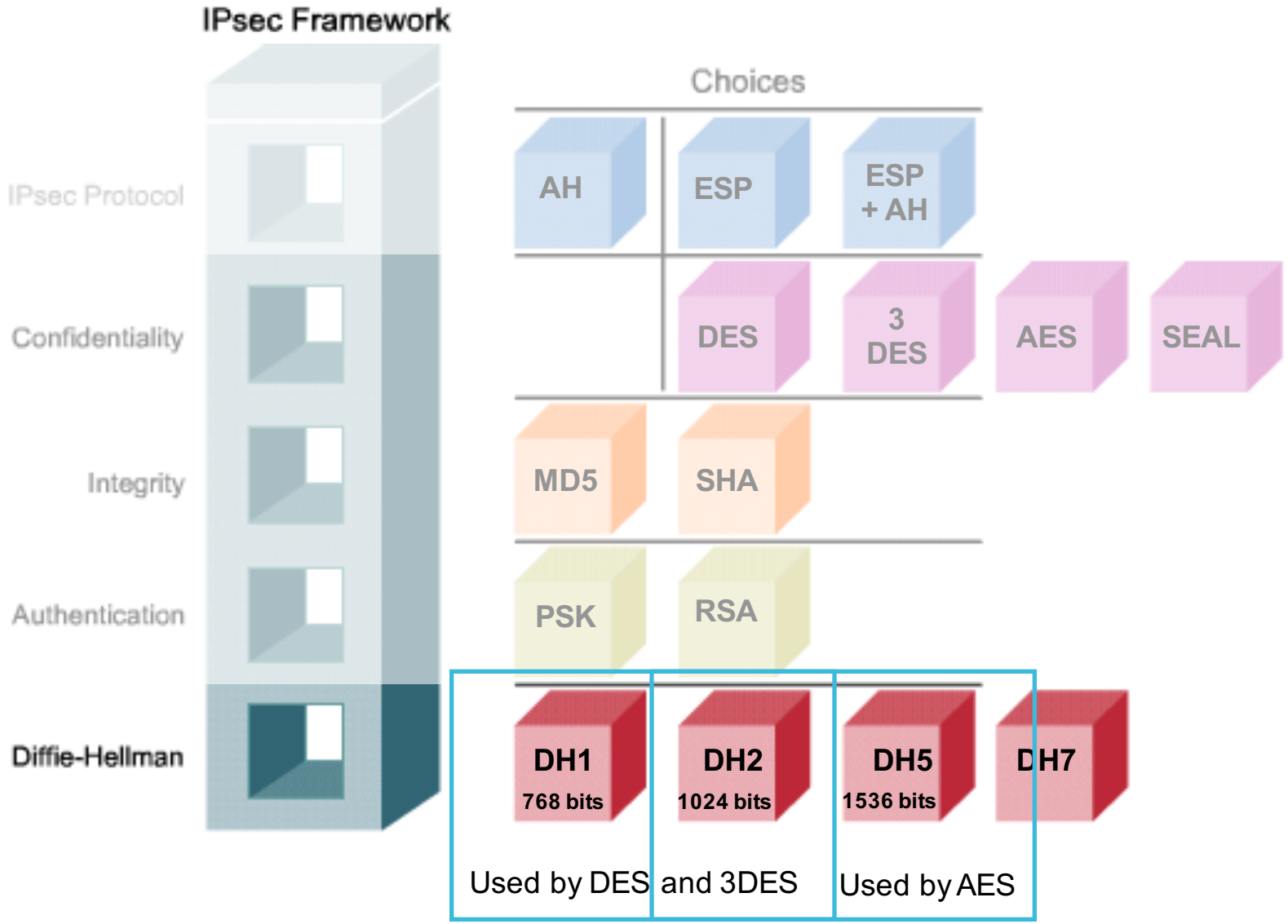
Integrity



Authentication

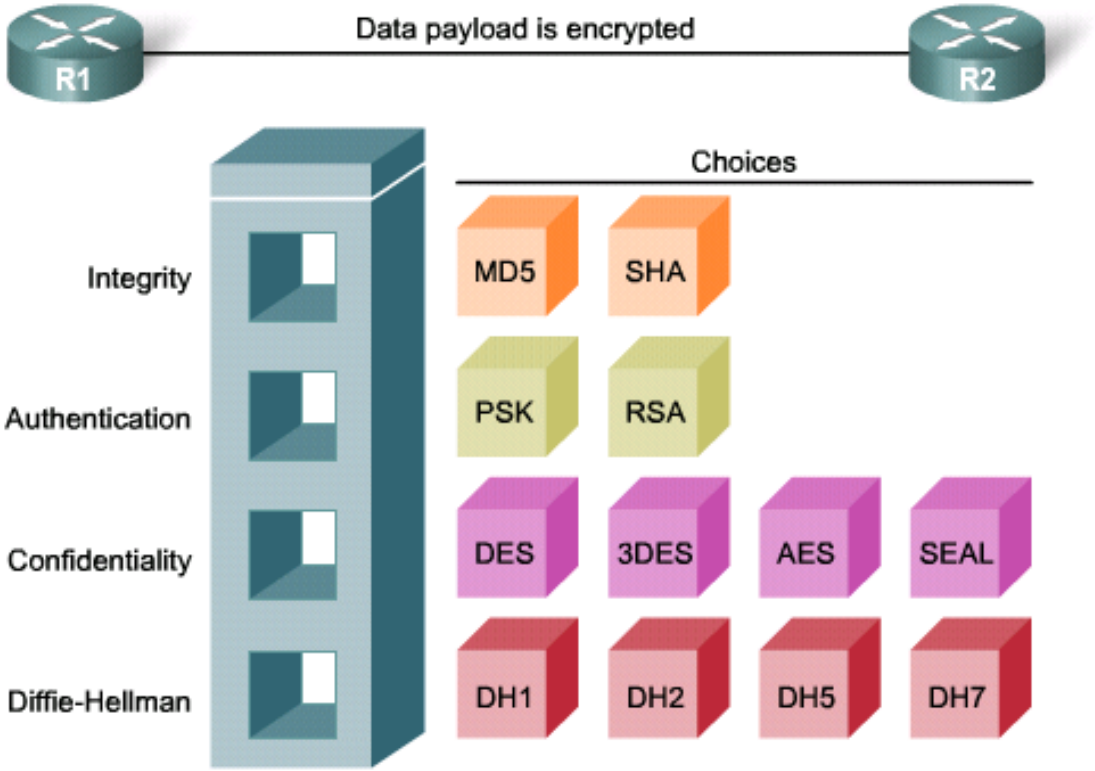


Secure Key Exchange



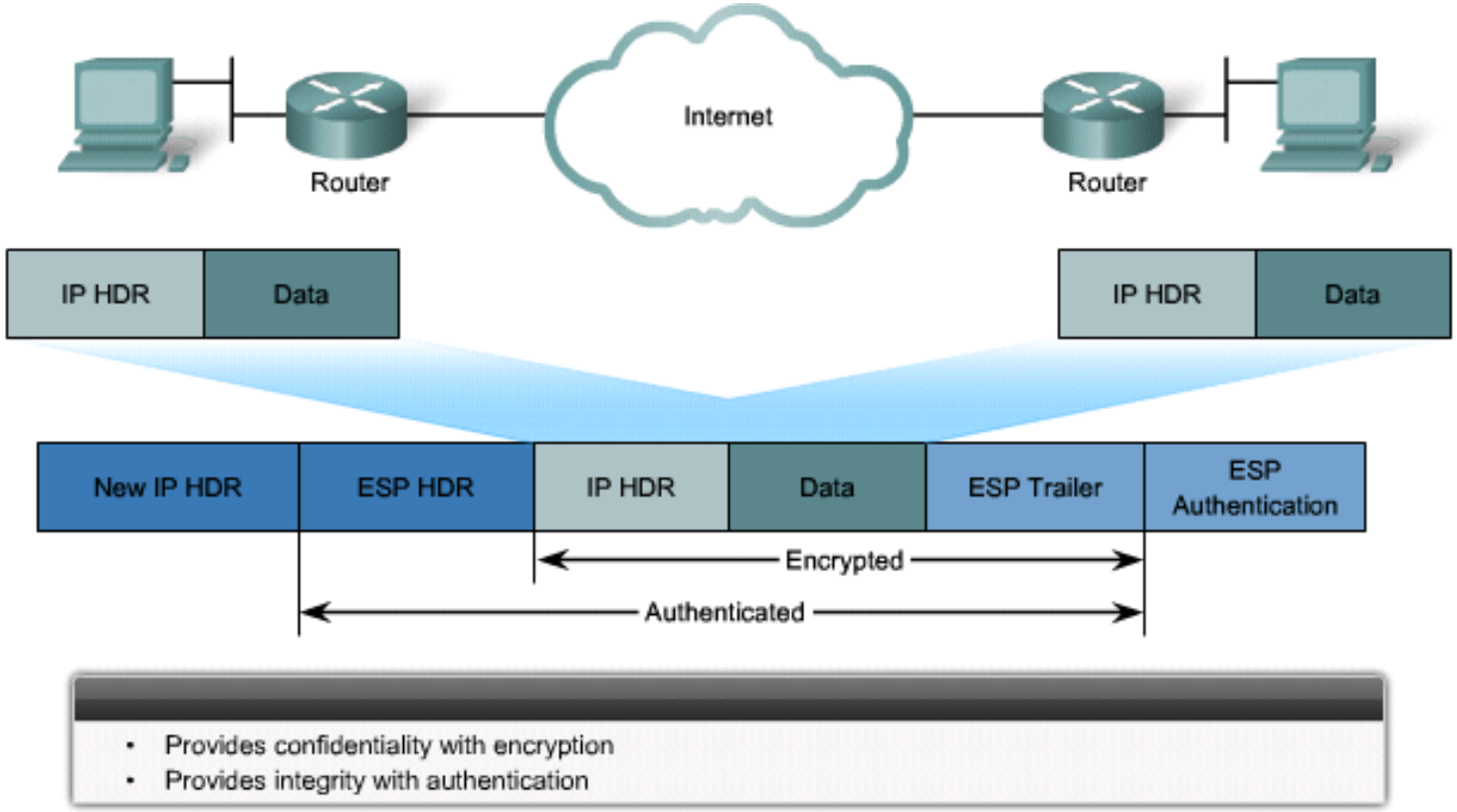
Encapsulating Security Payload (ESP)

- ESP provides the same security services as AH (authentication and integrity) AND encryption service.
  - It encapsulates the data to be protected.
  - It operates on protocol number 50.



Encapsulating Security Payload (ESP)

- ESP can also provide integrity and authentication.
  - First, the payload is encrypted using DES (default), 3DES, AES, or SEAL.
  - Next, the encrypted payload is hashed to provide authentication and data integrity using HMAC-MD5 or HMAC-SHA-1.



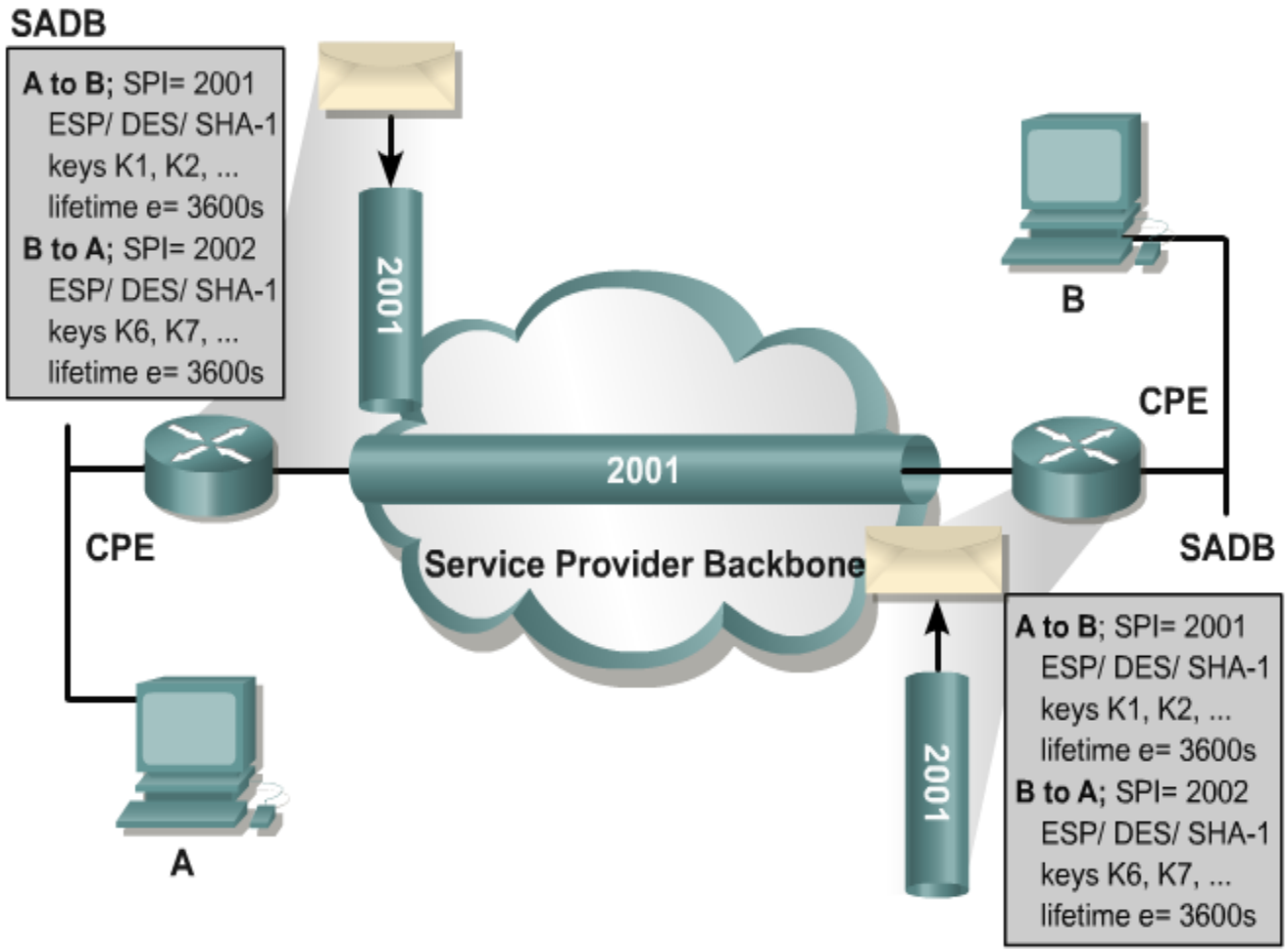
Key Exchange

- The IPsec VPN solution:
  - Negotiates key exchange parameters (IKE).
  - Establishes a shared key (DH).
  - Authenticates the peer.
  - Negotiates the encryption parameters.
- The negotiated parameters between two devices are known as a security association (SA).

Security Associations (SAs)

- SAs represent a policy contract between two peers or hosts, and describe how the peers will use IPsec security services to protect network traffic.
- SAs contain all the security parameters needed to securely transport packets between the peers or hosts, and practically define the security policy used in IPsec.

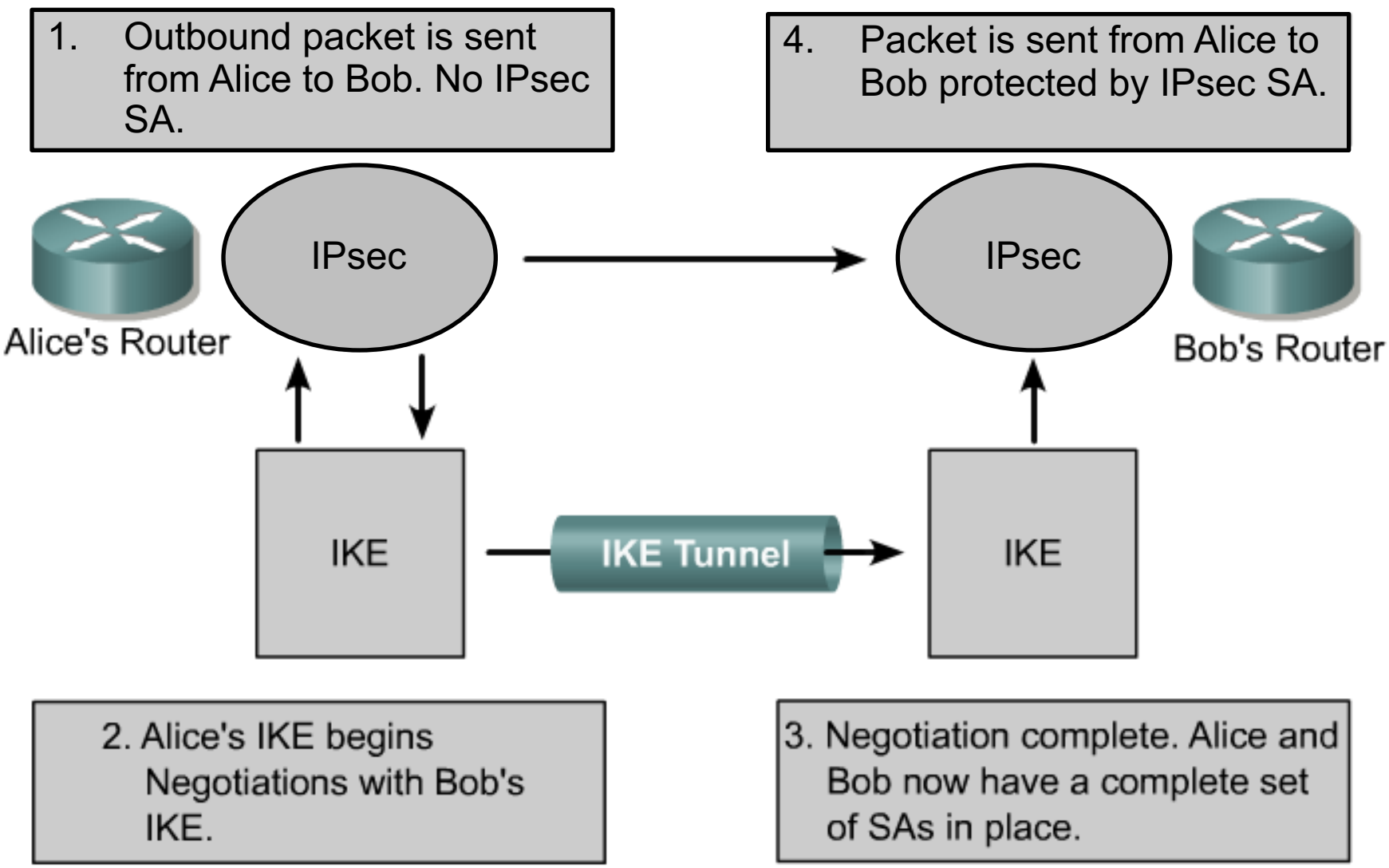
SA Security Parameters



IKE - Internet Key Exchange

- IKE helps IPsec securely exchange cryptographic keys between distant devices.
- Key Management can be preconfigured with IKE (ISAKMP) or with a manual key configuration.
  - IKE and ISAKMP are often used interchangeably.
- The IKE tunnel protects the SA negotiations.
  - After the SAs are in place, IPsec protects the data that Alice and Bob exchange.

How IPsec uses IKE



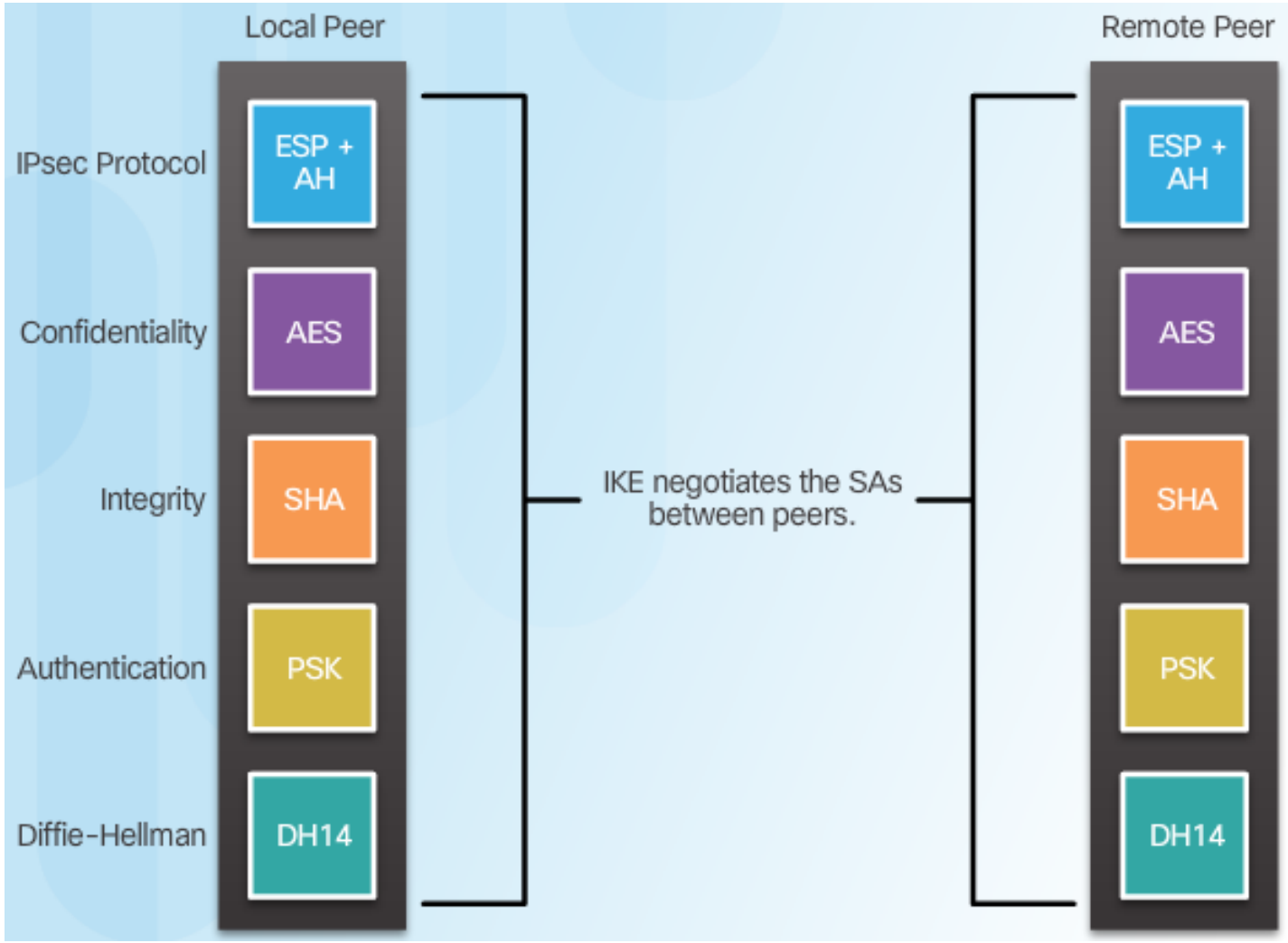
IKE - Internet Key Exchange

- There are two steps in Phase 1 (IKE) negotiation
  - Step 1 (Key Exchange)
  - Step 2 (Authentication)
- IKE negotiation can also occur in:
  - Main Mode
  - Aggressive mode
- The difference between the two is that Main mode requires the exchange of 6 messages while Aggressive mode requires only 3 exchanges.

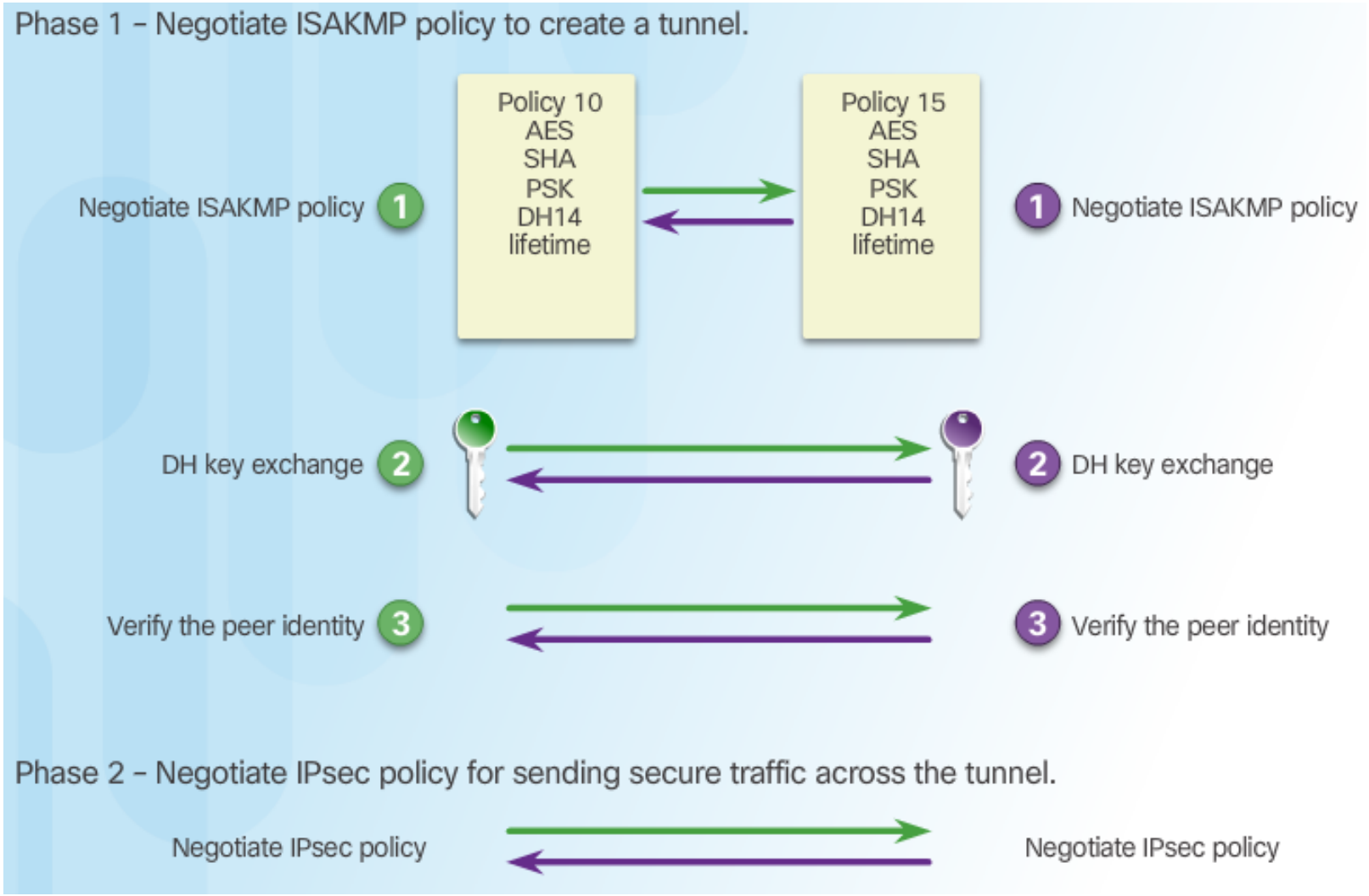
IKE Main Mode Phases

- IKE Phase One Tunnel:
  - Negotiates an IKE protection suite.
  - Exchanges keying material to protect the IKE session (DH).
  - Authenticates each other.
  - Establishes the IKE SA.
  - Main Mode requires the exchange of 6 messages while Aggressive mode only uses 3 messages.
- IKE Phase Two Tunnel:
  - Negotiates IPsec security parameters, known as IPsec transform sets.
  - Establishes IPsec SAs.
  - Periodically renegotiates IPsec SAs to ensure security.
  - Optionally performs an additional DH exchange.

The IKE Protocol

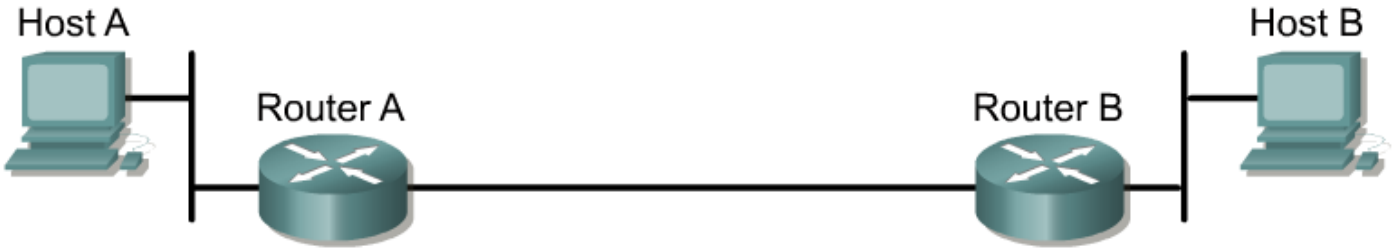


Phase 1 and 2 Key Negotiation



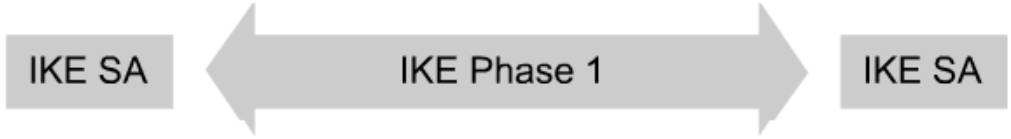
Phase 2: Negotiating SAs

Five Steps of IPsec

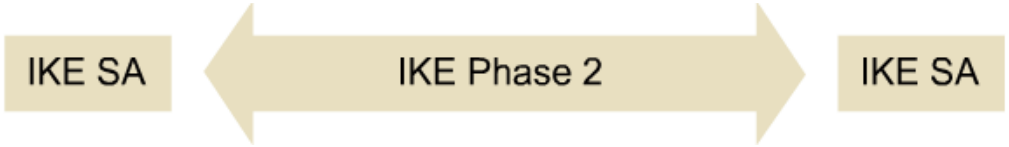


Step 1 Host A sends interesting traffic destined for Host B.

Step 2 **IKE Phase 1** authenticates IPsec peers and negotiates IKE SAs to create a secure communications channel for negotiating IPsec SAs in Phase 2.



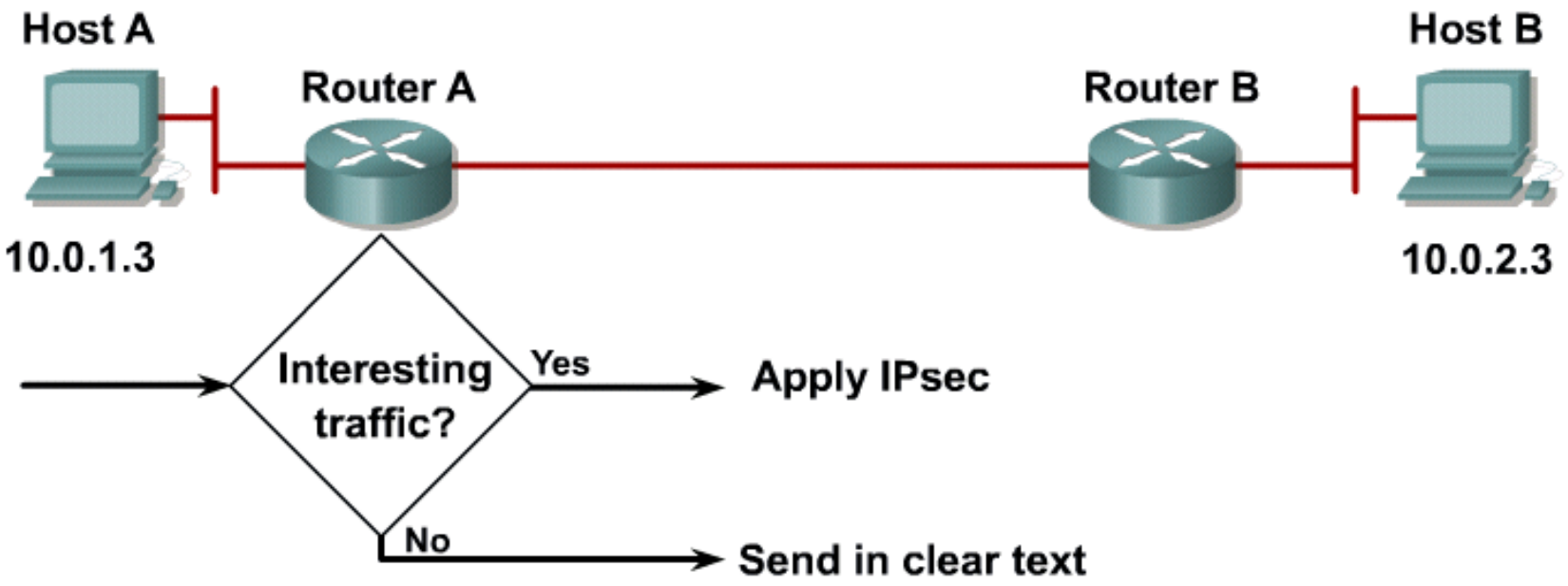
Step 3 **IKE Phase 2** negotiates IPsec SA parameters and creates matching IPsec SAs in the peers to protect data and messages exchanged between endpoints.



Step 4 Data transfer occurs between IPsec peers based on the IPsec parameters and keys stored in the SA database.

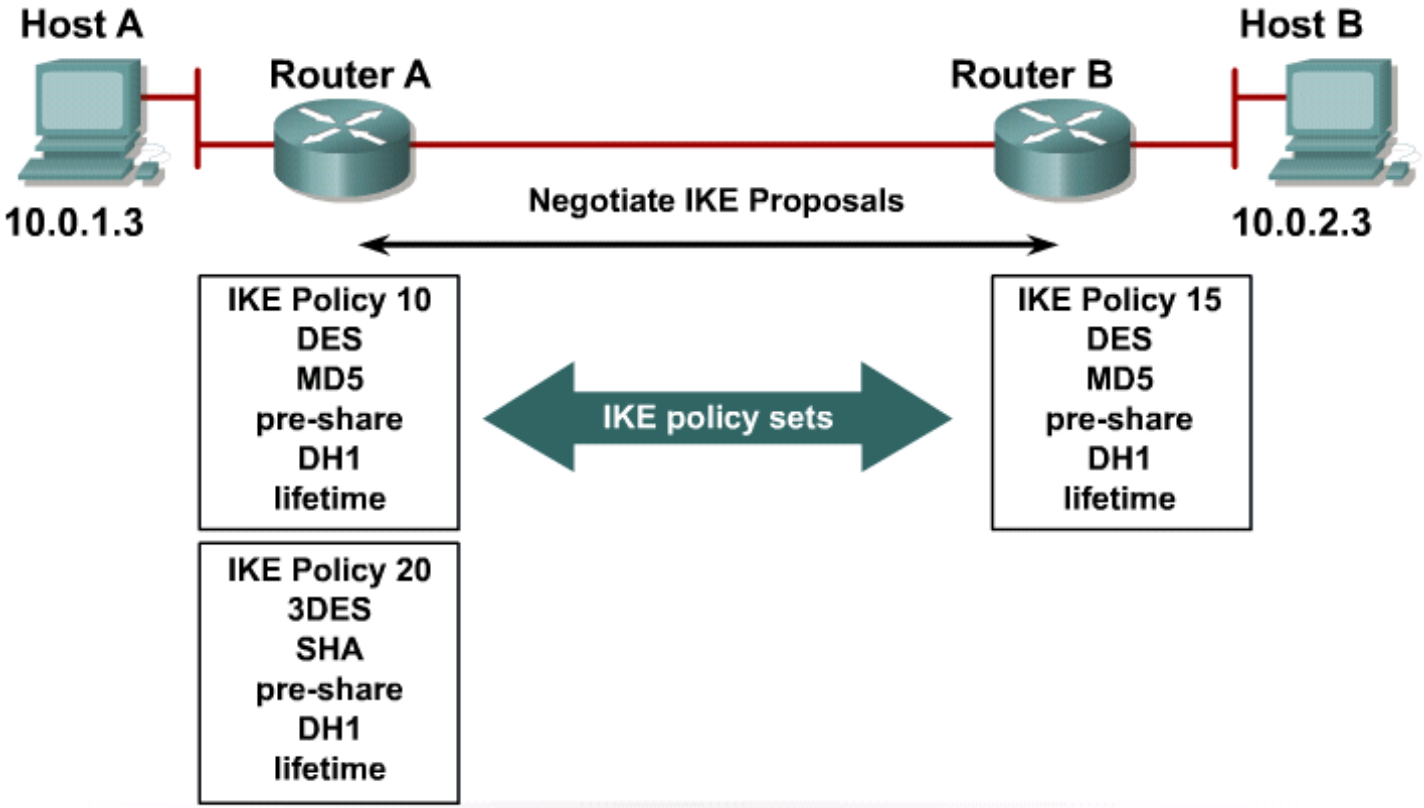
Step 5 IPsec tunnel termination occurs by SAs through deletion or by timing out.

Step 1 – Interesting Traffic



Step 2 – IKE Phase 1

IKE Policy Negotiation



Negotiates matching IKE transform sets to protect IKE exchange

Step 2 – IKE Phase 1

DH Key Exchange

RouterA randomly chooses a string and sends it to RouterB.

RouterB hashes the received string together with the pre-shared secret and yields a hash value.

RouterB sends the result of hashing back to RouterA.

RouterA calculates its own hash of the random string, together with the pre-shared secret, and matches it with the received result from the other peer.

If they match, RouterB knows the pre-shared secret, and is considered authenticated.

Step 2 – IKE Phase 1

DH Key Exchange

Now RouterB randomly chooses a different random string and sends it to RouterA.

RouterA also hashes the received string together with the pre-shared secret and yields a hash value.

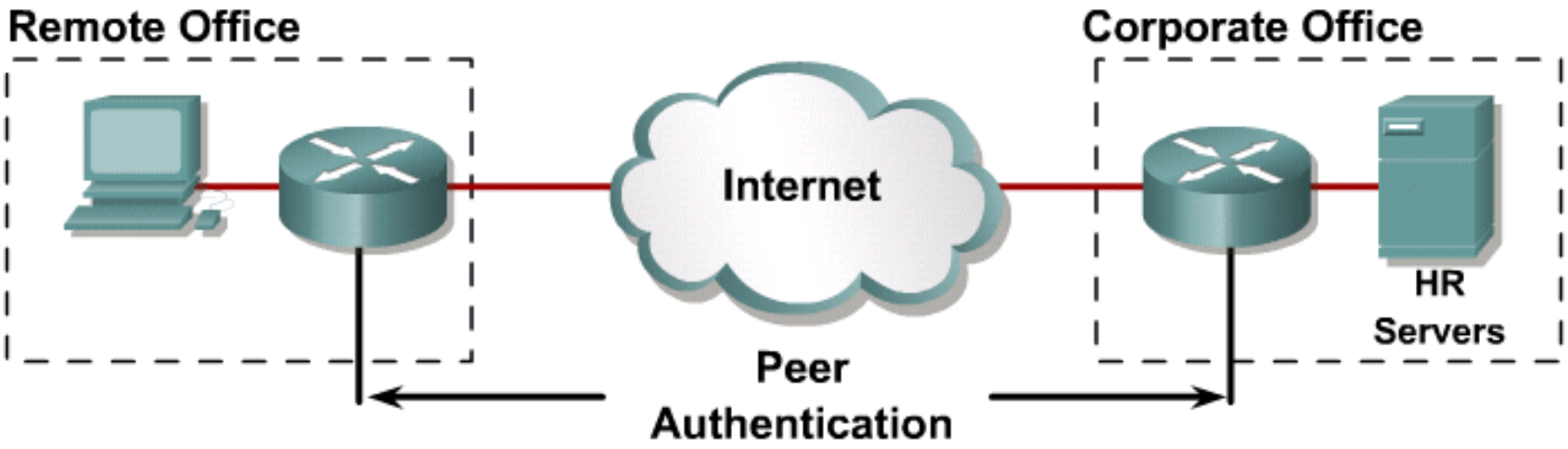
RouterA sends the result of hashing back to RouterB.

RouterB calculates its own hash of the random string, together with the pre-shared secret, and matches it with the received result from the other peer.

If they match, RouterA knows the pre-shared secret, and is considered authenticated.

Step 2 – IKE Phase 1

Peer Authentication



Peer authentication methods:
- Pre-shared keys
- RSA signatures
- RSA encrypted nonces

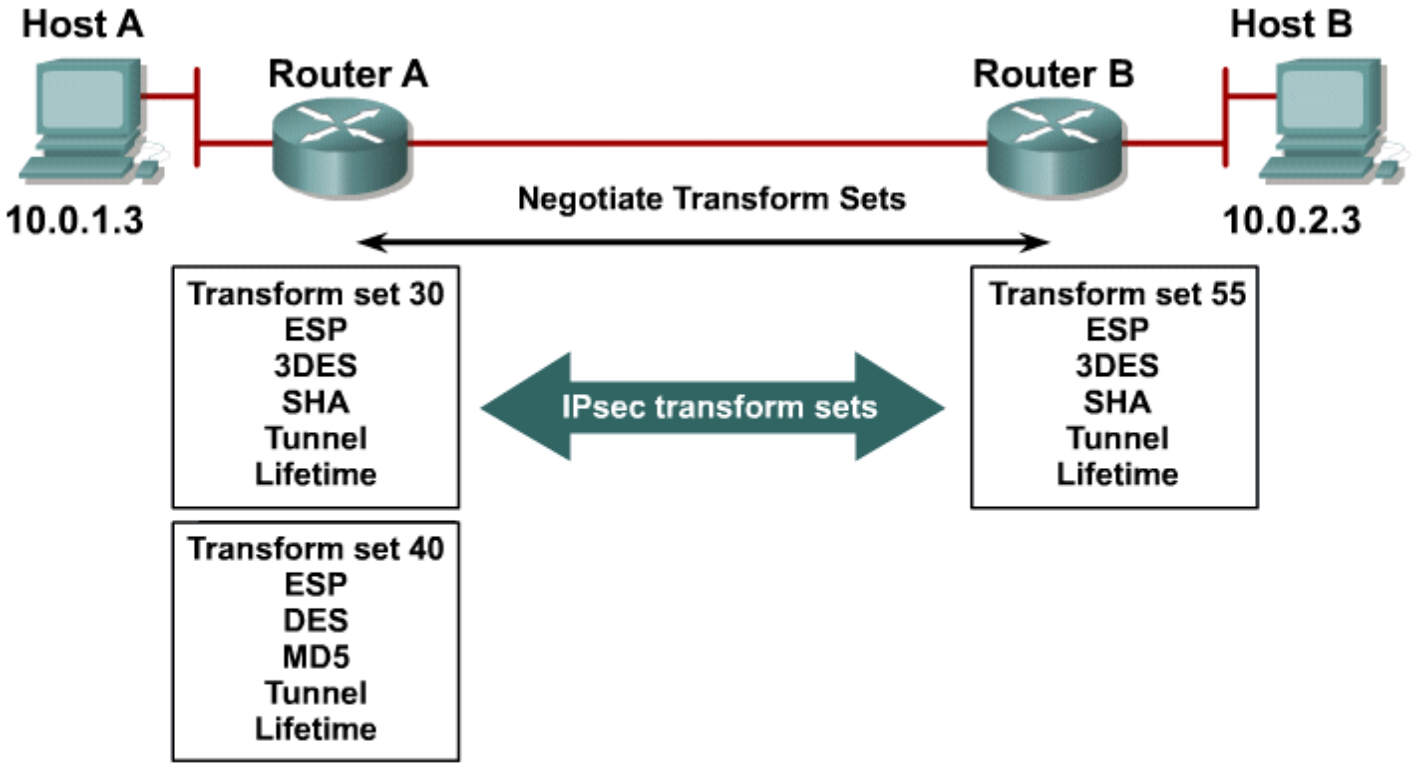
Step 3 – IKE Phase 2

IPsec Negotiation

- Negotiates IPsec security parameters and IPsec transform sets
- Establishes IPsec SAs
- Periodically renegotiates IPsec SAs to ensure security
- Optionally, performs an additional Diffie-Hellman exchange

Step 3 – IKE Phase 2

Transform Set Negotiation

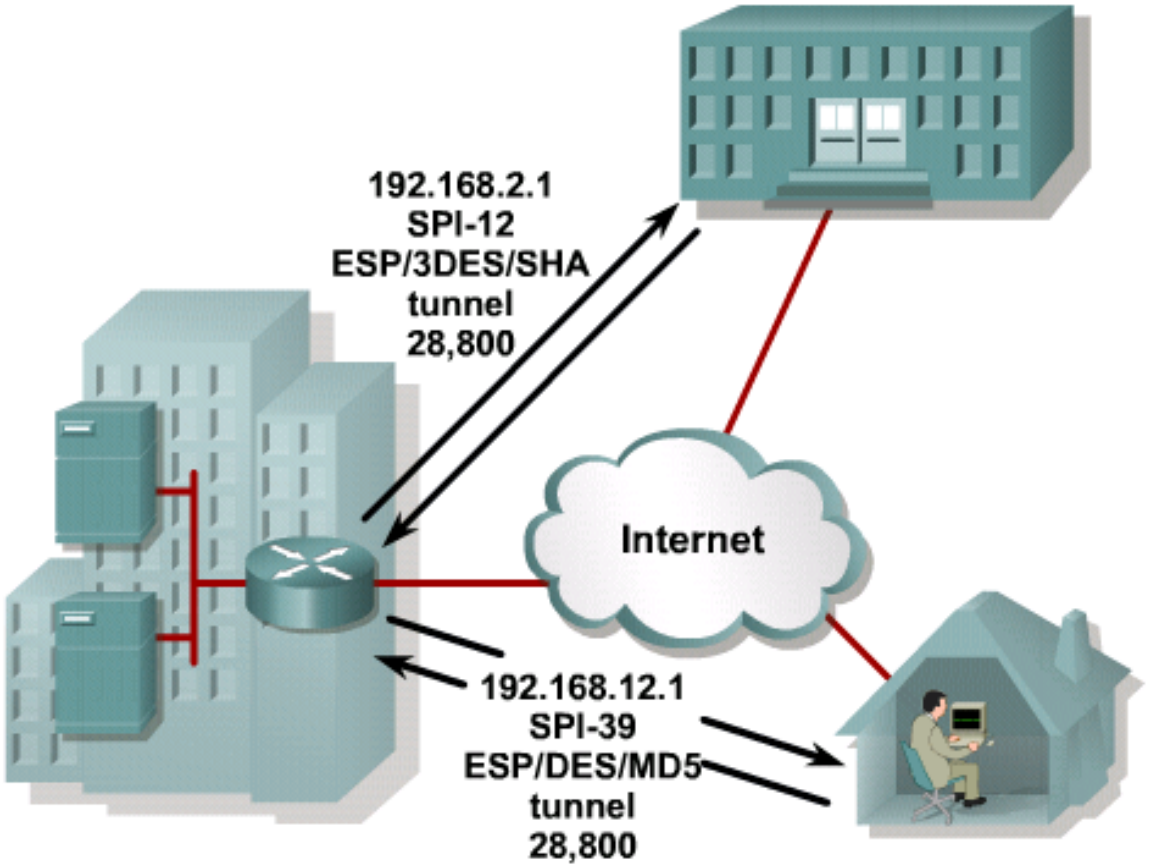


A transform set is a combination of algorithms and protocols that enact a security policy for traffic.

Step 3 – IKE Phase 2

Security Associations

- SA database:
  - Destination IP address
  - SPI
  - Protocol (ESP or AH)
- Security policy database:
  - Encryption algorithm
  - Authentication algorithm
  - Mode
  - Key lifetime



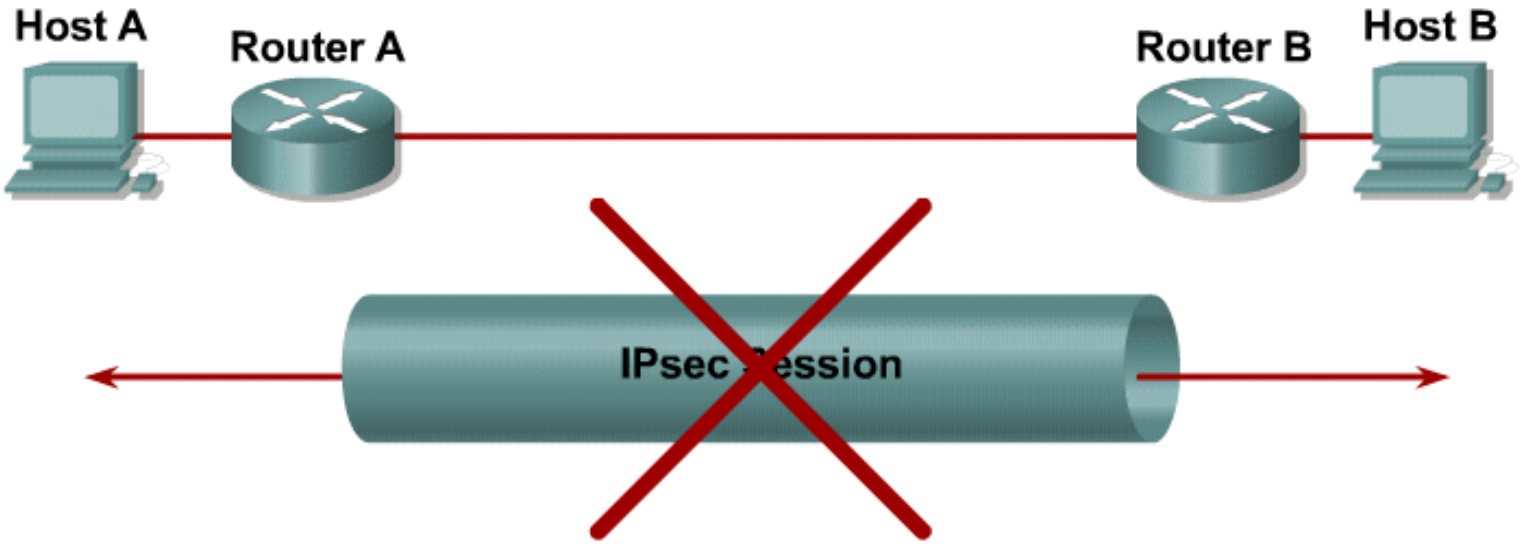
Step 4

IPsec Session

- SAs are exchanged between peers.
- The negotiated security services are applied to the traffic.

Step 5

Tunnel Termination



- A tunnel is terminated by one of the following:
  - By an SA lifetime timeout
  - The packet counter is exceeded
- IPsec SA is removed.

IPsec Tasks

1. Create an IKE policy to determine the parameters that will be used to establish the tunnel.
2. Configure the IPsec transform set which defines the parameters that the IPsec tunnel uses.
   - The set can include the encryption and integrity algorithms.
3. Create a crypto ACL.
   - The crypto ACL defines which traffic is sent through the IPsec tunnel and protected by the IPsec process.
4. Create and apply a crypto map.
   - The crypto map groups the previously configured parameters together and defines the IPsec peer devices.
   - The crypto map is applied to the outgoing interface of the VPN device.

Task 1: Configure IKE

- Creating a plan in advance is mandatory to configure IPsec encryption correctly to minimize misconfiguration.
- Determine the following policy details:
  - Key distribution method
  - Authentication method
  - IPsec peer IP addresses and hostnames
  - IKE phase 1 policies for all peers
  - Encryption algorithm, Hash algorithm, IKE SA lifetime
- Goal: Minimize misconfiguration.

IKE Phase 1 Policy Parameters

Parameter	Strong	Stronger
Encryption Algorithm	DES	3-DES or AES
Hash Algorithm	MD5	SHA-1
Authentication Method	Pre-share	RSA Encryption RSA Signature
Key Exchange	D-H Group 1	D-H Group 2 or D-H 5
IKE SA Lifetime	86400 seconds	less than 86400 seconds

Parameters		R2 Site	R3 Office
Key distribution method	Manual or **ISAKMP**	ISAKMP	ISAKMP
Encryption algorithm	**DES** or 3DES	DES	DES
Hash algorithm	MD5 or **SHA-1**	SHA-1	SHA-1
Authentication method	Pre-share or **RSA**	Pre-Share	Pre-Share
Key exchange	D-H **Group 1** or 2	Group 1	Group 1
IKE SA Lifetime	**86400** seconds or less	86400	86400

Enable IKE


```
router(config)#
[no] crypto isakmp enable
```

```
RouterA(config)#crypto isakmp enable
```

- This command globally enables or disables IKE at the router
- IKE is enabled by default
- IKE is enabled globally for all interfaces at the router
- Use the no form of the command to disable IKE
- An ACL can be used to block IKE on a particular interface

Create an IKE Policy


```
router(config)#
crypto isakmp policy priority
```

- Defines an IKE policy, which is a set of parameters used during IKE negotiation
- Invokes the config-isakmp command mode

```
RouterA(config)#crypto isakmp policy 110
```

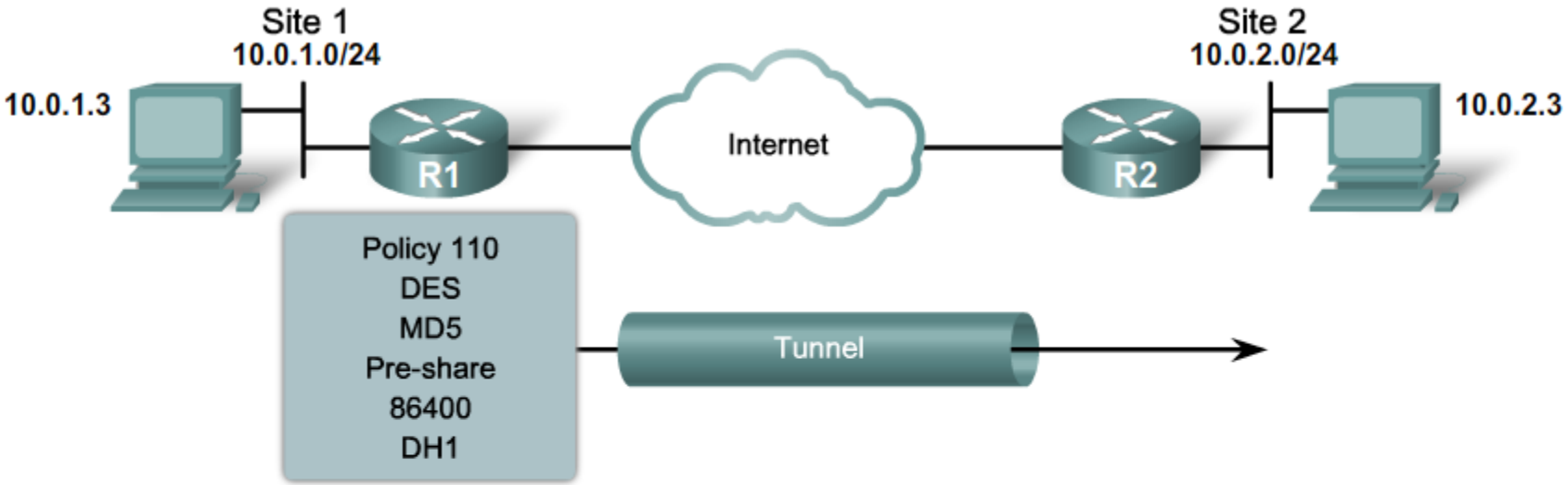
Default ISAKMP Settings

ISAKMP Parameters				
Parameter	Keyword	Accepted Values	Default Value	Description
encryption	des 3des aes aes 192 aes 256	56-bit Data Encryption Standard Triple DES 128-bit AES 192-bit AES 256-bit AES	des	Message encryption algorithm
hash	sha md5	SHA-1 (HMAC variant) MD5 (HMAC variant)	sha	Message integrity (Hash) algorithm
authentication	pre-share rsa-encr rsa-sig	pre-shared keys RSA encrypted nonces RSA signatures	rsa-sig	Peer authentication method
group	1 2 5	768-bit Diffie-Hellman (DH) 1024-bit DH 1536-bit DH	1	Key exchange parameters (DH group identifier)
lifetime	*seconds*	Can specify any number of seconds	86,400 sec (one day)	ISAKMP-established SA lifetime
Note: Actual parameters vary based on IOS image.				

Default ISAKMP Settings

```
RouterA# show crypto isakmp policy
Protection suite of priority 110
      encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
      hash algorithm:         Message Digest 5
      authentication method:  Pre-Shared Key
      Diffie-Hellman group:   #1 (768 bit)
      lifetime:               86400 seconds, no volume limit
Default protection suite
      encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
      hash algorithm:         Secure Hash Standard
      authentication method:  Rivest-Shamir-Adleman Signature
      Diffie-Hellman group:   #1 (768 bit)
      lifetime:               86400 seconds, no volume limit
```

Create an IKE Policy



router(config)#

crypto isakmp policy *priority*

Defines the parameters within the IKE policy

```
R1(config)# crypto isakmp policy 110
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# encryption des
R1(config-isakmp)# group 1
R1(config-isakmp)# hash md5
R1(config-isakmp)# lifetime 86400
```

ISAKMP Policy Negotiation

R1#

```
crypto isakmp policy 100
  hash md5
  authentication pre-share
!
crypto isakmp policy 200
  hash sha
  authentication rsa-sig
!
crypto isakmp policy 300
  hash md5
  authentication pre-share
```

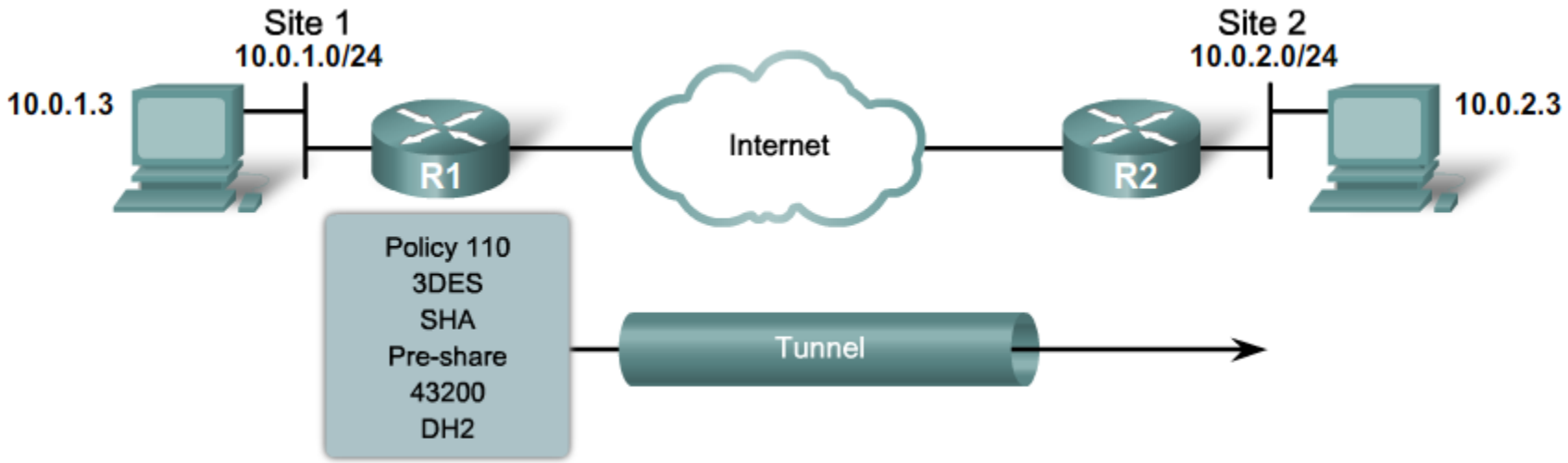
R2#

```
crypto isakmp policy 100
  hash md5
  authentication pre-share
!
crypto isakmp policy 200
  hash sha
  authentication rsa-sig
!
crypto isakmp policy 300
  hash md5
  authentication rsa-sig
```

Policy 100 and 200 can be successfully negotiated, but policy 300 cannot.

Note: smaller priority numbers have higher priority.

ISAKMP Policy Negotiation




```
R1(config)# crypto isakmp policy 110
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# encryption 3des
R1(config-isakmp)# group 2
R1(config-isakmp)# hash sha
R1(config-isakmp)# lifetime 43200
```

```
R2(config)# crypto isakmp policy 100
R2(config-isakmp)# authentication pre-share
R2(config-isakmp)# encryption 3des
R2(config-isakmp)# group 2
R2(config-isakmp)# hash sha
R2(config-isakmp)# lifetime 43200
```

- R1 attempts to establish a VPN tunnel with R2 because it has interesting traffic destined for R2 and therefore sends its IKE policy parameters.
- R2 must have an ISAKMP policy configured with the same parameters. Notice however, that policy numbers are only locally significant and do not have to match between IPsec peers.

Configure Pre-Shared Keys

```
router(config)#
crypto isakmp key keystring address peer-address
```

```
router(config)#
crypto isakmp key keystring hostname hostname
```

Parameter	Description
keystring	This parameter specifies the PSK. Use any combination of alphanumeric characters up to 128 bytes. This PSK must be identical on both peers.
peer-address	This parameter specifies the IP address of the remote peer.
hostname	This parameter specifies the hostname of the remote peer. This is the peer hostname concatenated with its domain name (for example, myhost.domain.com).

- The *peer-address* or *hostname* can be used, but must be used consistently between peers.
- If the *hostname* is used, then the **crypto isakmp identity hostname** command must also be configured.

- By default, the ISAKMP identity is set to use the IP address.

Configure Pre-Shared Keys


```
R1(config)# crypto isakmp policy 110
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# encryption 3des
R1(config-isakmp)# group 2
R1(config-isakmp)# hash sha
R1(config-isakmp)# lifetime 43200
R1(config-isakmp)# exit
R1(config)# crypto isakmp key cisco123 address 172.30.2.2
R1(config)#
```

Note:

- The keystring cisco123 matches.
- The address identity method is specified.
- The ISAKMP policies are compatible.
- Default values do not have to be configured.

```
R2(config)# crypto isakmp policy 110
R2(config-isakmp)# authentication pre-share
R2(config-isakmp)# encryption 3des
R2(config-isakmp)# group 2
R2(config-isakmp)# hash sha
R2(config-isakmp)# lifetime 43200
R2(config-isakmp)# exit
R2(config)# crypto isakmp key cisco123 address 172.30.1.2
R2(config)#
```

Verify IKE Configuration


```
RouterA# show crypto isakmp policy
Protection suite of priority 110
      encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
      hash algorithm:         Message Digest 5
      authentication method:  Pre-Shared Key
      Diffie-Hellman group:   #1 (768 bit)
      lifetime:               86400 seconds, no volume limit
Default protection suite
      encryption algorithm:   DES - Data Encryption Standard (56 bit keys).
      hash algorithm:         Secure Hash Standard
      authentication method:  Rivest-Shamir-Adleman Signature
      Diffie-Hellman group:   #1 (768 bit)
      lifetime:               86400 seconds, no volume limit
```

Task 2: Configure the Transform Sets

- Determine the following policy details:
  - IPsec algorithms and parameters for optimal security and performance
  - Transforms sets
  - IPsec peer details
  - IP address and applications of hosts to be protected
  - Manual or IKE-initiated SAs
- Goal: Minimize misconfiguration.

IPsec Transforms Supported in IOS

- Cisco IOS software supports the following IPsec transforms:

```
CentralA(config)# crypto ipsec transform-set transform-set-name ?
ah-md5-hmac    AH-HMAC-MD5 transform
ah-sha-hmac    AH-HMAC-SHA transform
esp-3des       ESP transform using 3DES(EDE) cipher (168 bits)
esp-des        ESP transform using DES cipher (56 bits)
esp-md5-hmac   ESP transform using HMAC-MD5 auth
esp-sha-hmac   ESP transform using HMAC-SHA auth
esp-null       ESP transform w/o cipher
```

Note:

esp-md5-hmac and **esp-sha-hmac** provide more data integrity.

They are compatible with NAT/PAT and are used more frequently than **ah-md5-hmac** and **ah-sha-hmac**.

IPsec Policy Example

Policy	Host A	Host B
Transform set	ESP-DES, Tunnel	ESP-DES, Tunnel
Peer hostname	RouterB	RouterA
Peer IP address	172.30.2.2	172.30.1.2
Hosts to be encrypted	10.0.1.3	10.0.2.3
Traffic (packet) type to be encrypted	TCP	TCP
SA establishment	ipsec-isakmp	ipsec-isakmp

Specific IPsec **show** Commands

```
RouterA# show crypto isakmp policy
Default protection suite
encryption algorithm: DES - Data Encryption Standard (56 bit keys)
hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman Group: #1 (768 bit)
lifetime: 86400 seconds, no volume limit
```

```
RouterA# show crypto map
Crypto Map “MYMAP" 10 ipsec-isakmp
Peer = 172.30.2.2
Extended IP access list 102
access-list 102 permit ip host 172.30.1.2 host 172.30.2.2
Current peer: 172.30.2.2
Security association lifetime: 4608000 kilobytes/3600 seconds
PFS (Y/N): N
Transform sets={ MY-SET, }
```

```
RouterA# show crypto ipsec transform-set MY-SET
Transform set MY-SET: { esp-des }
will negotiate = { Tunnel, },
```

Configure Transform Sets

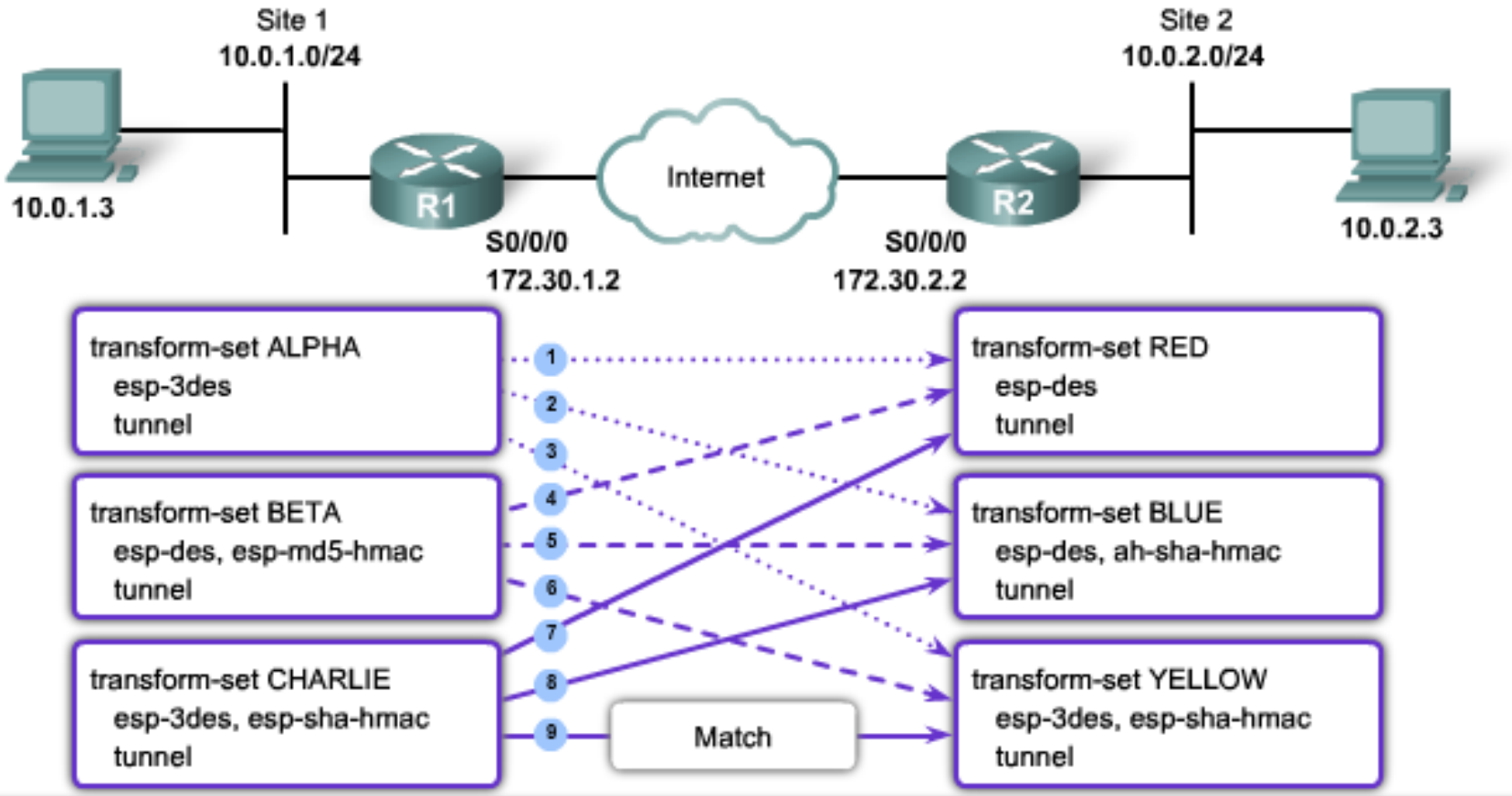
```
router(config)#
crypto ipsec transform-set transform-set-name transform1 [transform2]
[transform3][transform4]
```

crypto ipsec transform-set Parameters

Command	Description
transform-set-name	This parameter specifies the name of the transform set to create (or modify).
transform1, transform2, transform3, transform4	Type of transform set. Specify up to four "transforms": one Authentication Header (AH), one Encapsulating Security Payload (ESP) encryption, one ESP authentication. These transforms define the IP Security (IPsec) security protocols and algorithms.

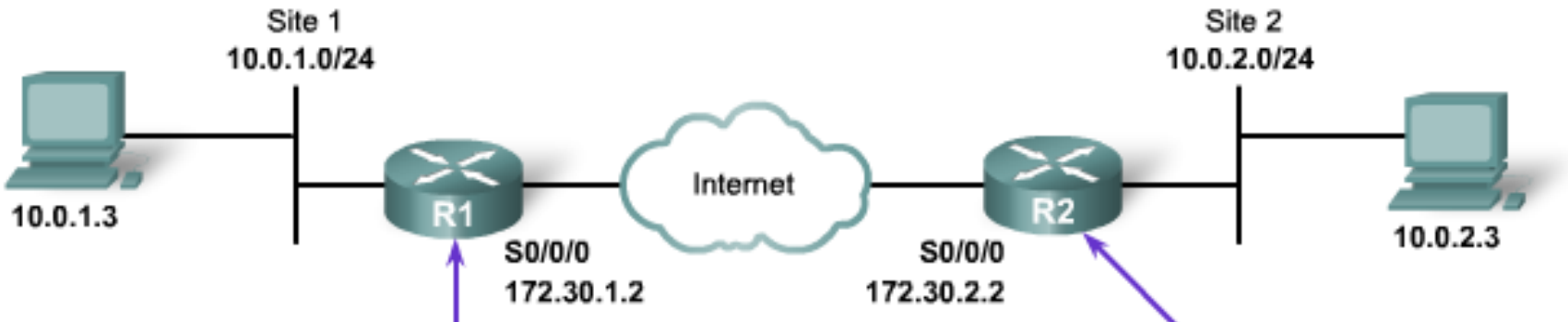
- A transform set is a combination of IPsec transforms that enact a security policy for traffic.
- A transform set can have one AH transform and up to two ESP transforms.

Transform Set Negotiation



- Transform sets are negotiated during IKE Phase 2.
- The 9th attempt found matching transform sets (CHARLIE - YELLOW).

Transform Set Negotiation




```
R1(config)# crypto isakmp key cisco123 address 172.30.2.2
R1(config)# crypto ipsec transform-set MYSET esp-aes 128
R1(cfg-crypto-trans)# exit
R1(config)#
```

```
R2(config)# crypto isakmp key cisco123 address 172.30.1.2
R2(config)# crypto ipsec transform-set OTHERSET esp-aes 128
R2(cfg-crypto-trans)# exit
```

Note:

- Peers must share the same transform set settings.
- Names are only locally significant.

Configure Security Association Lifetimes

- Configures global IPsec lifetime values used when negotiating IPsec security associations.
- IPsec SA lifetimes are negotiated during IKE phase 2.


```
router(config)#
crypto ipsec security-association lifetime
     {seconds seconds | kilobytes kilobytes}
```

```
RouterA(config)#crypto ipsec security-association
lifetime 86400
```

Task 3: Configure Crypto ACLs


```
router(config)#
```

```
access-list access-list-number {deny | permit} protocol source source-
wildcard destination destination-wildcard
```

Command	Description
permit	This option causes all IP traffic that matches the specified conditions to be protected by cryptography, using the policy described by the corresponding crypto map entry.
deny	This option instructs the router to route traffic in plaintext.
protocol	This option specifies which traffic to protect by cryptography based on the protocol, such as TCP, UDP, or ICMP. If the protocol is IP, then all IP traffic matching that permit statement is encrypted.
source **and** *destination*	If the ACL statement is a permit statement, these are the networks, subnets, or hosts between which traffic should be protected. If the ACL statement is a deny statement, then the traffic between the specified source and destination is sent in plaintext.

Configure Symmetrical Peer Crypto ACL

RouterA#(config)

```
access-list 110 permit tcp 10.0.1.0 0.0.0.255 10.0.2.0 0.0.0.255
```

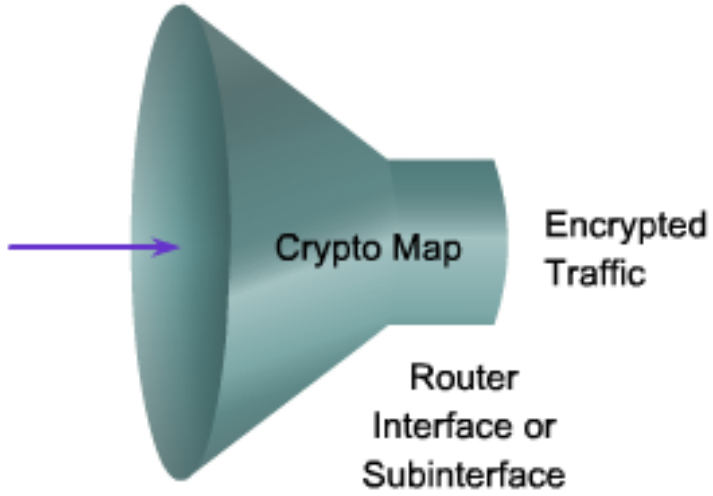
RouterB#(config)

```
access-list 110 permit tcp 10.0.2.0 0.0.0.255 10.0.1.0 0.0.0.255
```

Task 4: Apply the Crypto Map

Crypto maps define the following:

- ACL to be used
- Remote VPN peers
- Transform set to be used
- Key management method
- SA lifetimes



Configure IPsec Crypto Maps

```
router(config)#
```

```
crypto map map-name seq-num ipsec-manual
```

```
crypto map map-name seq-num ipsec-isakmp [dynamic dynamic-map-name]
```

`crypto map` Parameters

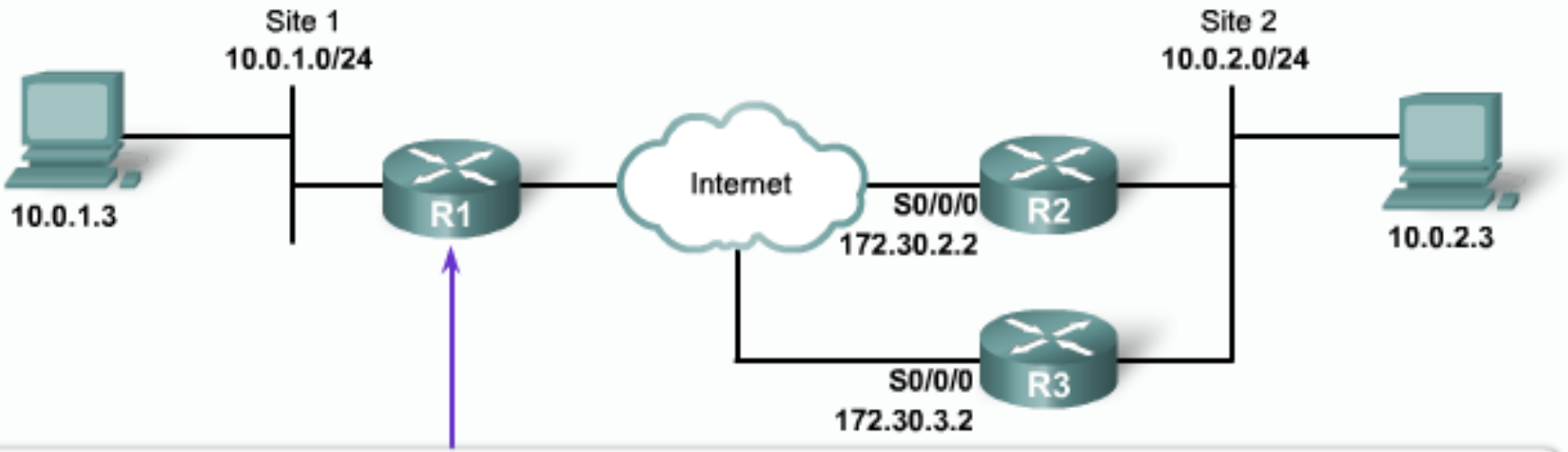
Command Parameters	Description
map-name	Defines the name assigned to the crypto map set or indicates the name of the crypto map to edit.
seq-num	The number assigned to the crypto map entry.
ipsec-manual	Indicates that ISAKMP will not be used to establish the IPsec SAs.
ipsec-isakmp	Indicates that ISAKMP will be used to establish the IPsec SAs.
cisco	(Default value) Indicates that CET will be used instead of IPsec for protecting the traffic.
dynamic	(Optional) Specifies that this crypto map entry references a preexisting static crypto map. If this keyword is used, none of the crypto map configuration commands are available.
dynamic-map-name	(Optional) Specifies the name of the dynamic crypto map set that should be used as the policy template.

Configure IPsec Crypto Maps

`crypto map` Configuration Mode Commands

Command	Description
`set`	Used with the **`peer`**, **`pfs`**, **`transform-set`**, and **`security-association`** commands.
`peer` [*hostname* | *ip-address*]	Specifies the allowed IPsec peer by IP address or hostname.
`pfs` [**`group1`** | **`group2`**]	Specifies DH Group 1 or Group 2.
`transform-set` [*set_name(s)*]	Specify list of transform sets in priority order. When the **`ipsec-manual`** parameter is used with the **`crypto map`** command, then only one transform set can be defined. When the **`ipsec-isakmp`** parameter or the **`dynamic`** parameter is used with the **`crypto map`** command, up to six transform sets can be specified.
`security-association lifetime`	Sets SA lifetime parameters in seconds or kilobytes.
`match address` [*access-list-id* | *name*]	Identifies the extended ACL by its name or number. The value should match the access-list-number or name argument of a previously defined IP-extended ACL being matched.
`no`	Used to delete commands entered with the **`set`** command.
`exit`	Exits crypto map configuration mode.

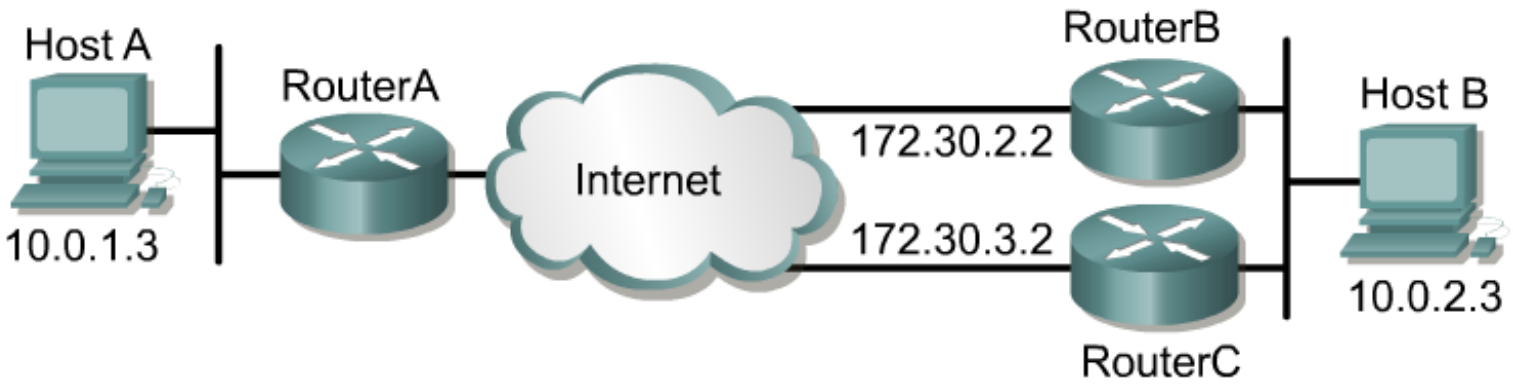
Configure IPsec Crypto Maps




```
R1(config)# crypto map MYMAP 10 ipsec-isakmp
R1(config-crypto-map)# match address 110
R1(config-crypto-map)# set peer 172.30.2.2 default
R1(config-crypto-map)# set peer 172.30.3.2
R1(config-crypto-map)# set pfs group1
R1(config-crypto-map)# set transform-set mine
R1(config-crypto-map)# set security-association lifetime seconds 86400
```

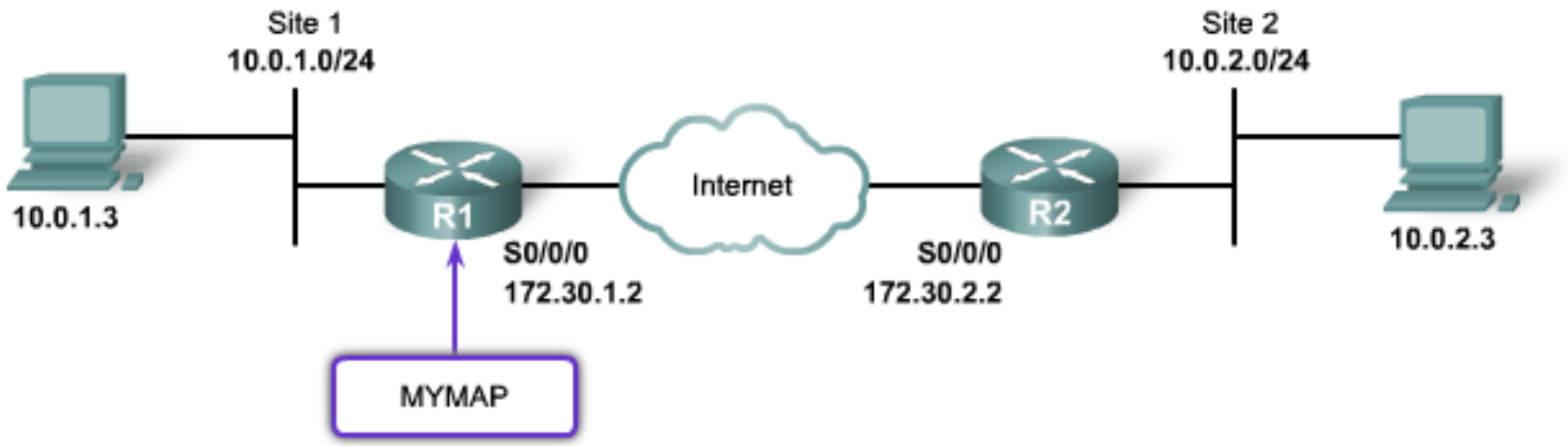
- Multiple peers can be specified for redundancy.

Example Crypto Map Commands




```
RouterA(config)# crypto map MYMAP 110 ipsec-isakmp
RouterA(config-crypto-map)# match address 110
RouterA(config-crypto-map)# set peer 172.30.2.2
RouterA(config-crypto-map)# set peer 172.30.3.2
RouterA(config-crypto-map)# set transform-set MINE
RouterA(config-crypto-map)# set security-association lifetime 86400
```

Applying Crypto Maps to Interfaces




```
router(config-if)#
crypto map map-name
```

```
R1(config)# interface serial0/0/0
R1(config-if)# crypto map MYMAP
```

- Applies the crypto map to outgoing interface
- Activates the IPsec policy

IPsec Configuration Examples


```
RouterA#show running config
crypto ipsec transform-set mine
esp-des
!
crypto map mymap 10 ipsec-isakmp
set peer 172.30.2.2
set transform-set mine
match address 110
!
interface Ethernet 0/1
ip address 172.30.1.2 255.255.255.0
no ip directed-broadcast
crypto map mymap
!
access-list 110 permit tcp 10.0.1.0
0.0.0.255 10.0.2.0 0.0.0.255
```

```
RouterB#show running config
crypto ipsec transform-set mine
esp-des
!
crypto map mymap 10 ipsec-isakmp
set peer 172.30.1.2
set transform-set mine
match address 110
!
interface Ethernet 0/1
ip address 172.30.2.2 255.255.255.0
no ip directed-broadcast
crypto map mymap
!
access-list 110 permit tcp 10.0.2.0
0.0.0.255 10.0.1.0 0.0.0.255
```

Verify IPsec

Show Command	Description
`show crypto map`	Displays configured crypto maps
`show crypto isakmp policy`	Displays configured IKE policies
`show crypto ipsec sa`	Displays established IPsec tunnels
`show crypto ipsec transform-set`	Displays configured IPsec transform sets
`debug crypto isakmp`	Debugs IKE events
`debug crypto ipsec`	Debugs IPsec events