UNIVERSITY OF ŽILINA
Faculty of Management Science and Informatics

Chapter 8: Implementing Virtual Private Networks

Concepts of Virtual Private Networks (VPS / VPN) and IPsec (IP Security)

KATEDRA INFORMAČNÝCH SIETÍ * FAKULTA RIADENIA A INFORMATIKY ŽU *

CCNA Security v2.0 / Network Security 1.0

Ch. 8 / Modules 18-19

cisco
Networking Academy

Chapter Outline

- What is a VPN
- VPN types
- GRE tunnels
- IPsec

- Introduction
- VPNs
- IPsec VPN Components and Operations
- Implementing Site-to-Site IPsec VPNs with CLI

VPNs - Virtual Private Networks

Upon completion of this section, you should be able to:

Describe VPNs and their benefits.

Compare site-to-site and remote-access VPNs.

VPN Overview

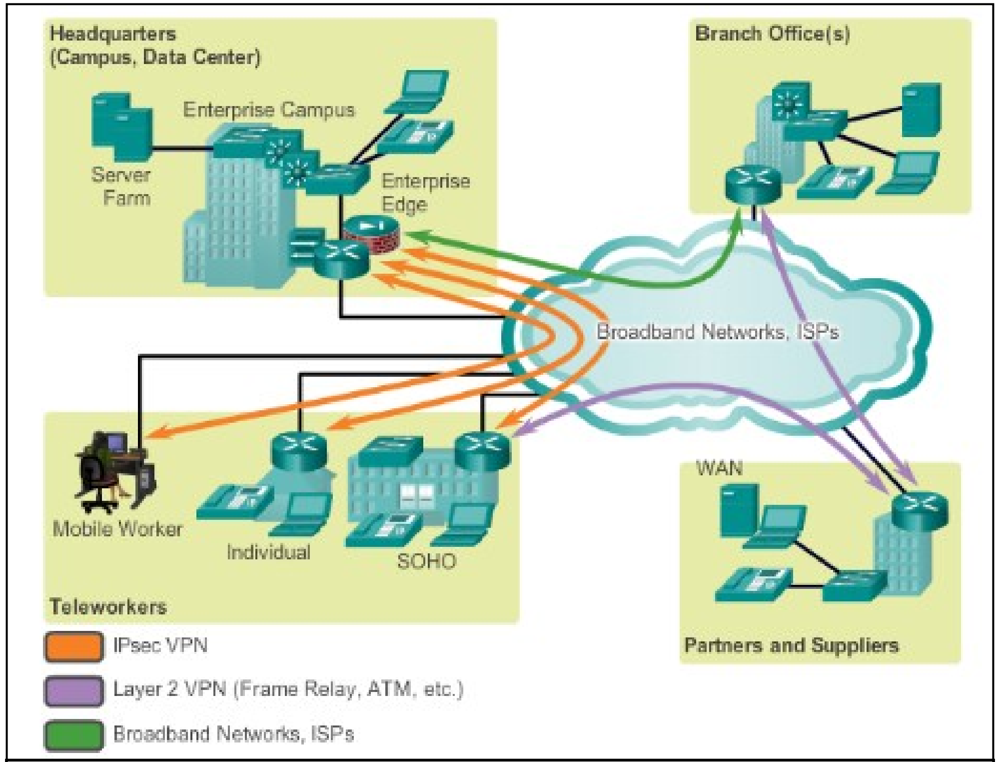
VPN – remote access solution

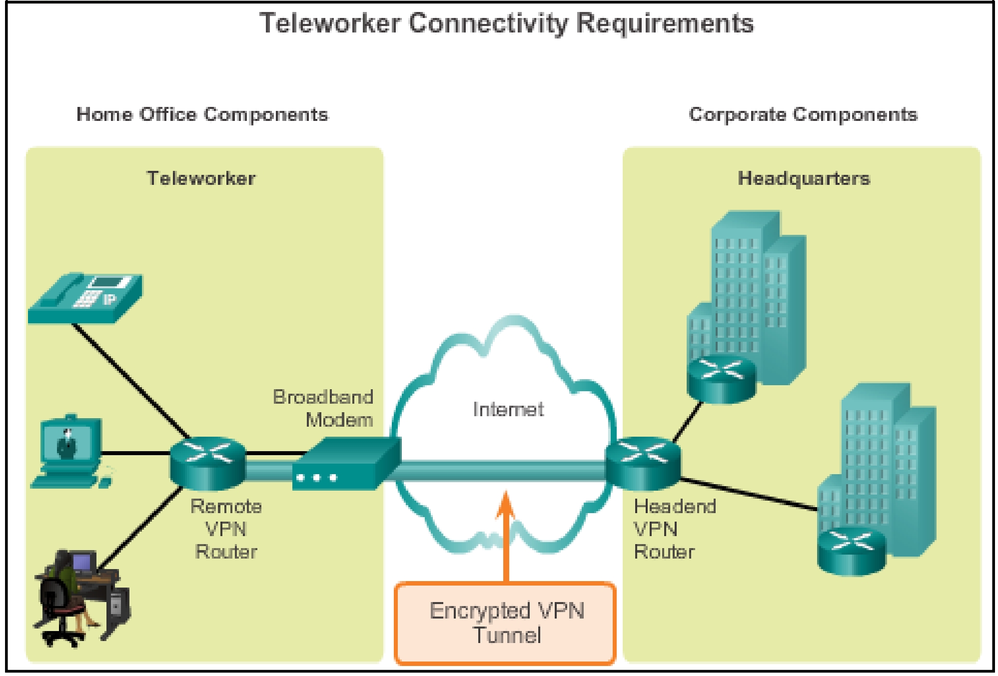
- Why do I need remote access?
  - Businesses typically need to address remote access to a company's network for reasons:
    - **Integration of branch networks with headquarters**
      - E.g. access from the branch office network (s) to headquarters services (internal services and servers)
    - **Customer access** to the company's internal services
      - E.g. various production systems for the supply of goods and services
    - **Teleworking/Homeworking**
      - Allowing employees to work from home
      - Freelancing

Remote access solutions

Solution requirements

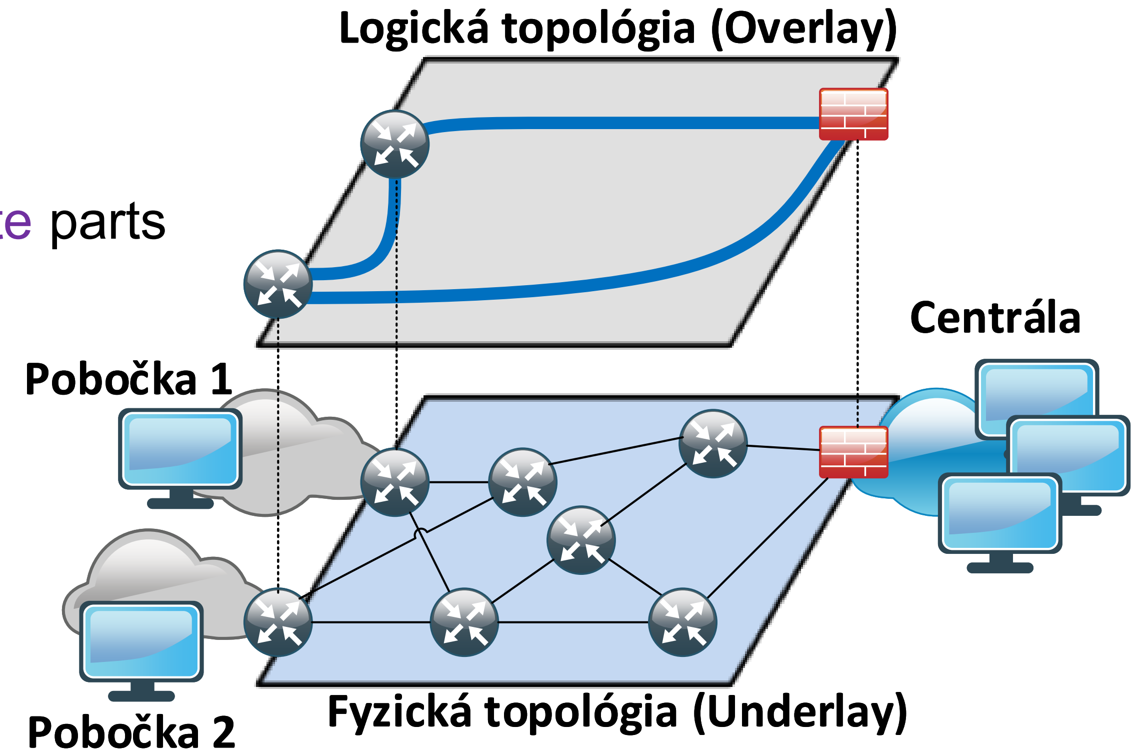
- Each of the previous options requires:
  - Broadband / fast access
    - Various services (VoIP, TelePresence, sharing, etc.)
  - Secure access
- Broadband and fast access solutions
  - Speed higher than 200kbps
    - Cable / DSL / WiFi / WiMAX / Fiber („Always-on“ technologies)
  - It is necessary to consider when choosing
    - Price, speed
    - Security
    - Simplicity and reliability
- Secure access solution
  - **Private VPN services ISP**
    - e.g. VPLS via MPLS to SK, Frame Relay and so on
  - L3 VPN over the public internet
    - This is how the CCNA understands it





VPN basics

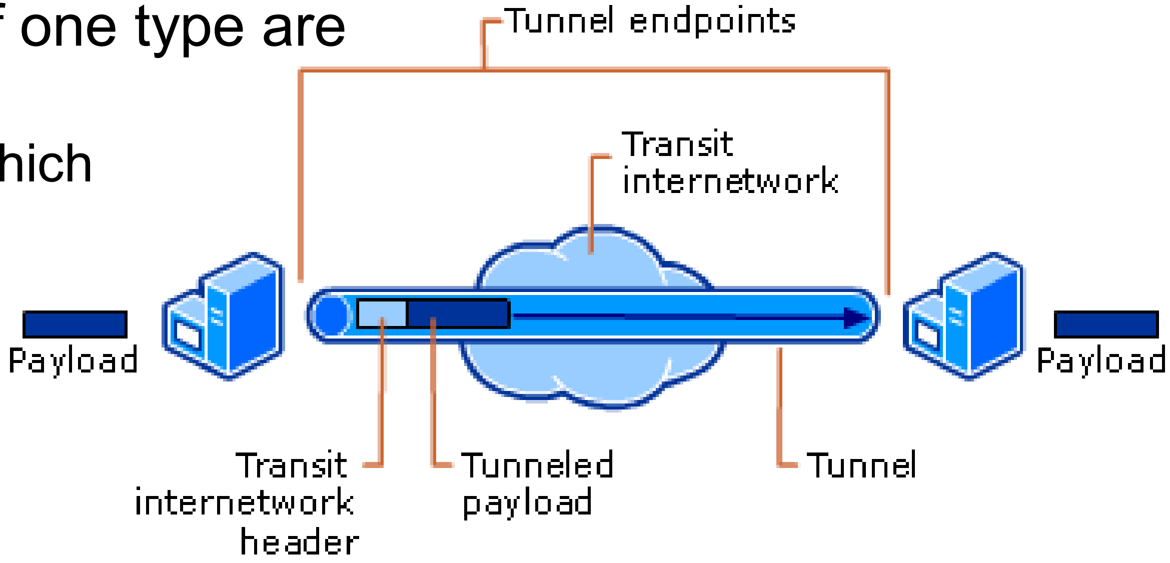
- **Virtual Private Networks**
  - Technically, private end-to-end network, through which organizations connect their private parts (eg branches)
    - Typically, realized over physical infrastructure of SPs (third-party networks)



- **Realization => virtual interconnection – a network tunnel - over existing networks of ISP providers**
  - Creating so-called **Overlay** (VPN tunnel **network**)
  - On top of so-called **Underlay** (ISP networks)
  - Note: *At present, VPN is already mainly understood as a secure (encrypted) network created via IPSec in the form of a tunnel*

What is protocol tunneling?

- Many times, it is necessary to create the illusion of a new network over an existing network
  - The existing network does not know the protocol we need to transfer through it or the service we want to use
  - We want to use the existing network only as a transport, but from the point of view of our internal network it should be almost invisible
  - We need to link multiple sites, potentially with a private address range
  - We do not trust the existing network and we want to transfer data through it in a secure way
- Tunneling is a technique in which packets of one type are repackaged into new packets
  - The original packets become a payload into which the existing network does not look at

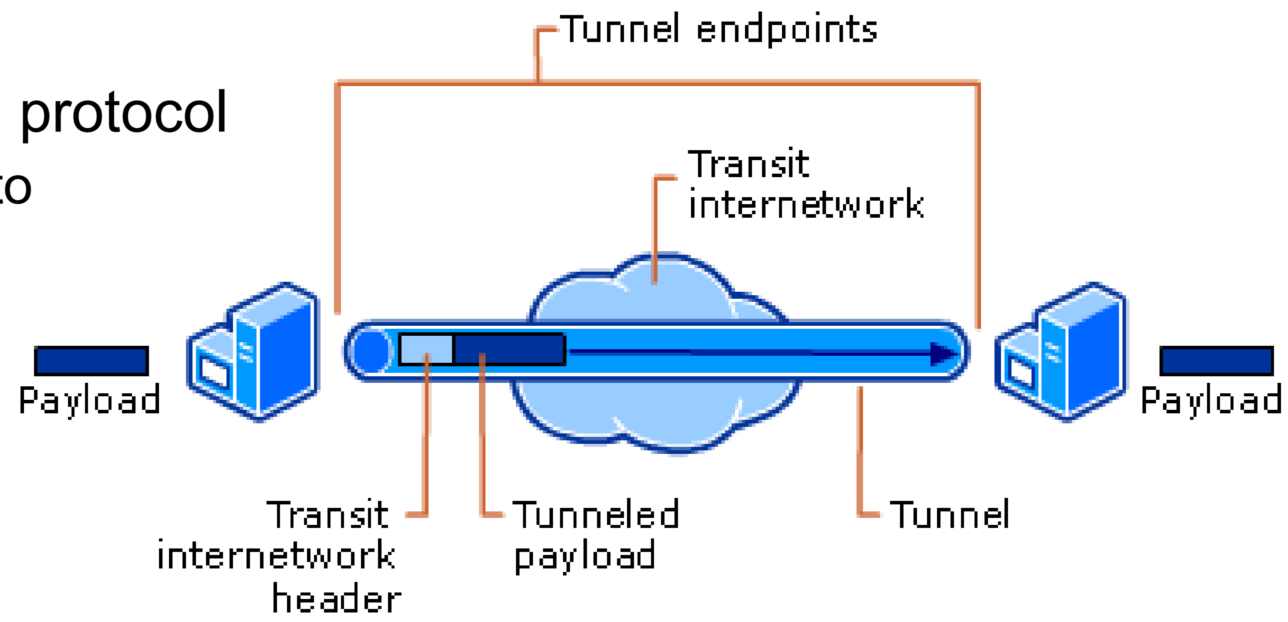


Tunneling protocols - terminology

- Transmitted protocol (passenger protocol)
  - A protocol whose datagrams we need to tunnel over an existing network
    - IPv4 or IPv6
- Auxiliary tunneling protocol (carrier protocol)
  - A protocol whose header is appended to the datagrams of the transmitted protocol
  - It allows you to identify the transmitted protocol, implement security, authentication and other functions
    - With us GRE
- Carrier protocol (transport protocol)
  - The protocol on which the existing network works and inside of which we transport the datagrams of the transmitted protocol wrapped in the auxiliary tunneling protocol
    - IPv4 or IPv6

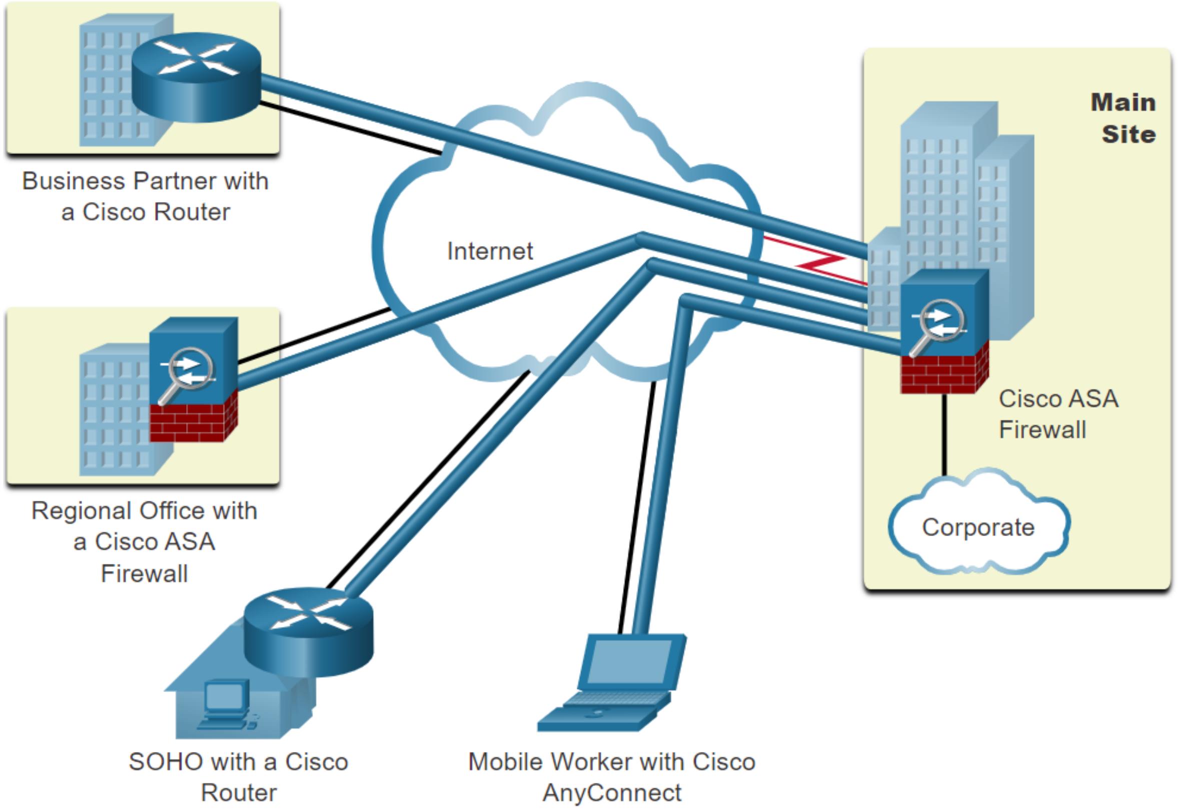
Tunneling protocols

- Tunneling can be performed with or without an auxiliary tunneling protocol
- Tunneling with the Auxiliary Tunneling Protocol
  - Tunneled (passenger) packets are wrapped in the auxiliary tunneling protocol header, and then re-inserted into new packets
  - Authentication options, multiple tunnels between the same devices, different types of tunneled protocols, encryption
  - Potentially higher overhead
  - For example: GRE, L2TP, PPTP
- Tunneling without an auxiliary tunneling protocol
  - Tunneled packets are inserted directly into new packets
  - Minimum overhead
  - Limited options
  - For example: IP-in-IP, IPv6-in-IPv4



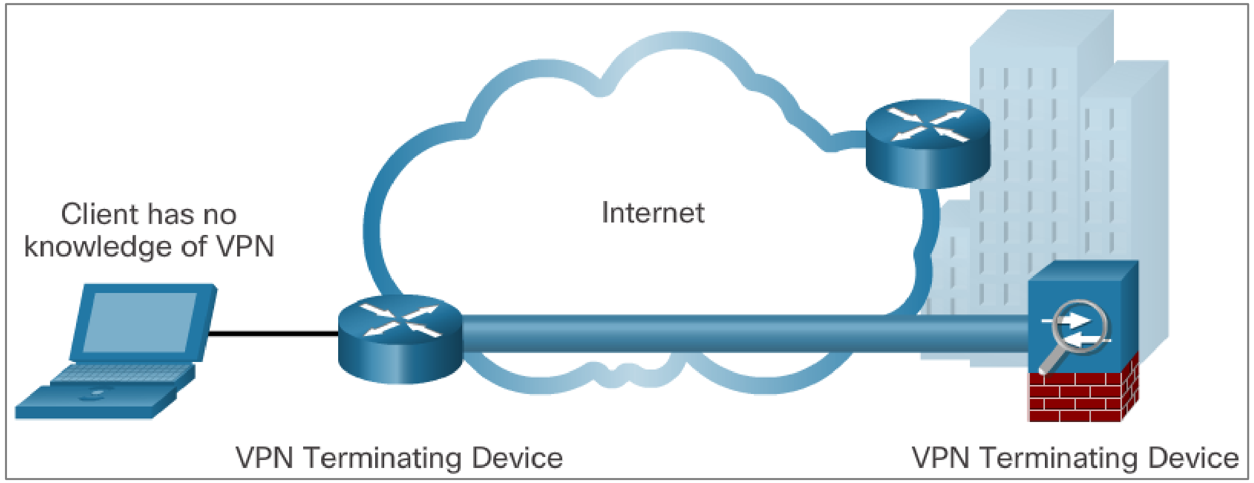
What do we need to implement VPN?

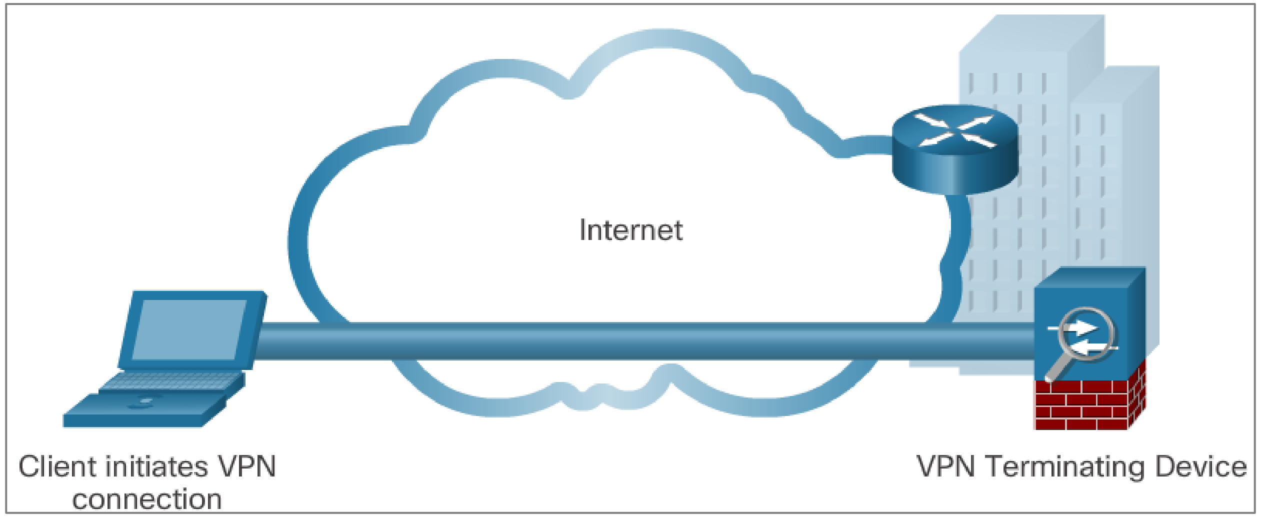
- What we need to implement VPN?
  - VPN gateway/s
    - Network devices between or against which VPN tunnels are created
      - They implement the support of the necessary VPN protocols in their OS
    - Example:
      - Router, Firewall, Cisco Adaptive Security Appliance (ASA), VPN Server, VPN concentrator, etc.
      - Ideally, the VPN gateway should have hardware encryption support
  - VPN client
    - VPN software running on a computer/terminal OS



VPN types in terms of deployment options

- Site-to-Site VPN
  - It interconnects a VPN gateway to a VPN gateway
    - Entire networks, e.g. branches with headquarters
  - All activities implemented on VPN gateways
    - No software is required on the end PCs, they have no idea about a VPN
- Remote Access VPN
  - Used to connect individual PCs to the VPN gateway,
    - e.g. for access to headquarters
  - Client-based or non-client based





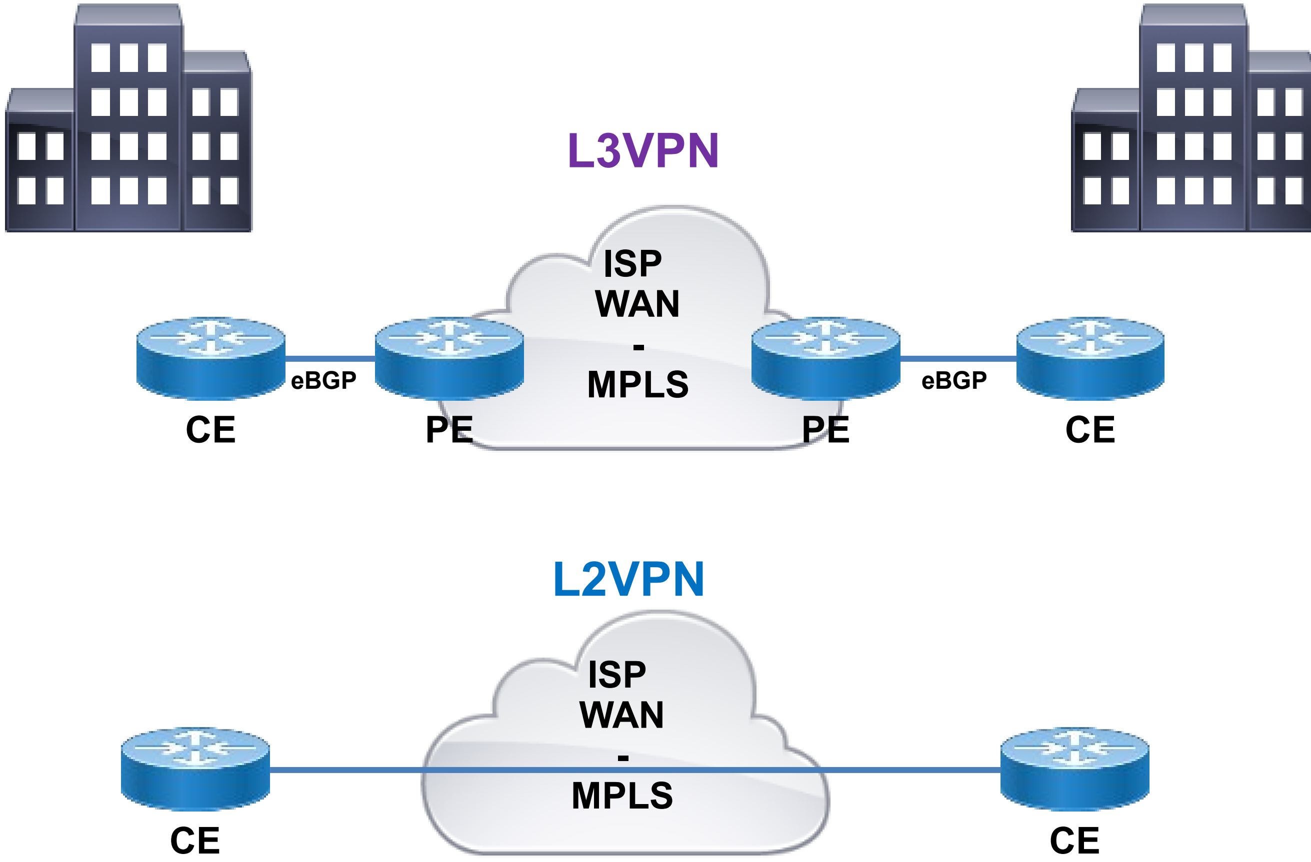
VPN types in terms of who manages them

- **Enterprise VPN**
  - The establishment and removal of VPN is managed by the company itself
    - By own employees on their VPN devices
  - Site-to-site VPN solution technologies
    - GRE (unencrypted)
    - IPSec (encrypted)
    - GRE over IPSec (encrypted)
    - Cisco Dynamic Multipoint Virtual Private Network (DMVPN)
    - Cisco IPsec Virtual Tunnel Interface (VTI)
  - Remote-access VPN solution technologies
    - Using a VPN client: IPSec VPN
    - With or without a VPN client: SSL VPN

- **Private VPN services provided by SP / ISP**
  - Setting up and managing a VPN service is ordered as a turnkey product from a specific ISP provider
  - We are currently distinguishing
    - Layer 2 MPLS VPN
    - Layer 3 MPLS VPN
    - *Out of CCNA Scope*
  - Legacy, but obsolete solutions
    - Frame Relay, ATM Asynchronous Transfer Mode

Private VPN Services SP (CCNA does not cover)

- ISP => Guaranteed Service
  - Stability, speed, loss, safety, etc.
    - For this purpose, the ISP builds its own WAN only for customers of this service
  - Rather for companies => price
    - E.g. only setting up 34Mbps MPLS service
      - 9950 Euro with DPH
- Types of private VPN services
  - L3VPN (via MPLS)
    - Customer routers exchange updates with ISP routers
  - L2VPN (via MPLS)
    - Customer routers exchange updates directly



VPN basics

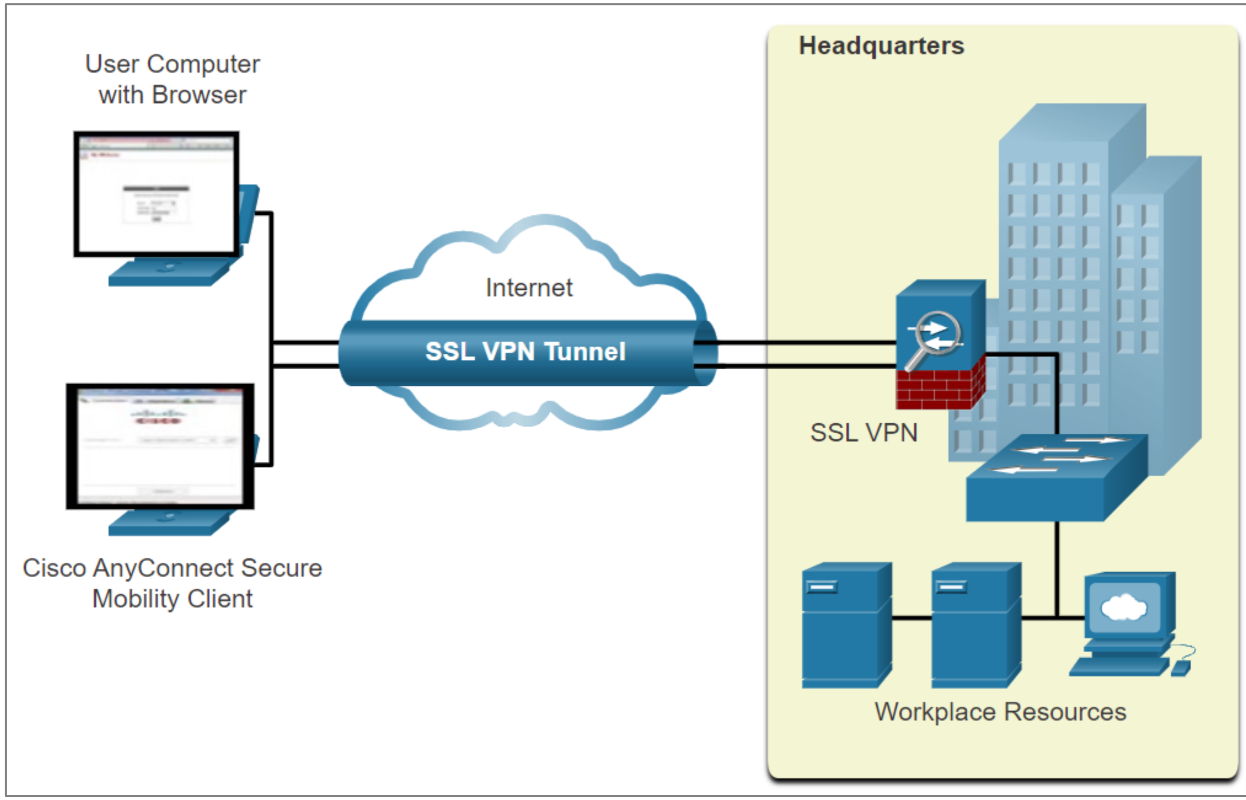
- VPN advantages
  - Cost savings
    - Teleworking, mobility, use of cheap Internet for secure access to the corporate network
  - Scalability
    - Easy management of adding / removing users and networks through the creation of a new tunnel
  - Compatibility, resp. independence from broadband Internet connection technologies
  - Security
    - When using encrypted solutions with authentication (or solutions from ISPs) a high level of communication security

Types and possibilities of VPN solutions - a closer look

Remote-Access VPN

Remote-Access VPN

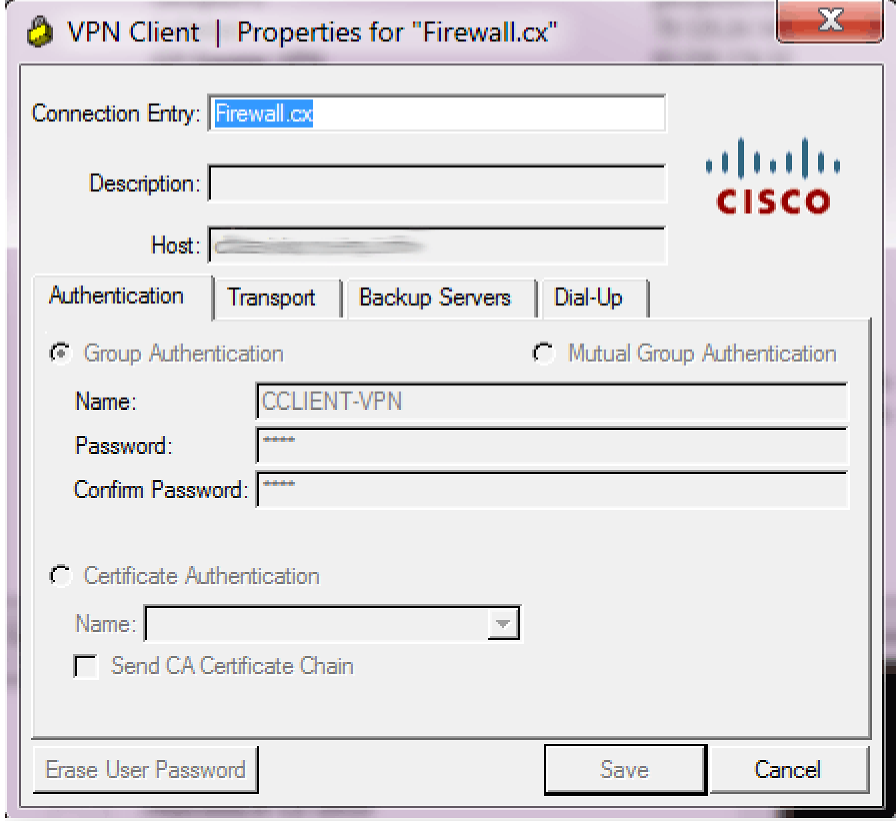
- Primarily intended for mobile workers / homeworkers / company partners
  - The user connects from his NB / mobile / tablet to the employer's network
  - Creates a tunnel from your device to configured VPN gateway
    - By starting the client application
  - VPN offers
    - Access to specific services behind a VPN gateway,
      - Access to web / file server, etc.

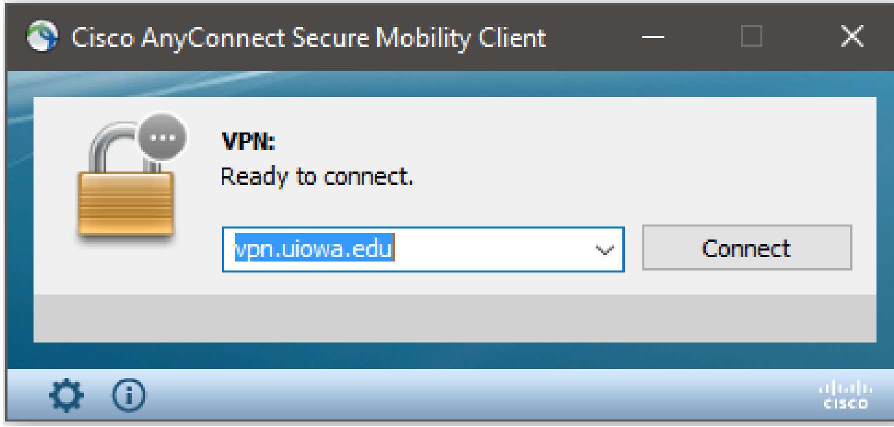


- Secure type of dynamic VPN
  - Created only for a certain time
  - When the required action is completed, the user turns it off

VPN gateway connection options

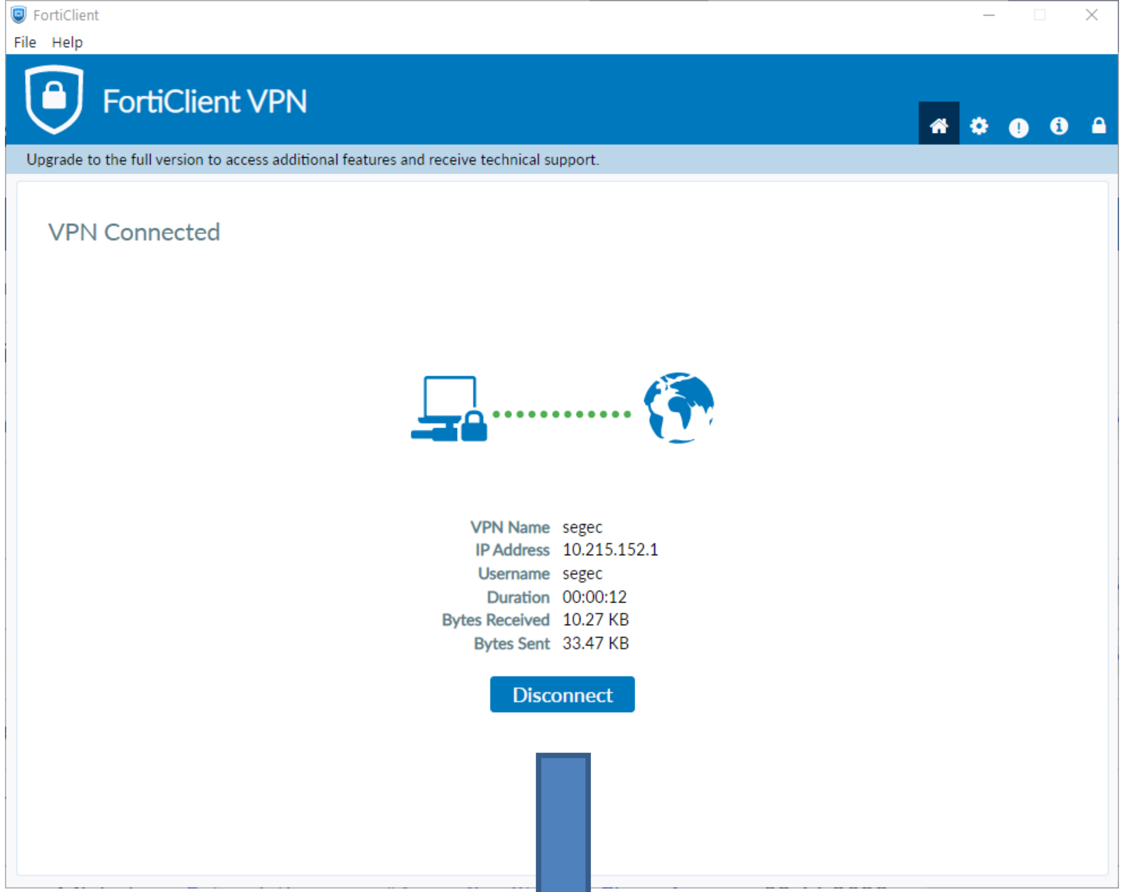
- Two different Remote Access VPN solutions
- **Client-based VPN – L3 IPsec VPN and L4 SSL VPN**
  - The installed and configured VPN software is required on the end device
    - IPSec examples: Cisco IPSec client (older version), Cisco AnyConnect Secure Mobility Client (current software), built-in IPSec in Win 10 (KIS) (L2TP over IPsec)
    - SSL VPN:Cisco AnyConnect, FortiClient, other
  - Disadvantages:
    - The VPN client must be installed and configured correctly
    - Each time a user wants to connect, he must start the client
  - Advantages:
    - Works for all services from L3 up
- **Client-less L4 VPN – SSL VPN**
  - No need to install a client => SSL VPN (TLS - Transport Layer Security)
  - Uses PKI infrastructure of keys and certificates
  - Currently popular, but only suitable for some services from L4 up
    - Primarily accessible through a Web browser





Connectivity

- Routing
  - **Full routing**
    - Everything is routed to remote GW and then out
    - Good to enforce company polices
  - **Split routing**
    - For local breakout
    - Only some prefixes are routed to remote site
    - Other internet access go out locally
      - Especially needed in Cloud era



Command Prompt

IPv4 Route Table
===
Active Routes:

Network Destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.108	281
10.50.10.164	255.255.255.255	10.215.152.2	10.215.152.1	1
10.215.152.1	255.255.255.255	On-link	10.215.152.1	257
10.244.95.10	255.255.255.255	10.215.152.2	10.215.152.1	1
10.244.100.10	255.255.255.255	10.215.152.2	10.215.152.1	1
10.244.100.11	255.255.255.255	10.215.152.2	10.215.152.1	1
37.61.167.132	255.255.255.255	192.168.10.1	192.168.10.108	25

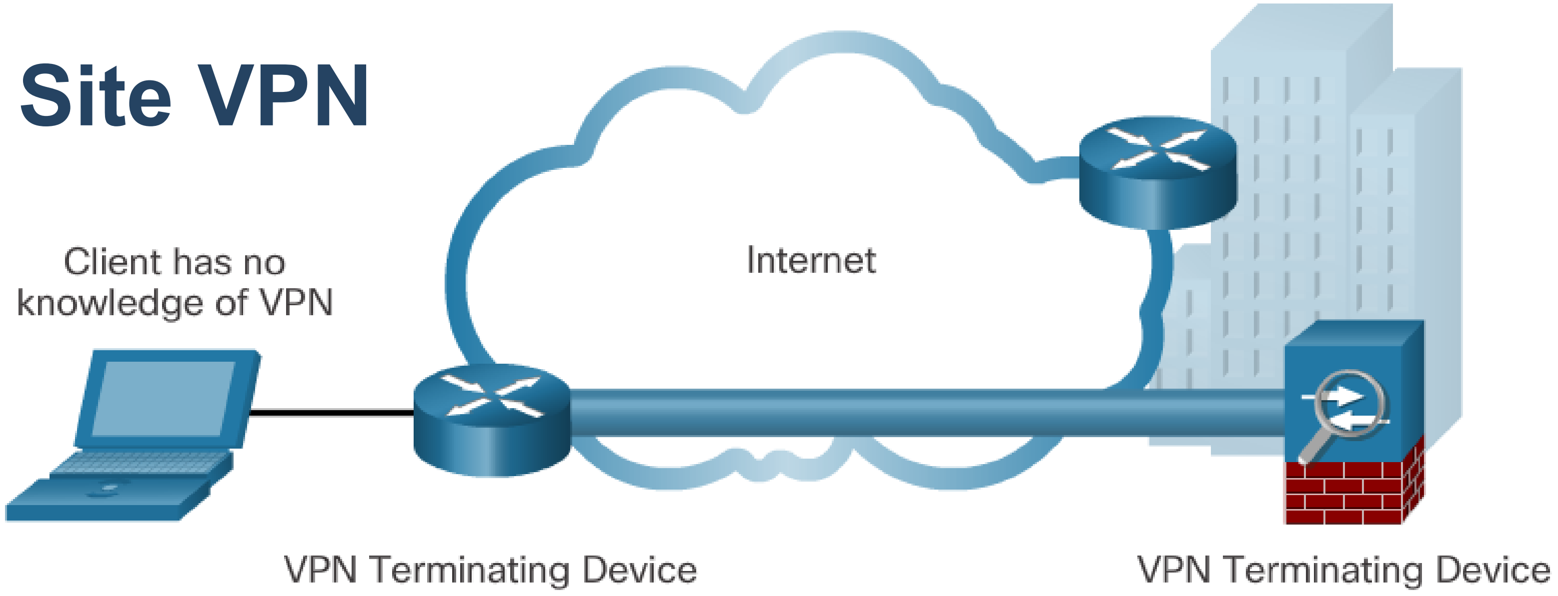
New routes

Comparison IPsec vs. SSL VPN

Attribute	IPSec	SSL
Application support	**Extensive** - support for all applications from L3 up	**Limited** - support for web applications and file sharing only
Authentication level	**High** - two-way authentication with passwords or certificates	**Medium high** - one and two way authentication
Encryption level	**High** - key size from 56 to 256 bits, many types of algorithms	**Medium to high** - key size from 40 to 256 bits, fewer algorithm types
Connection complexity	**Medium** - because it requires client installation and configuration	**Low** - just a web browser
Connection options	**Limited** - only a device with a client, client support for different OS is limited	**Extensive** - can use any device with a browser

Site – to – Site VPN

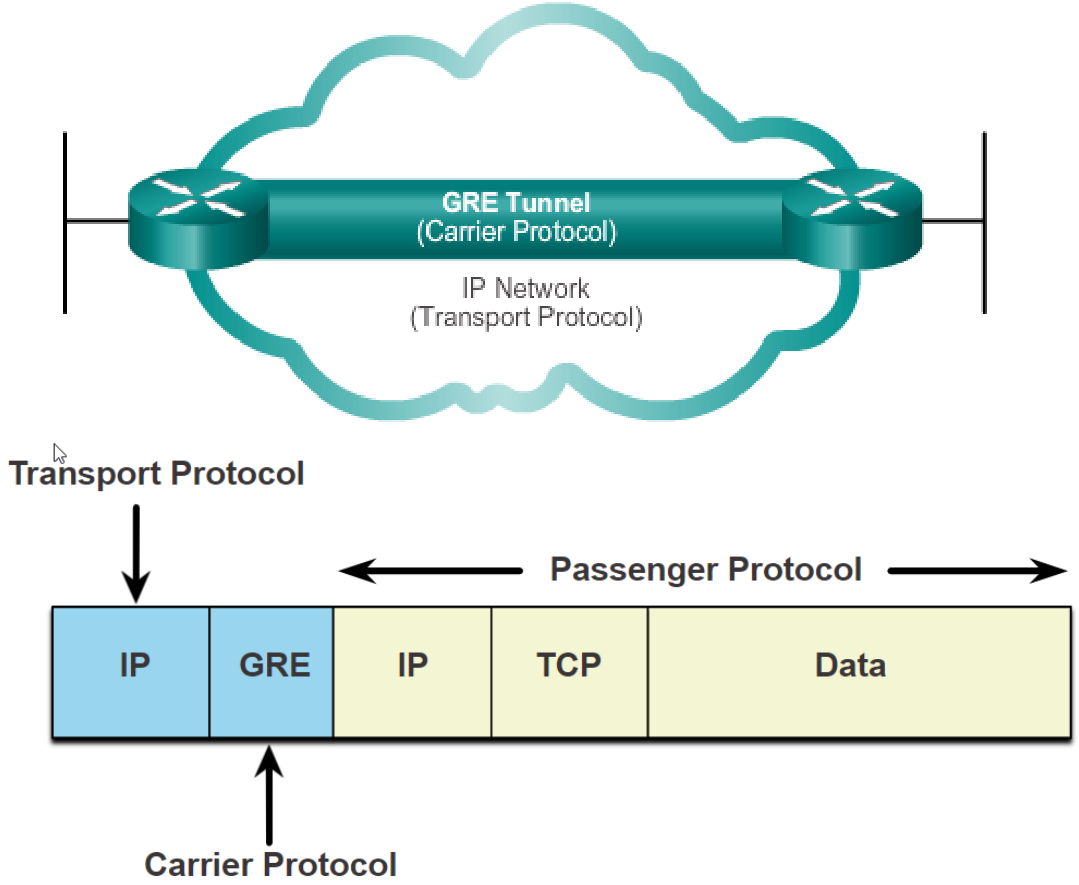
Site – to – Site VPN



- It uses the concept of tunneling between two network devices
- Cisco Site-to-Site VPN Solutions
  - GRE
    - unencrypted, so it is no longer recommended
  - IPSec – we do extra
    - encrypted VPN, routing problem
  - GRE over IPSec:
    - solves routing problem, config. overhead
  - Cisco Dynamic Multipoint Virtual Private Network (DMVPN):
    - addresses overhead GREoverIPsec configuration
  - IPsec Virtual Tunnel Interface (VTI)
  - SD-WAN

Generic Routing Encapsulation – GRE

- GRE is a Layer 3 auxiliary tunneling protocol
  - It supports different types of tunneled packets
    - E.g. IPv4, IPv6, IPX...
  - Creates a virtual point-to-point connection between a pair of routers
  - It also allows you to transfer multicast traffic
- GRE characteristics
  - is stateless, without data flow control
  - GRE does not provide security
    - no confidentiality, authentication or integrity checking
  - It is inserted into IP packets, the overhead of GRE tunnels is 24B
    - 20B for the new IP header and 4B for the GRE header
  - It creates a "normal" interface with an IP address on the router
    - It can therefore be inserted into the routing process

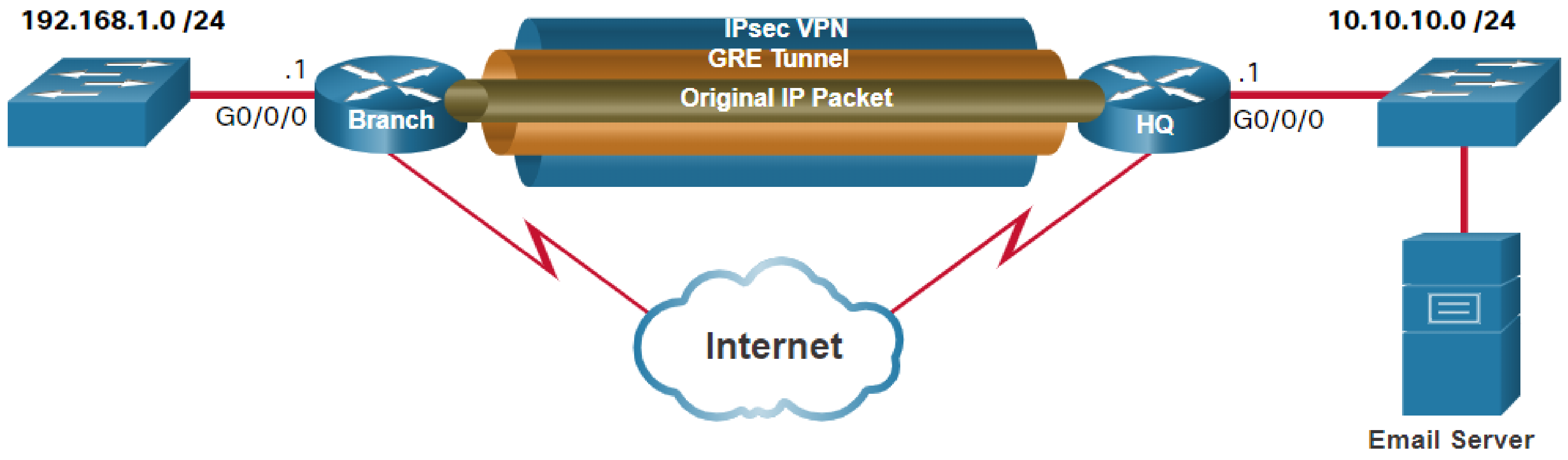


IPsec

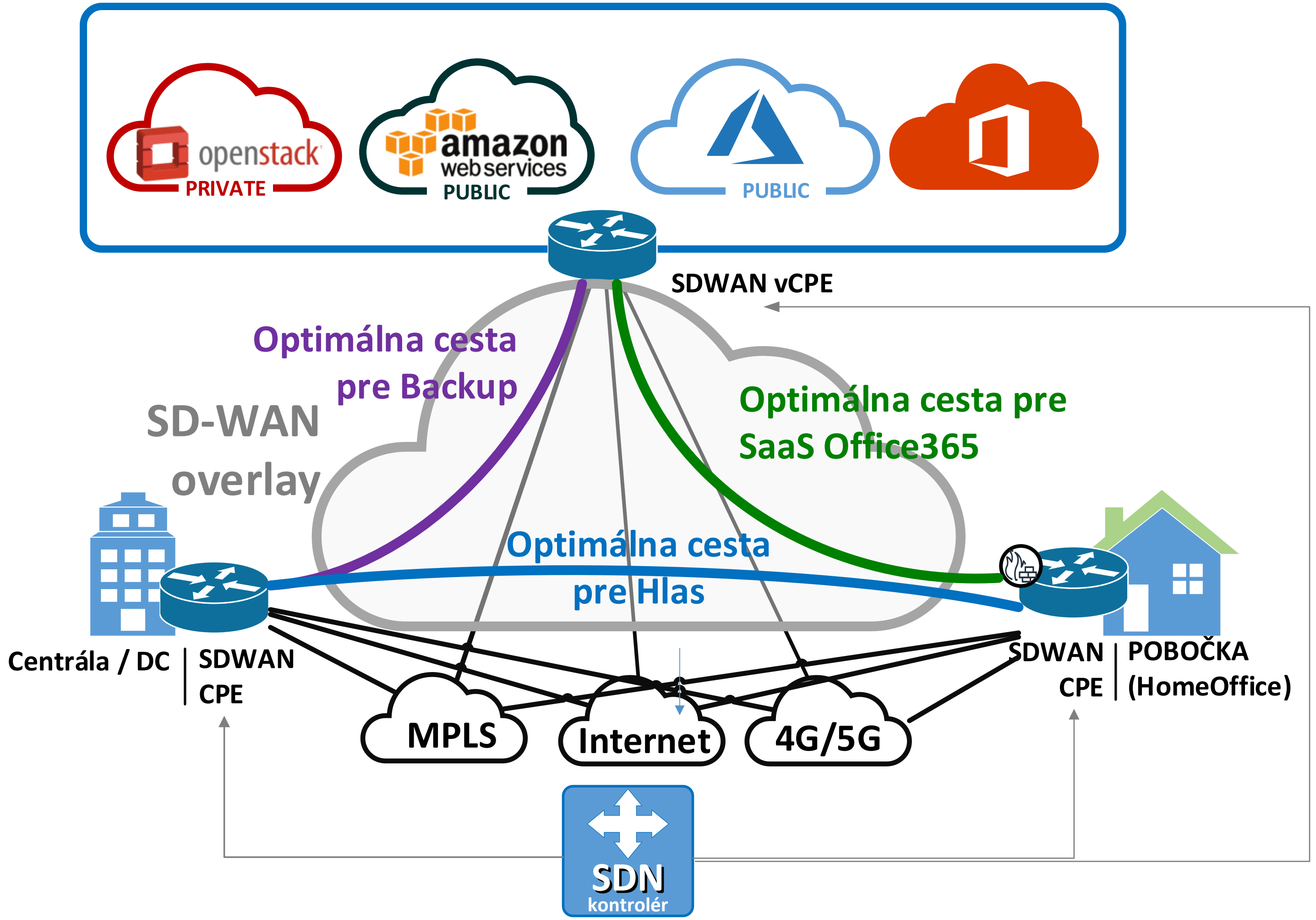
- Presented on next few slides….

GRE over IPsec

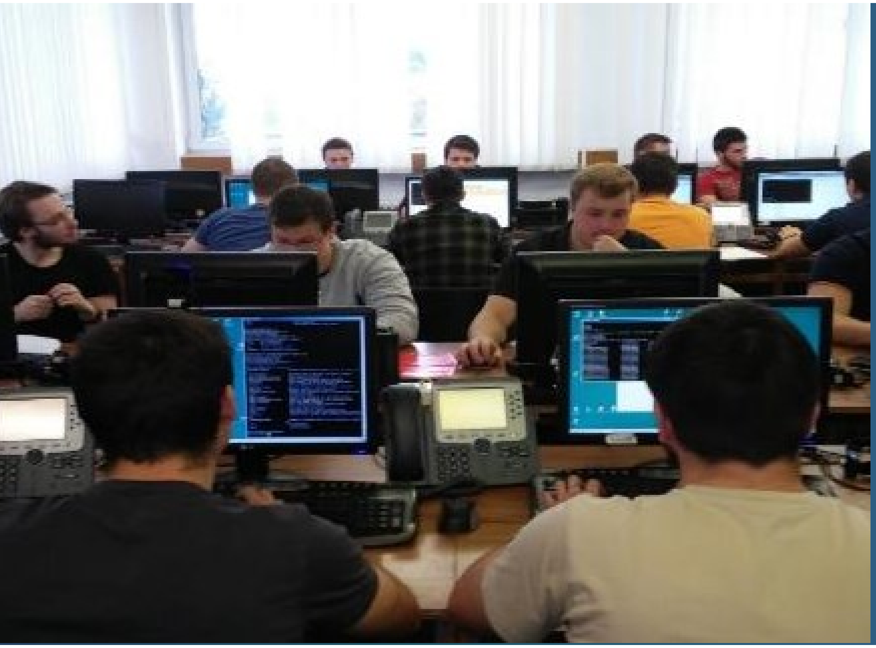
- In reality, we have the following problem:
  - GRE
    - It supports routing via the GRE interface
    - However, it is unencrypted => it is not recommended to use it separately in a "living" environment
  - IPSec
    - is encrypted
    - but in the common configuration it does not have an interface in Cisco IOSe
      - Unable to start routing over it
  - Solution => connect and deploy both => GRE over IPsec



SD-WAN – application and „*cloud-centric*“ solution



- **SD-WAN = SDN + cloud + WAN siete + virtualizácia + automatizácia + bezpečnosť**
  - Autonómne riadenie siete podľa požiadaviek aplikácii (SLA a kvalitu služby (QoS))



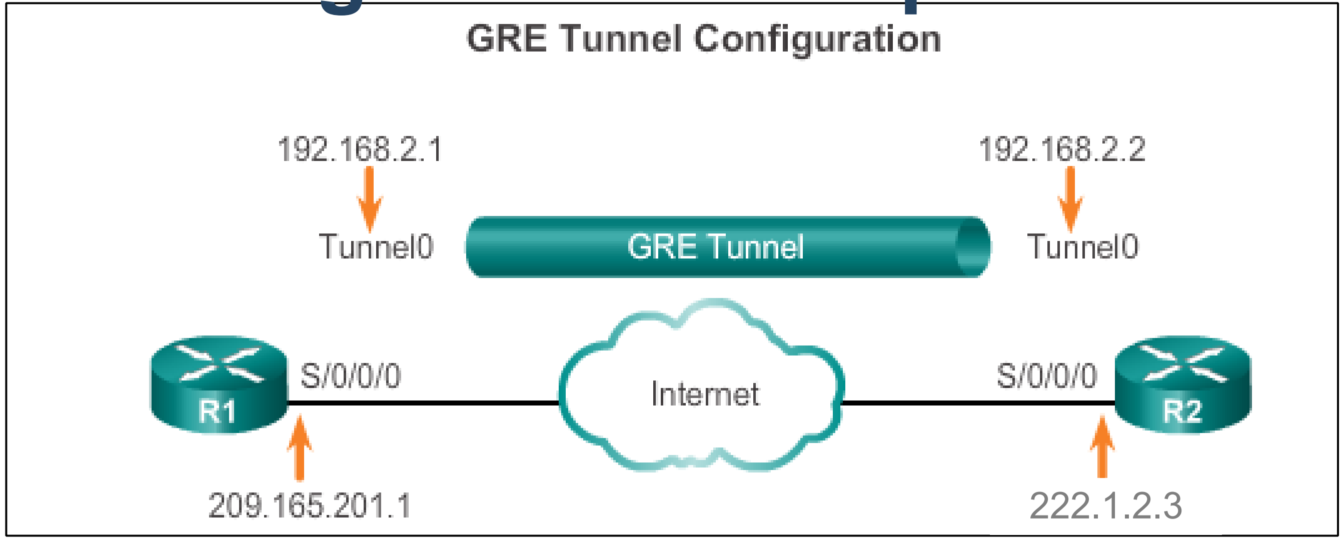
Simple GRE configuration
(Generic Routing Encapsulation)

Site-to-site GRE tunnels

GRE tunnel configuration

- GRE tunnels are represented on a router by a virtual Tunnel interface
- The Tunnel interface must be manually defined
  - Own IP address (like any other interface)
  - Sender's IP address
    - Sending interface or IP address of the sending interface
  - The IP address of the carrier of the carrier packets
  - Tunneling mode
- A pair of Tunnel interfaces on different routers that communicate must meet these criteria:
  - Tunnel's own IP addresses must be on the same network (as well as on a pair of interconnected interfaces)
  - The sender's and recipient's IP addresses must correspond to each other (the sender's IP on one router must match the recipient's IP on the other router and vice versa)
- The default bandwidth of the Tunnel interface is 9 Kbps
  - Think of EIGRP or OSPF metrics
  - It is recommended to increase it to a realistic value

GRE tunnel configuration example




```
hostname Bratislava
!
interface Serial0/0/0
 ip address 209.165.201.1 255.255.255.0
 no shut
!
interface Tunnel0
 bandwidth 1000
 tunnel source s0/0/0
 ! Or
 ! tunnel source 209.165.201.1
 tunnel destination 223.1.2.3
 tunnel mode gre ip ! OPTIONAL
 ip address 192.168.2.1 255.255.255.0
!
router ospf 1
 network 192.168.2.0 0.0.0.255 area 0
```

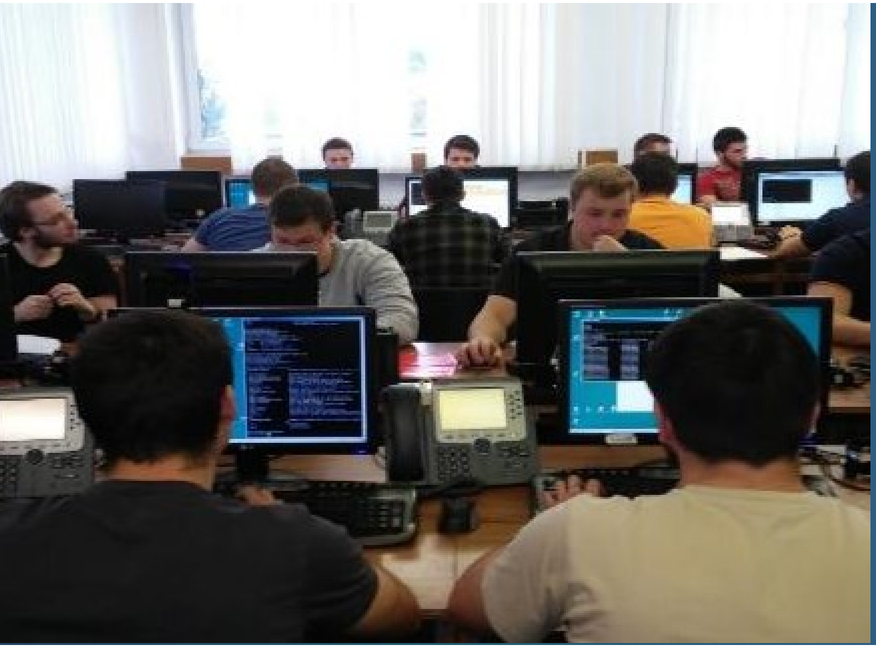
```
hostname Kosice
!
interface Serial0/0/0
 ip address 222.1.2.3 255.255.255.0
 no shut
!
interface Tunnel7
 bandwidth 1000
 tunnel source s0/0/0
 ! Or
 ! tunnel source 222.1.2.3
 tunnel destination 209.165.201.1
 tunnel mode gre ip ! OPTIONAL
 ip address 192.168.2.2 255.255.255.0
!
router ospf 1
 network 192.168.2.0 0.0.0.255 area 0
```

Tunnel interface status

- GRE Tunnel interfaces will be "up, protocol up" if all of the following conditions are met at the same time
  - The interface has a defined source and destination commands **tunnel source**, **tunnel destination**
    - The tunnel has a valid source and destination IP defined
  - The real interface from which we borrow the source IP in the **tunnel source** command is in the "up, protocol up" state
    - The source IP address must be alive
  - In the routing table, we can find the path to the opposite end of the tunnel defined by the command **tunnel destination**
    - According to our RT, the destination IP address must be reachable
  - If GRE Keepalive is enabled, the other party responds to our Keepalive packets
    - The interior of the transport network must be able to deliver packets between the ends of the tunnel

Verification

```
Branch# show interfaces tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.2.1/24
  MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 209.165.201.1, destination 223.1.2.3
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transport MTU 1476 bytes
  Tunnel transmit bandwidth 1000 (kbps)
  Tunnel receive bandwidth 1000 (kbps)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec

<output omitted>
```



IPsec VPN – Components and function

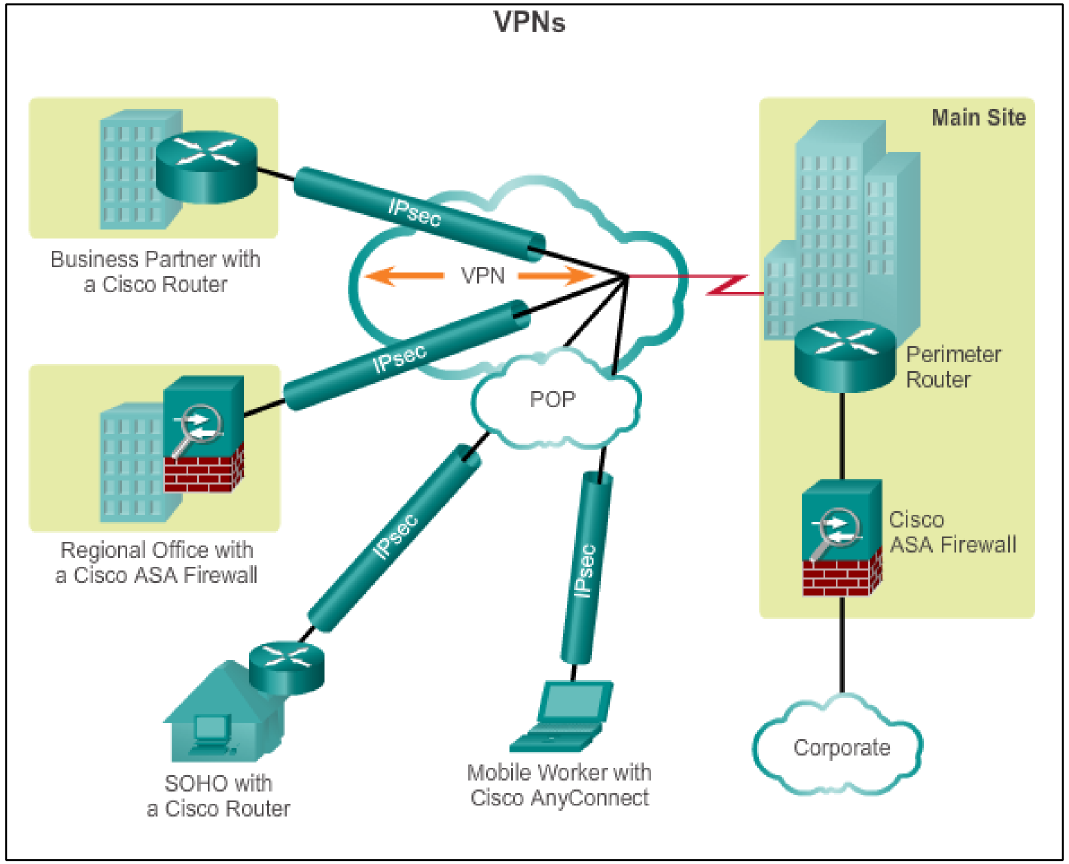
Upon completion of this section, you should be able to:

- Describe the IPsec protocol and its basic functions.
- Compare AH and ESP protocols.
- Describe the IKE protocol.

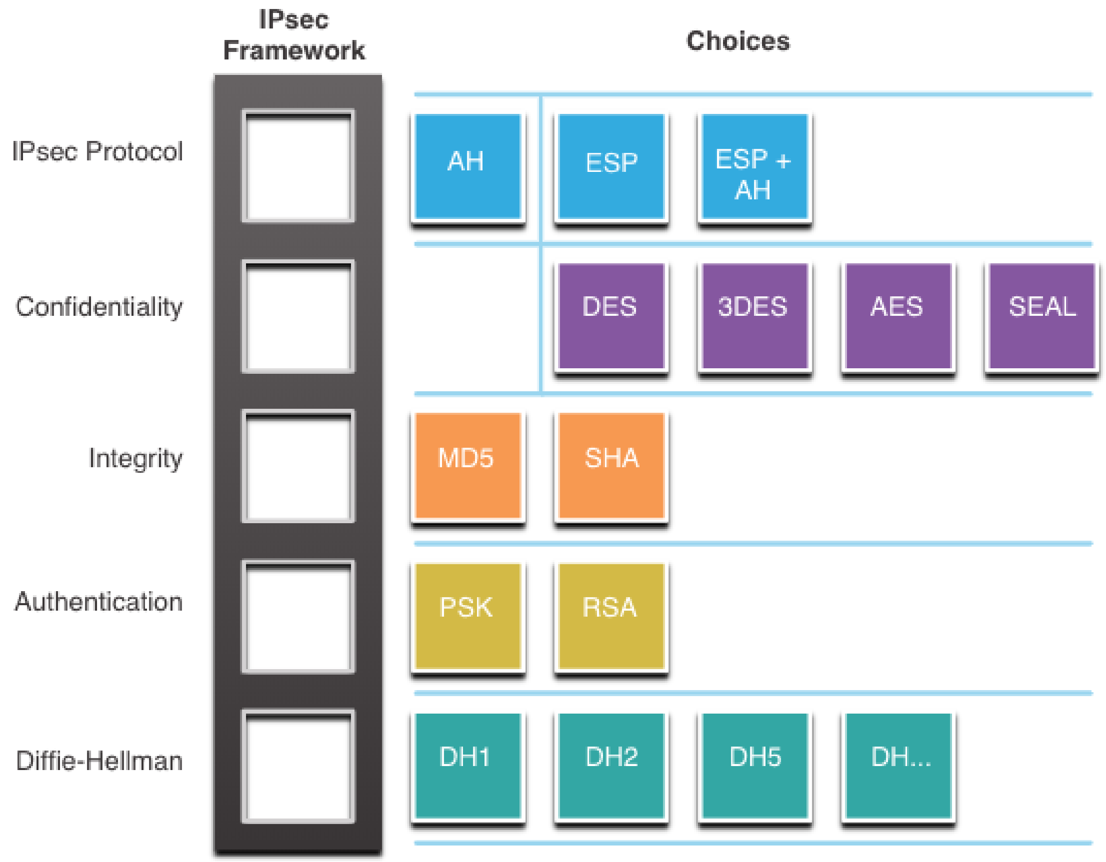
Introduction to IPsec

IPsec VPNs

- IPsec is a series of IETF standards that describe how IP packets are securely transmitted
- It does not relate to a specific algorithm / mechanism
  - Encryption, authentication or other security algorithm / mechanism
  - Able to use various existing and future mechanisms
- He works as a tunneling mechanism on L3 ISO OSI
  - It thus secures L3 packets and evrything over L3
    - in IPv4 added, requires a client
    - in IPv6 native component
  - Type of connection
    - Site-to-site even remote-access



IPsec Technology



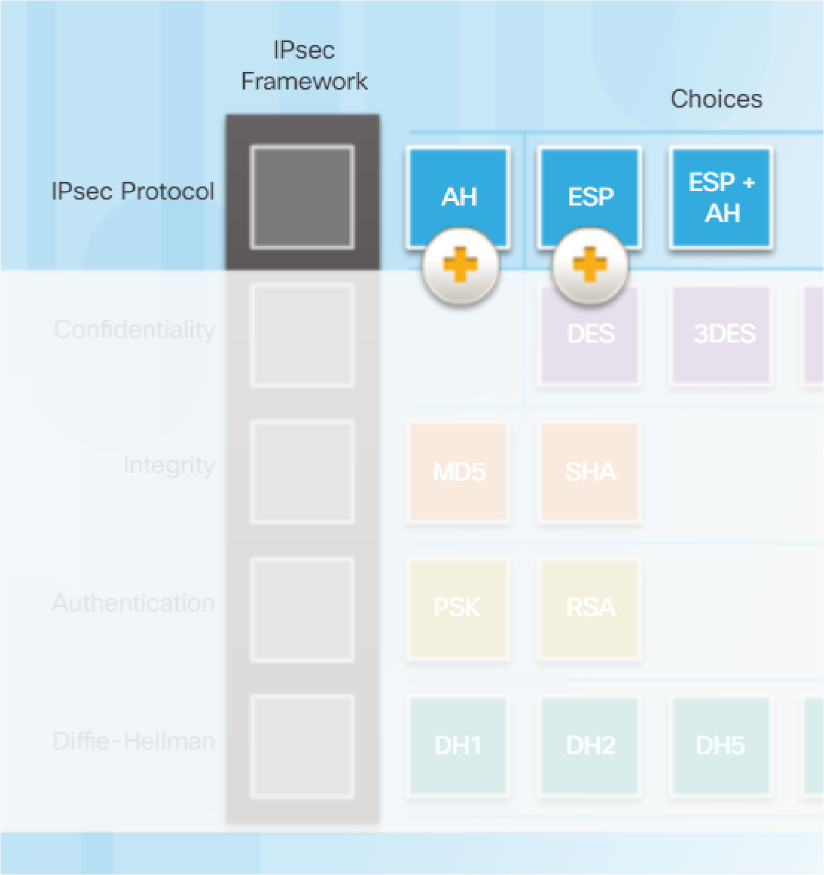
Note: The set of used parameters creates the so-called Security association (SA)

- IPsec provides CIA features
- Four IPsec building blocks
  - **IPsec framework protocol**
    - Packet transmission solutions (ESP, AH, ESP+AH)
  - **Data confidentiality** (**Confidentiality**)
    - Encryption so that the message cannot be decrypted and read (DES, 3DES, AES ...)
  - **Data integrity** (**Integrity**)
    - Proof that the message was not modified
    - Achieved by hashing (MD5 or SHA).
  - **Sender authentication** (**Authentication**)
    - Proof. that the report is not a scam and came from who I think it is.
    - Achieved by authentication (PSK or RSA)
  - and **Diffie-Hellman**
    - Secure exchange of encryption keys

IPsec – protocol framework

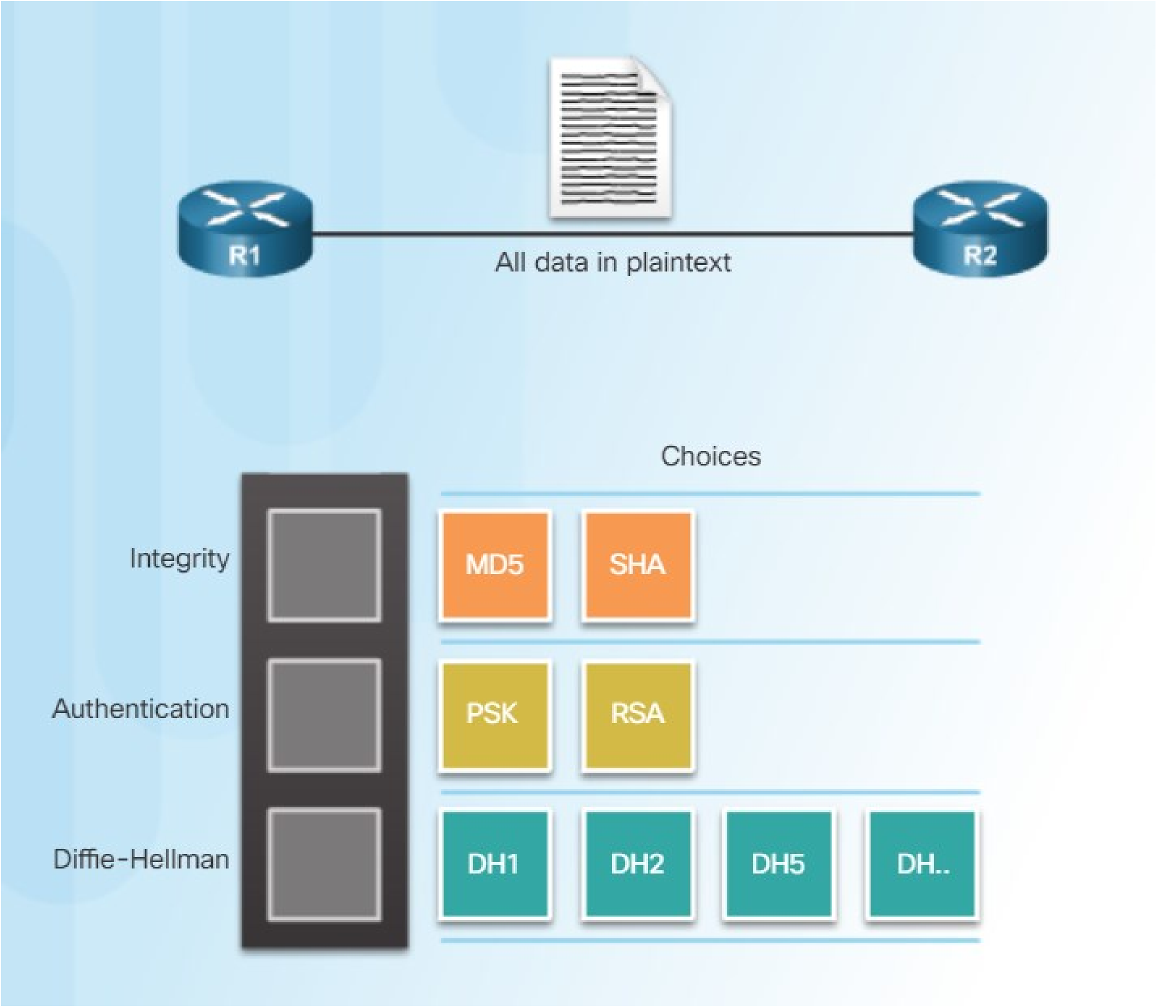
IPsec Protocol Framework (cont.)

- It defines the way IPsec works => two basic ones:
  - **Authentication header (AH)**
    - It protects the complete content of the packet, including the fixed parts of the IP header,
    - It does not provide encryption
    - Does not like NAT (rewrites of IP addresses in the header)
  - **Encapsulation Security Payload (ESP)**
    - Protects the payload of the packet by encryption
    - It does not secure the packet header in transport mode
    - It additionally protects authenticity and integrity
- Notes
  - *The use of AS or ESP determines what other CIA options will be on offer*
  - *AH is currently rarely used, ESP very often (ASA firewalls do not support AH at all)*
  - *AH and ESP can be used simultaneously*

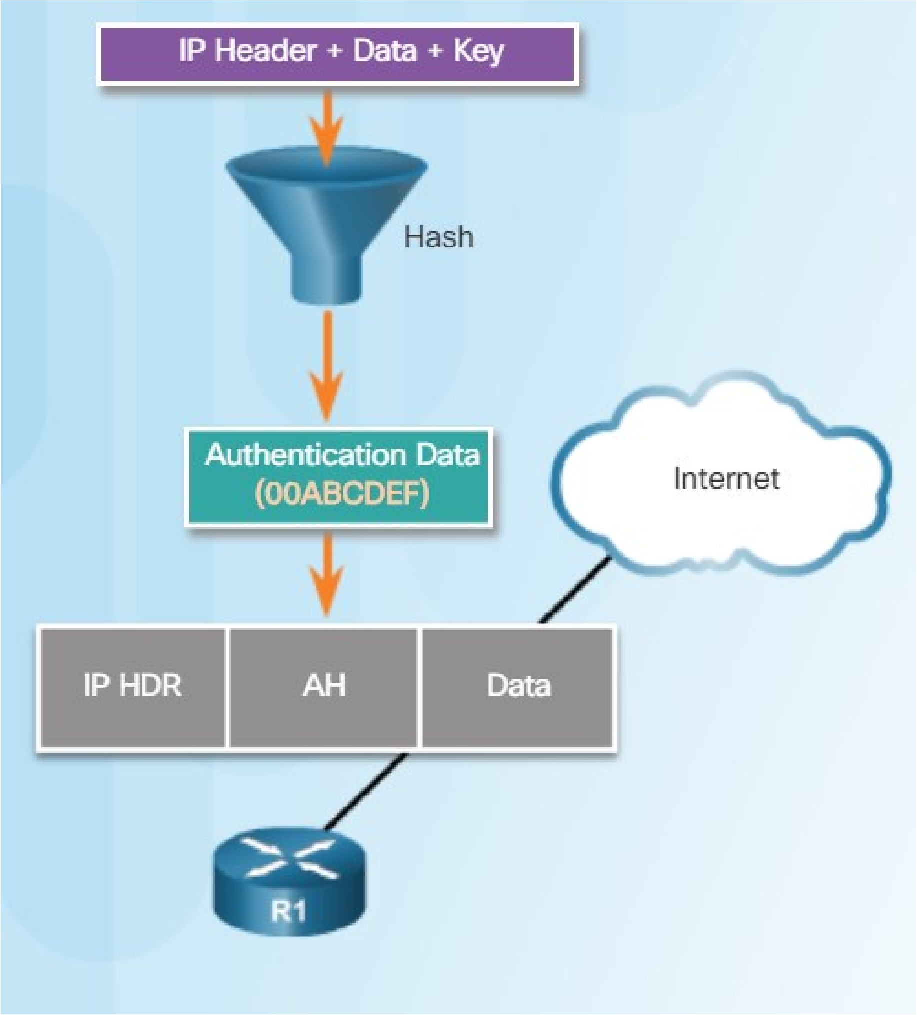


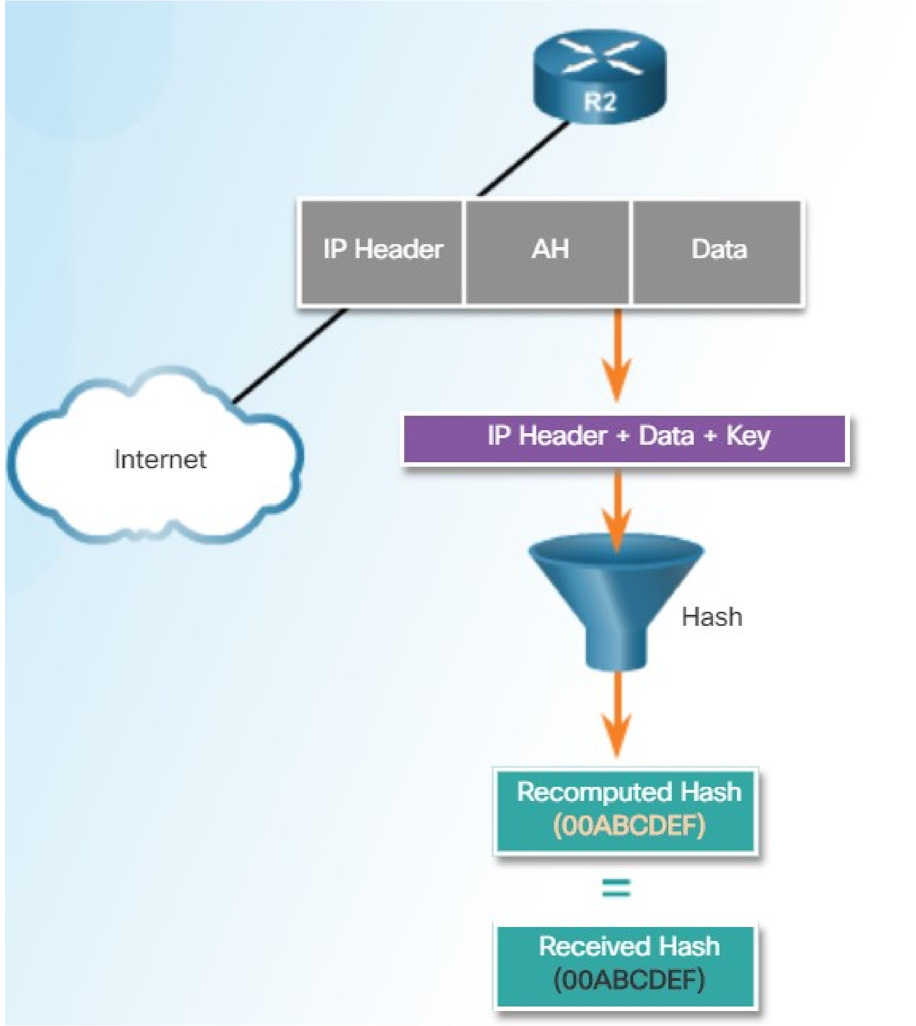
Authentication Header (AH)

- Provide
  - Authenticity and Integrity
- Does not provide
  - Confidentiality
- Transmited in plaintext
- Both sides are using one way hash function
  - Using a shared secret key
- AH functions are applied to the entire packet
  - Except TTL and Checksum



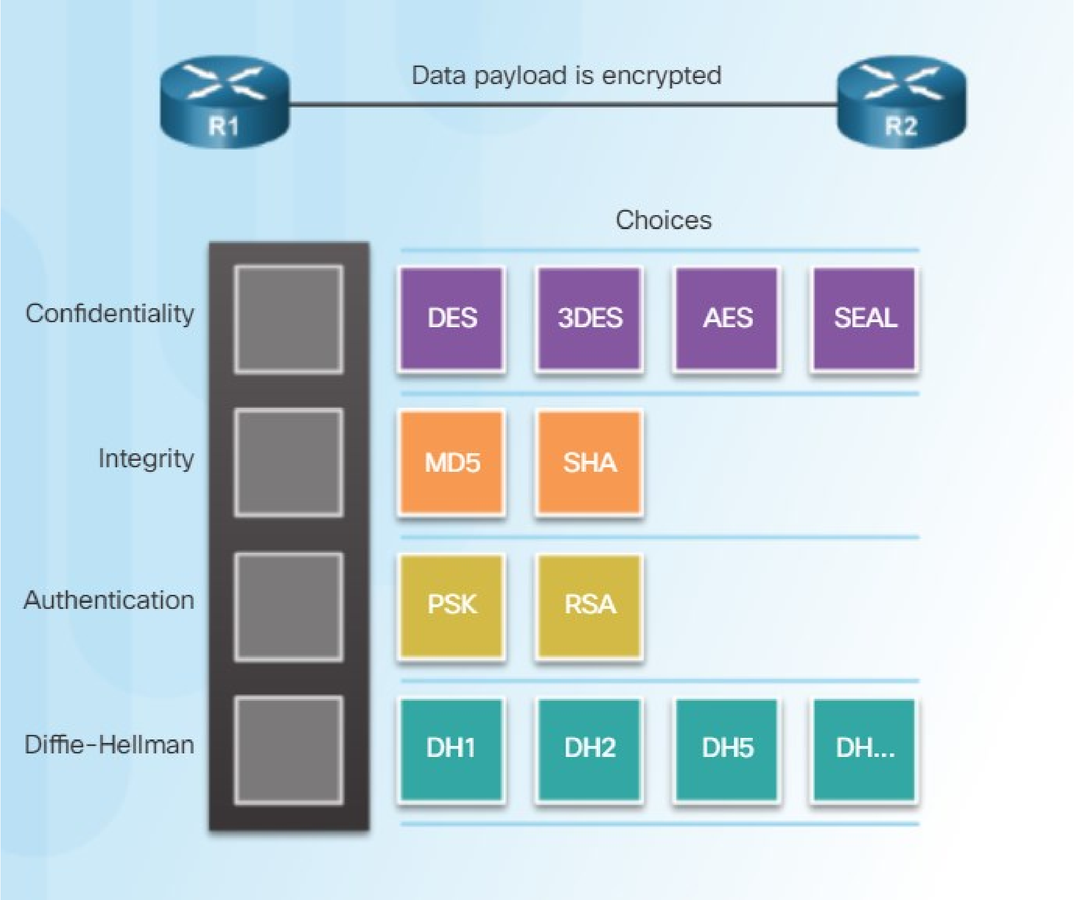
Authentication Header (AH)





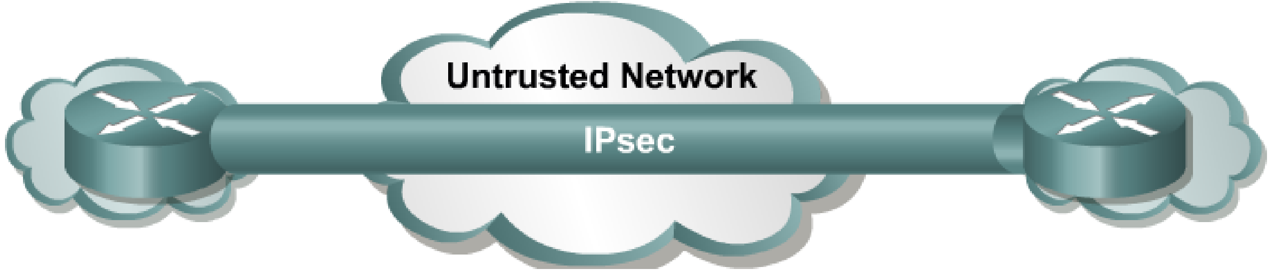
Encapsulation Security Protocol (ESP)

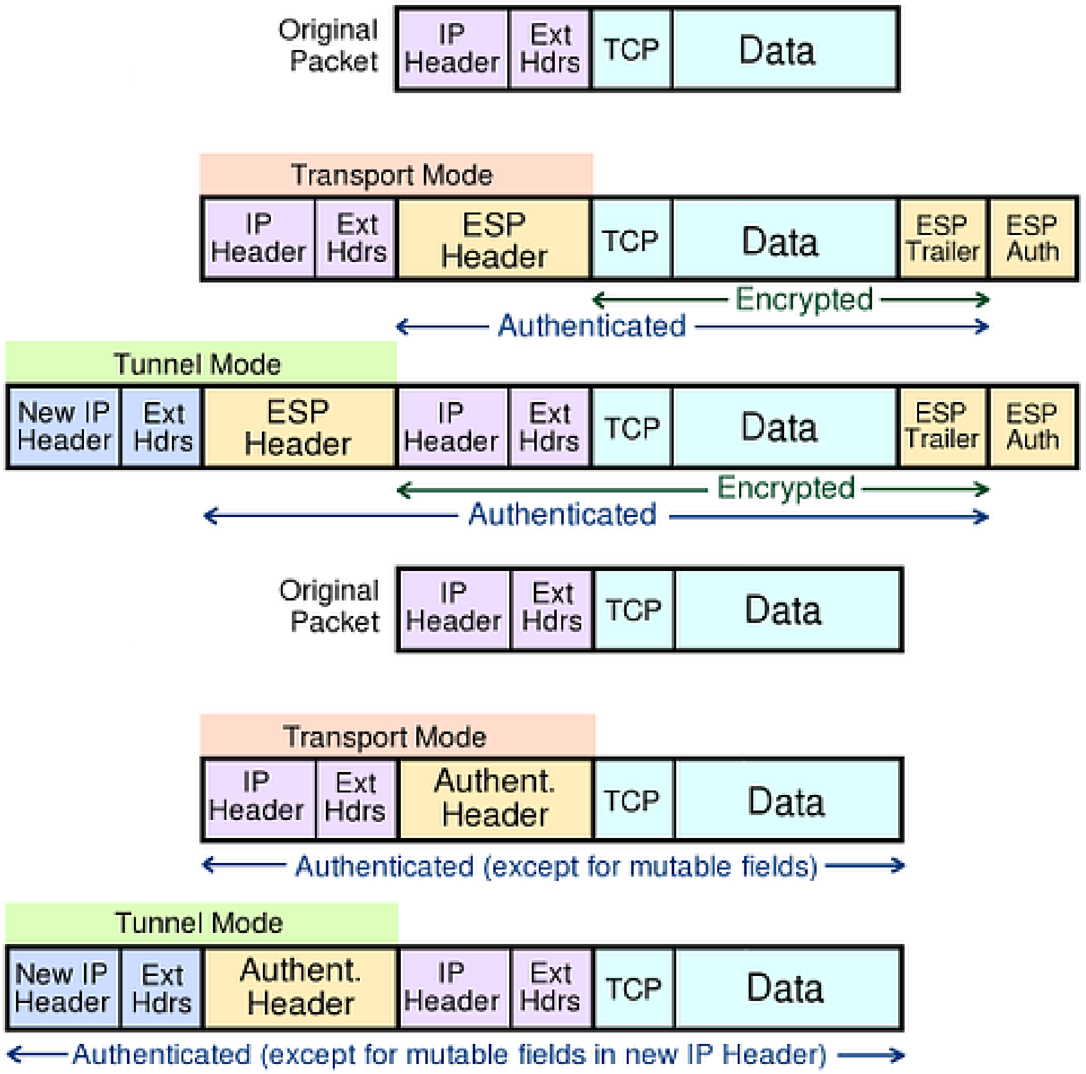
- Provides full CIA
  - Confidentiality by encrypting the payload
  - Authentication
  - and Integrity
- With NULL encrypt algorithm same as AH
- Authentication is performed first
- Encryption is performed second



IPsec - modes of operation

- Tunnel mode
  - Appends a new IP header and tunnels the original IP packet without its modification
  - Preferred method today
- Transport mode
  - Keeps the original IP header
  - On Cisco routers, the transport mode is used only if the sender (author) of the packet is the router itself

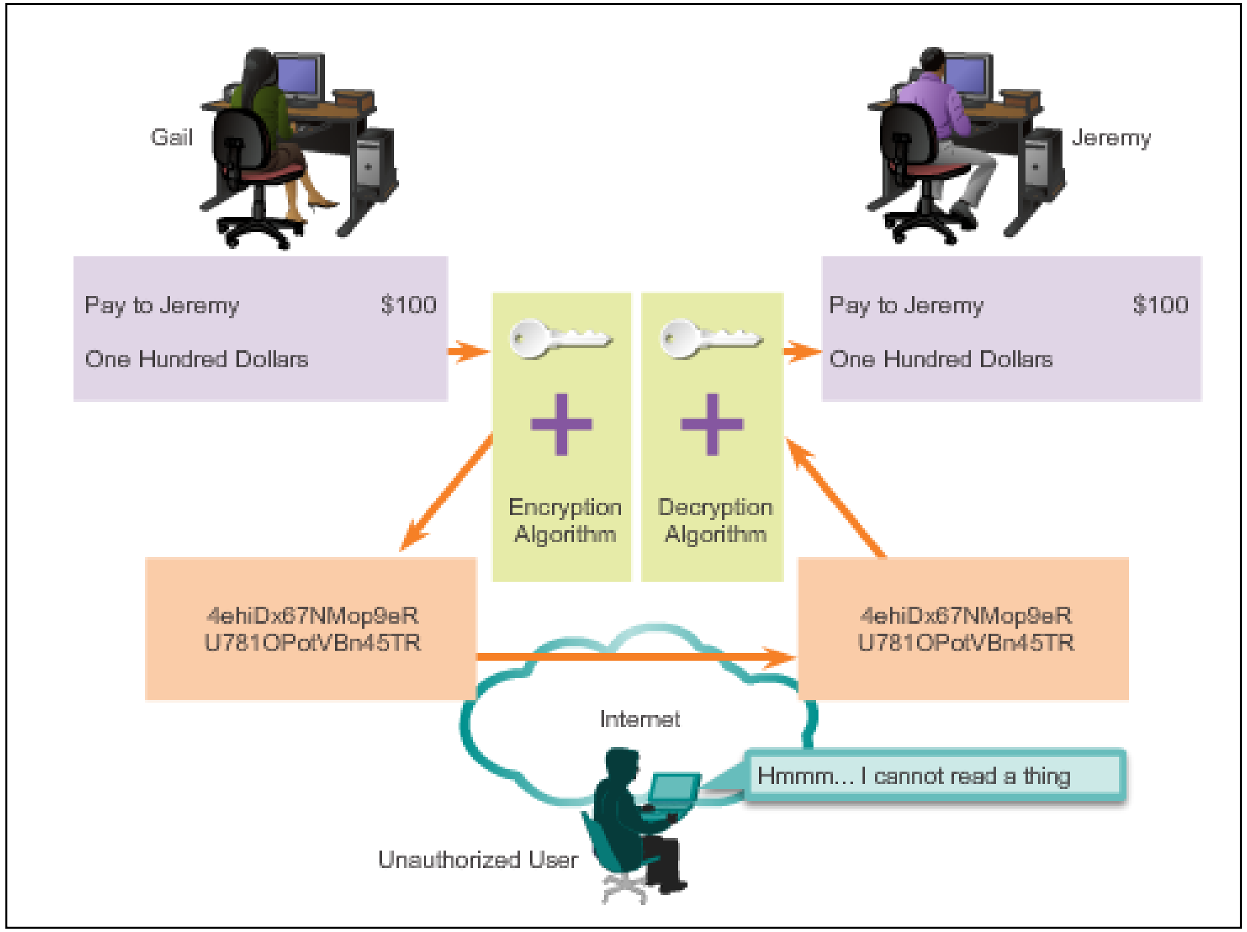




- Note. Pic. From http://www.ipv6now.com.au/primers/IPv6PacketSecurity.php

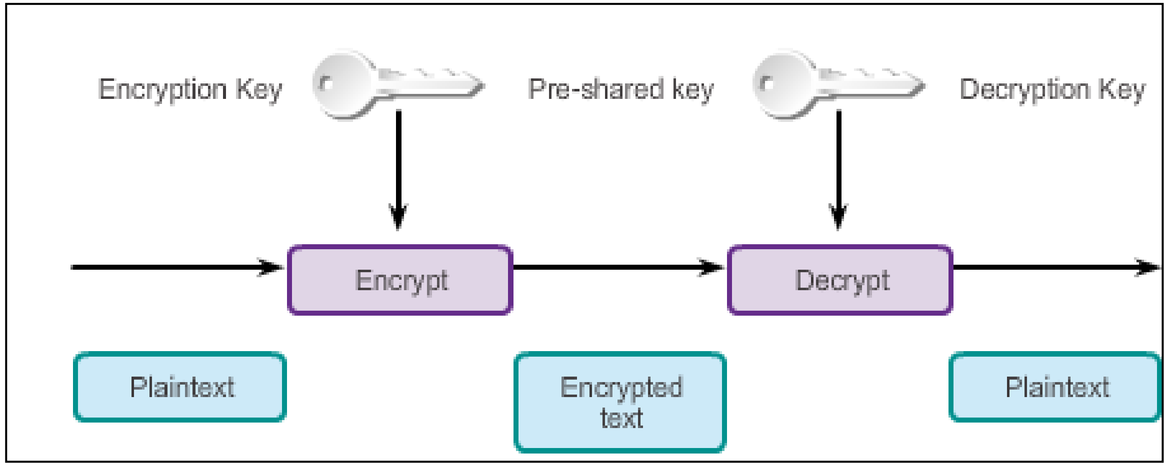
Confidentiality with Encryption

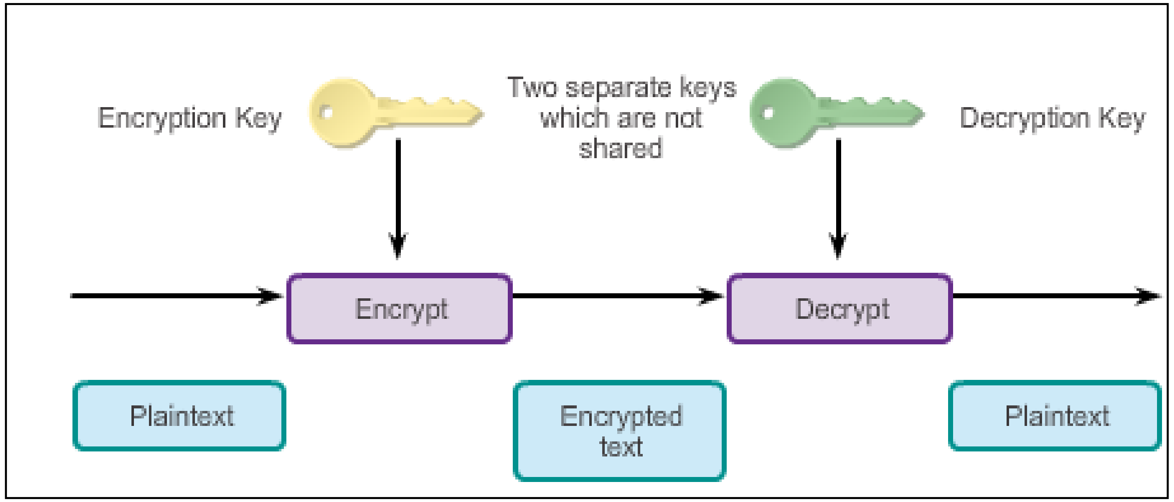
- It uses encryption techniques
  - Converts the original message to its ciphered variant
- To make the encryption work properly
  - Both the sender and the recipient must know the rules used to transform the original message into its coded form and back.
- The rules are based on algorithms and associated keys.
  - Decryption is extremely difficult (or impossible) without the right key



Encryption Algorithms

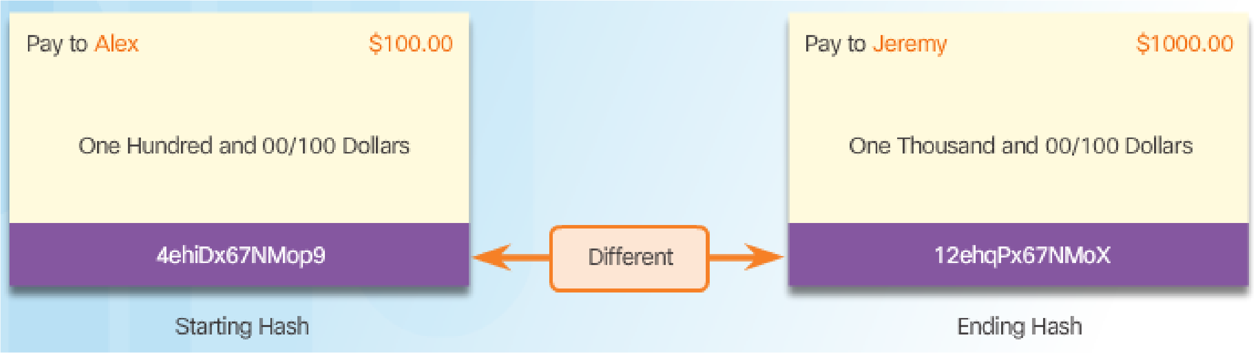
- Two main types:
  - Symmetric algorithms
    - **Same** key for encryption and decryption
    - DES, 3DES, AES, SEAL, RC ciphers
    - They differ in speed, key strength (56-256b)
    - Lower safety, high speed
  - Asymmetric algorithms
    - Another key for encryption, another for decryption
    - Uses RSA and PKI
      - Private and public key
    - Higher security, however, are slower if they want more resources
- Both use encryption keys
  - Balance between length (safer) and resource consumption and time
  - Problem: how do you exchange keys?
- Algorithm selection
  - Durability, speed, credibility, key strength





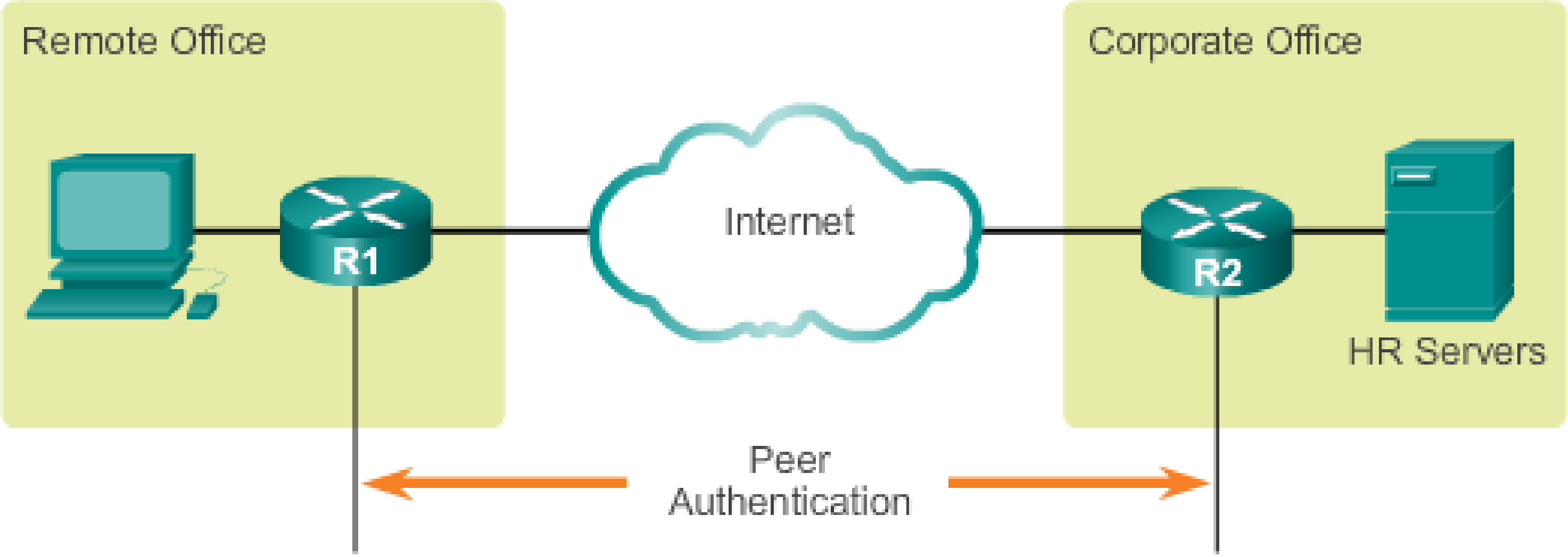
IPsec Data Integrity

- A means of getting the recipient to know that the message has not been manipulated with
  - Original sender
    - Generates a hash of the sent message
    - Which he sends with the message itself.
  - Recipient
    - Creates its own hash from the received message
    - Analyzes the message and the received hash
    - If they are the same, the recipient can be reasonably sure of the integrity of the original message.
  - Problem:
    - It is not possible to verify that the hash itself has not been manipulated
- Mechanisms => Hashing
  - MD5 (key 182-bit)
    - fast but breakable is no longer recommended
  - or SHA (160/256/512-bit)



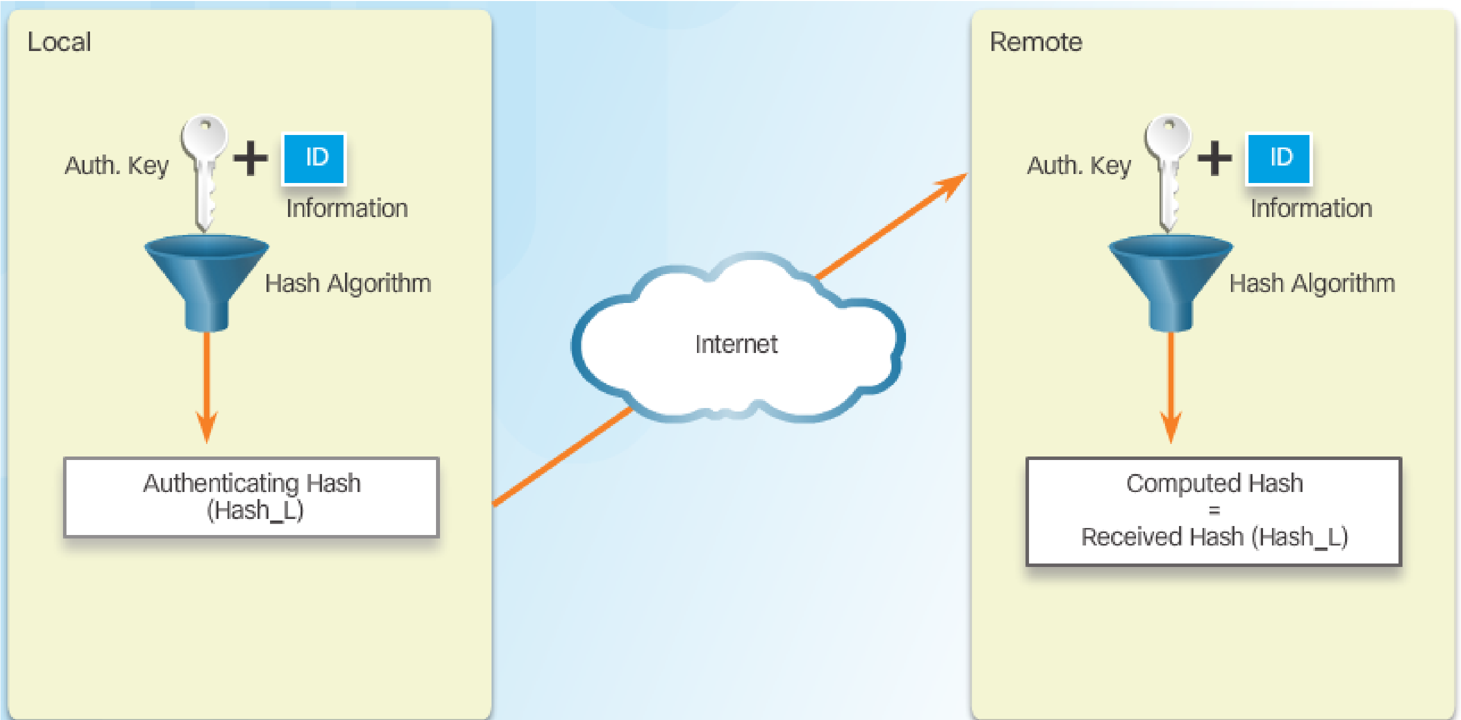
IPsec Authentication

- Is the other side the team I think it is?
  - Before the communication path can be considered secure, the device at the other end of the VPN tunnel must be authenticated
- IPsec supports two authentication methods
  - **Pre-shared key (PSK)**
  - **Signatures Rivest, Shamir and Adleman (RSA)**



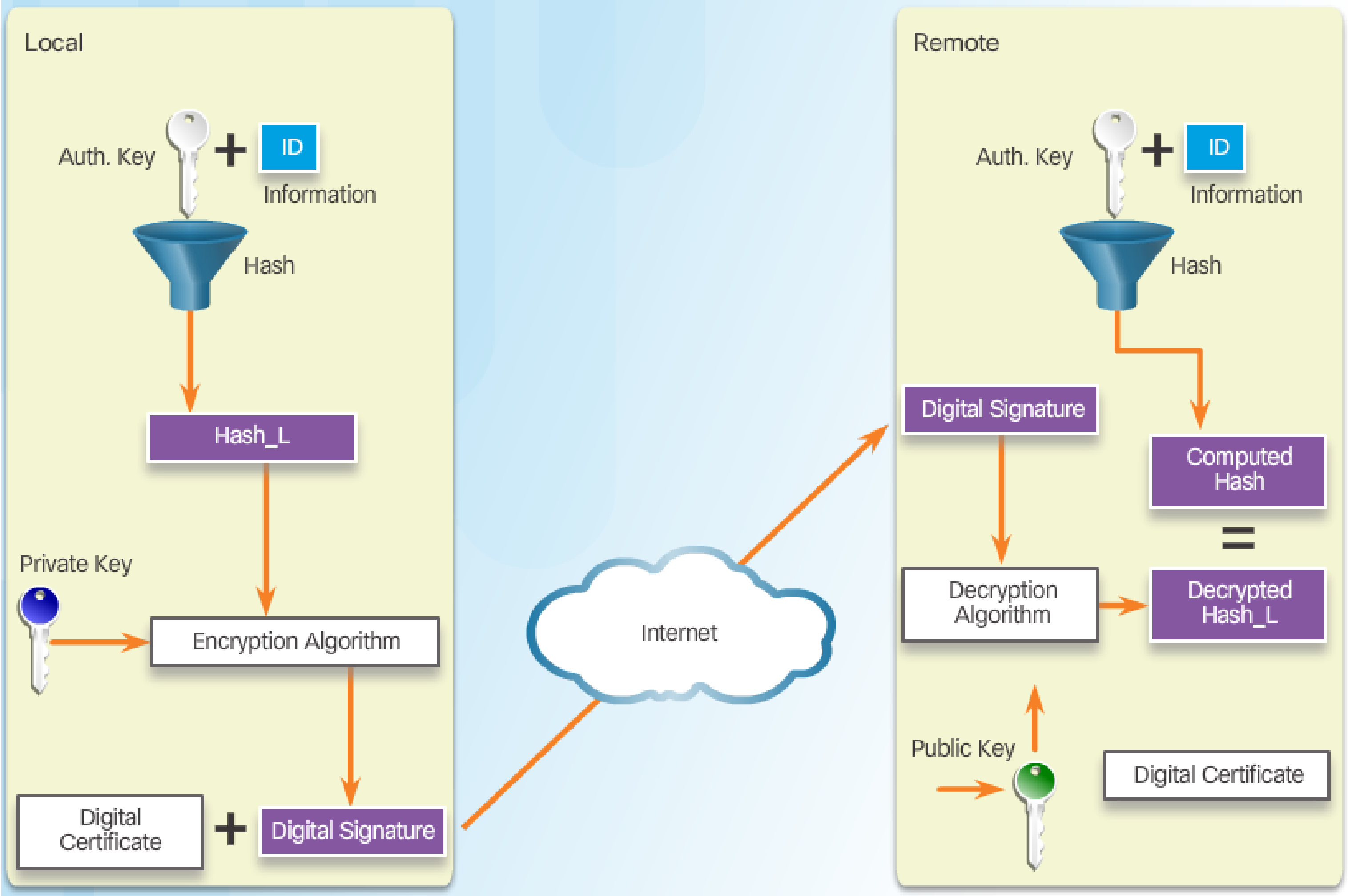
IPsec - PSK Authentication

- **Pre-shared key (PSK)**
  - The key is entered on each neighbor manually by the admin, easy manual configuration
  - Uses symmetric encryption => key transfer problem
  - The solution is not very scalable (key on every neighbor, many neighbors, many keys)



IPsec RSA authentication

- **Signatures Rivest, Shamir and Adleman(RSA)**
  - It uses a digital signature to transmit certificates
  - Digital certificates exchanged between neighbors are used to authenticate neighbors
  - It uses asymmetric algorithms for encryption



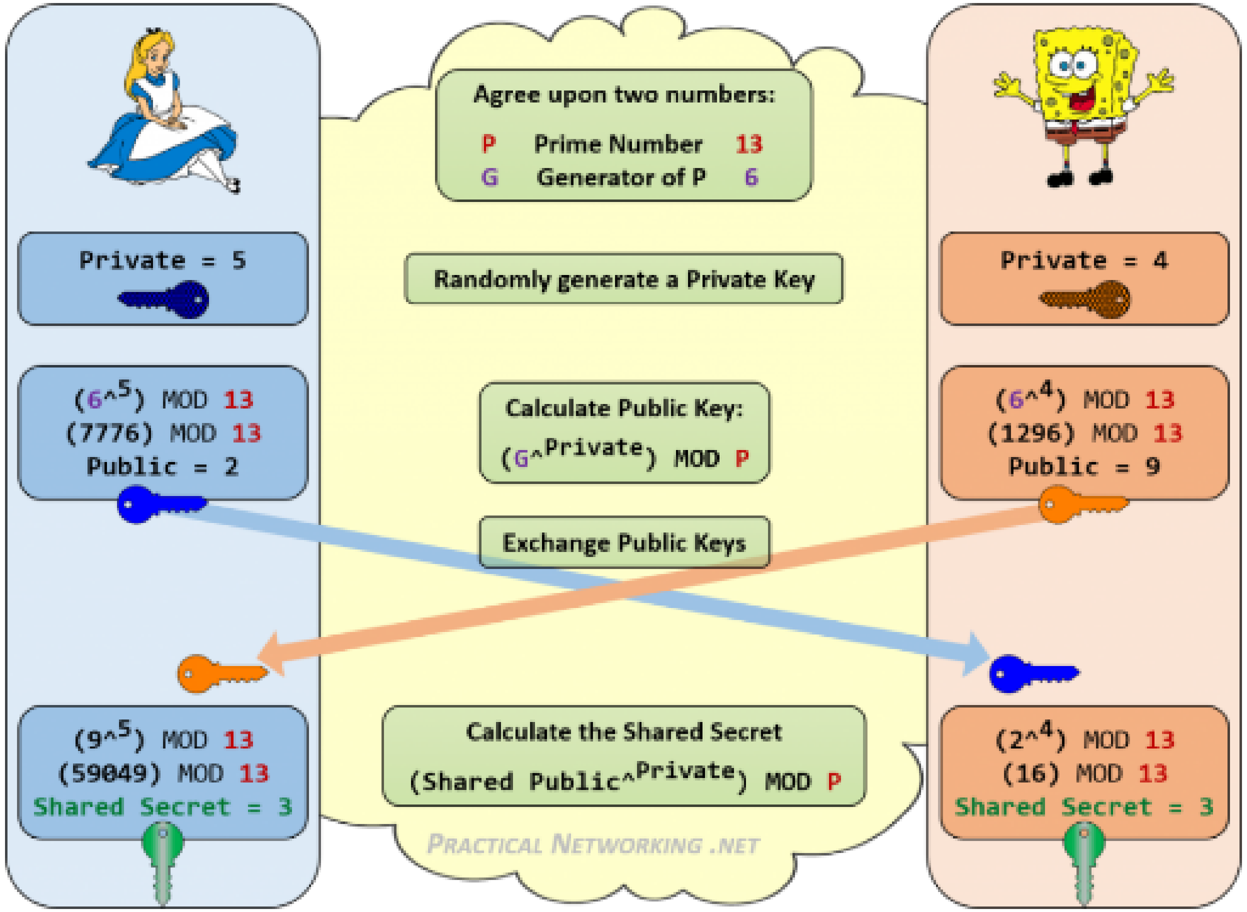
Key exchange via Diffie-Hellman

- Symmetric encryption algorithms (DES, 3DES and AES) as well as hashing algorithms (MD5 and SHA-1)
  - They require a symmetric shared secret key to perform encryption and decryption.
  - But how to transfer the key through an untrusted environment?
- Solves **Diffie Hellman** (DH) algorithm
  - DH is not an encryption algorithm
  - It is a method by which two parties can securely agree on encryption keys without the keys themselves being transmitted
    - The algorithm allows both neighbors to generate the same password without communicating at any time beforehand
  - There are multiple groups by key length = DH groups
  - It is part of IPsec for the build phase

Description	Diffie-Hellman Algorithm
Timeline	1976
Type of Algorithm	Asymmetric
Key Size (in bits)	512, 1024, 2048, 3072, 4096
Speed	Slow
Time to Crack (Assuming a computer could try 255 keys per second)	Unknown but considered safe using keys of 2048 or higher
Resource Consumption	Medium

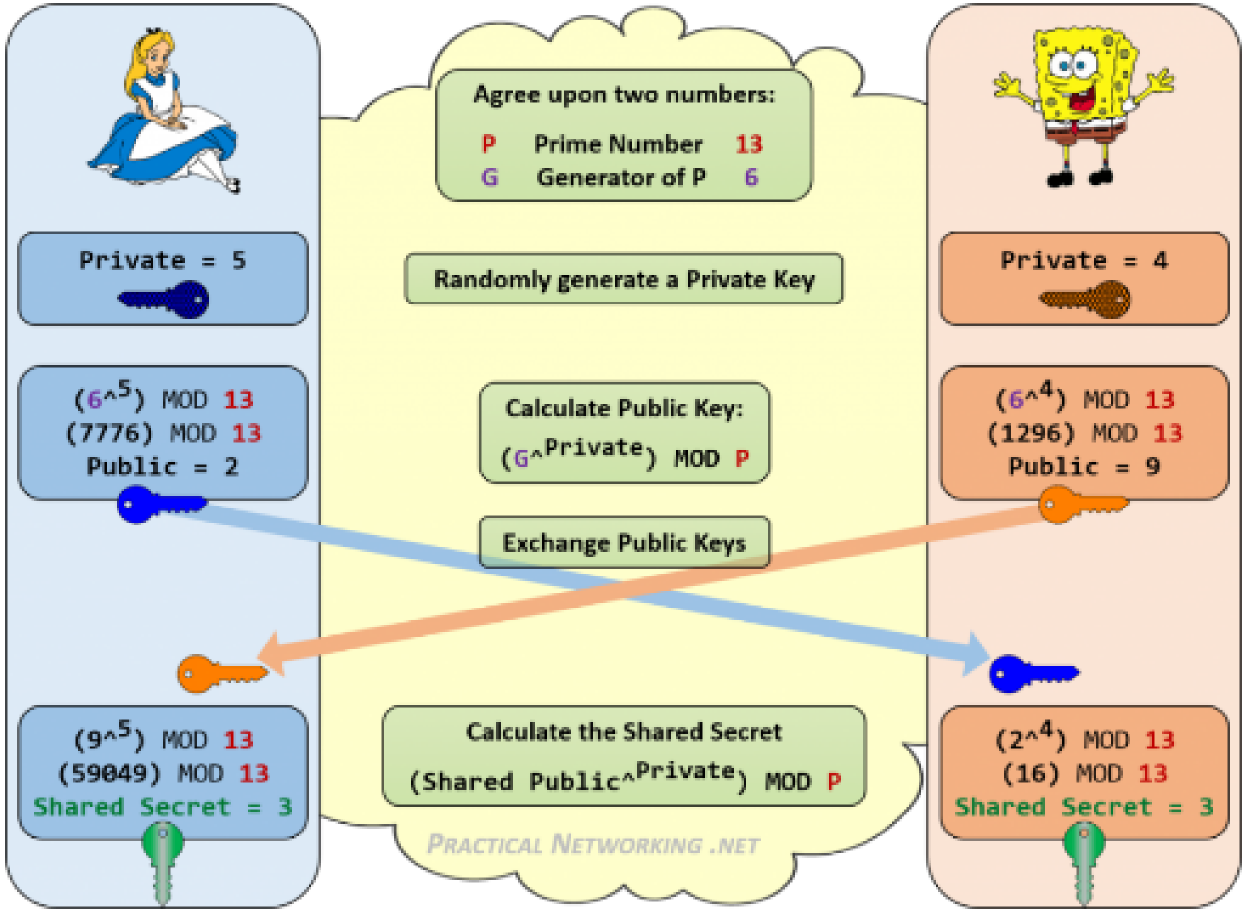
Diffie-Hellman Key Exchange

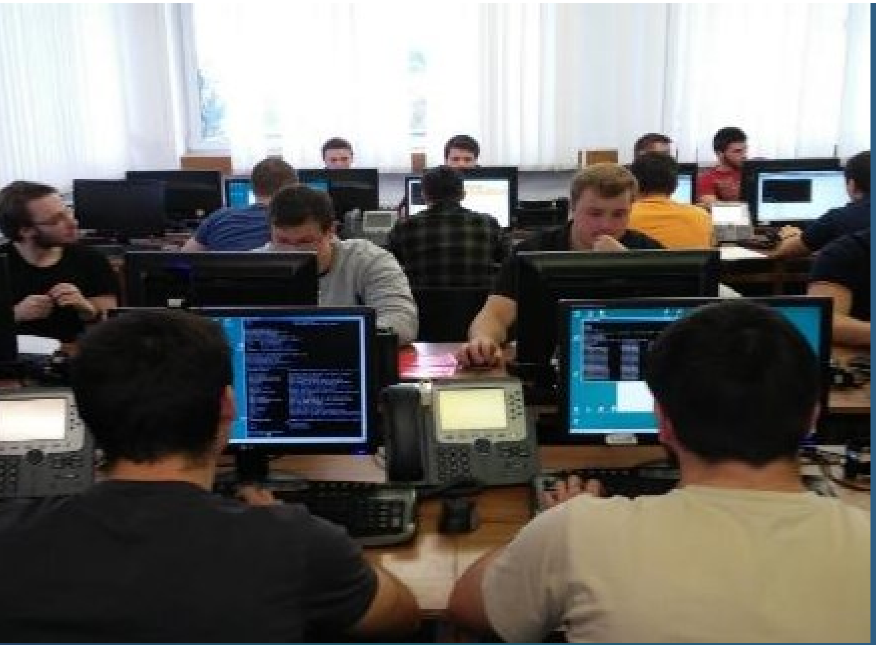
- Multi-step procedure
  - 1) The two sides must first agree on two numbers to share
    - Numbers do not have to be kept secret
    - P: prime number, usually large
    - G: base of the power, usually small
  - 2) Each of the parties will generate a random private number PRIVATE locally
  - 3) Each party, using G, P and private number, calculates its ***Public*** number (key)
    - $(G^{PRIVATE})$ MOD P = SHARED PUBLIC KEY
  - 4) The Parties shall exchange their *Public* Keys over an unencrypted network
  - 5) Each party calculates a secret key using G, P and the received public number
    - $(SHARED\_PUBLIC^{PRIVATE})$ MOD P = SECRET KEY
    - It is the same on both sides
    - Can be used for symmetric encryption



Diffie-Hellman groups

- In practice, there are numbered so-called DH groups
  - The group number determines how long is the DH key
  - DH groups 1, 2 and 5 should no longer be used
  - Dh groups
    - 14, key: 2048bitov
    - 15, key: 3072bitov
    - 14, key: 4096bitov



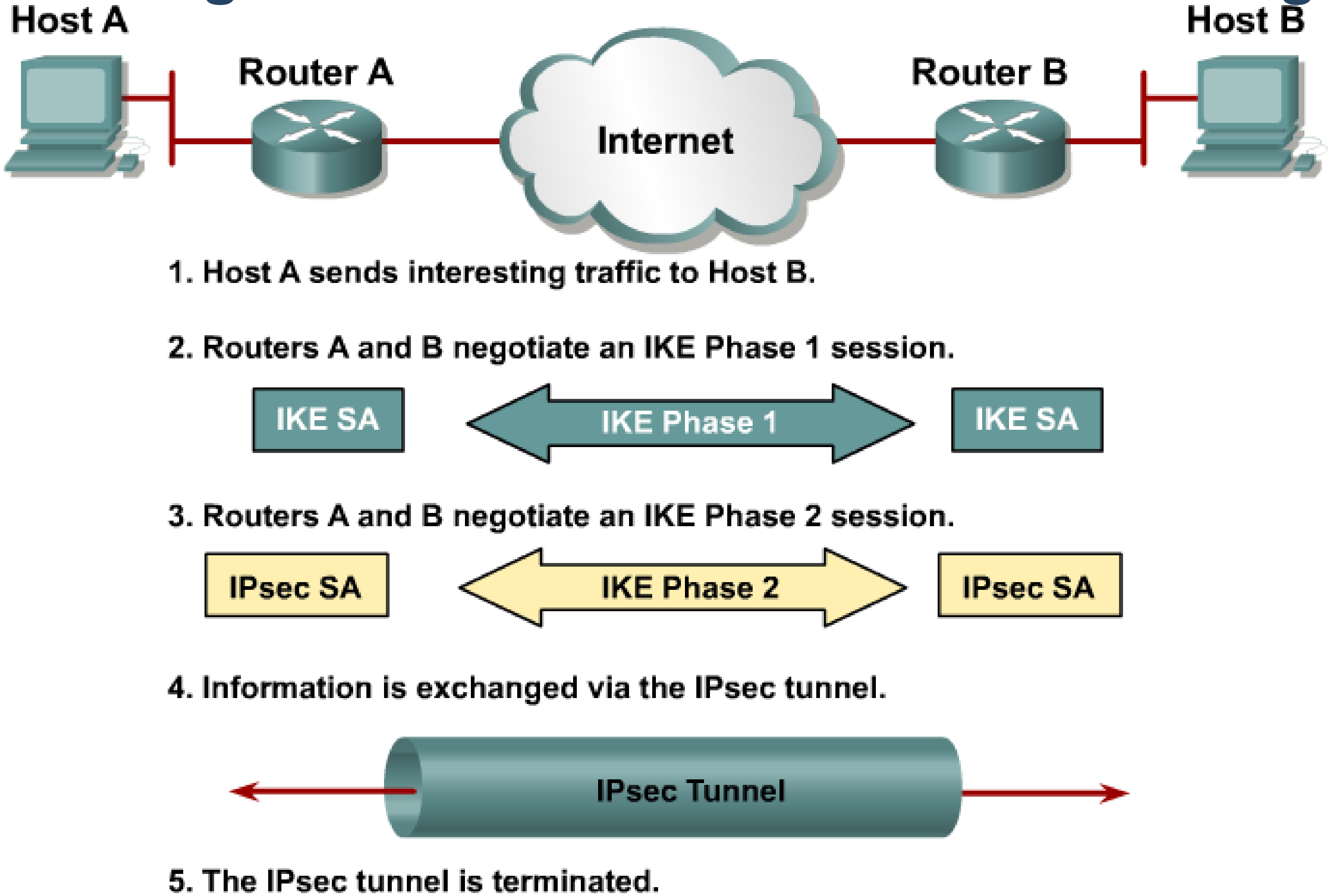


Simple IPsec configuration

Establish a connection between IPsec neighbors

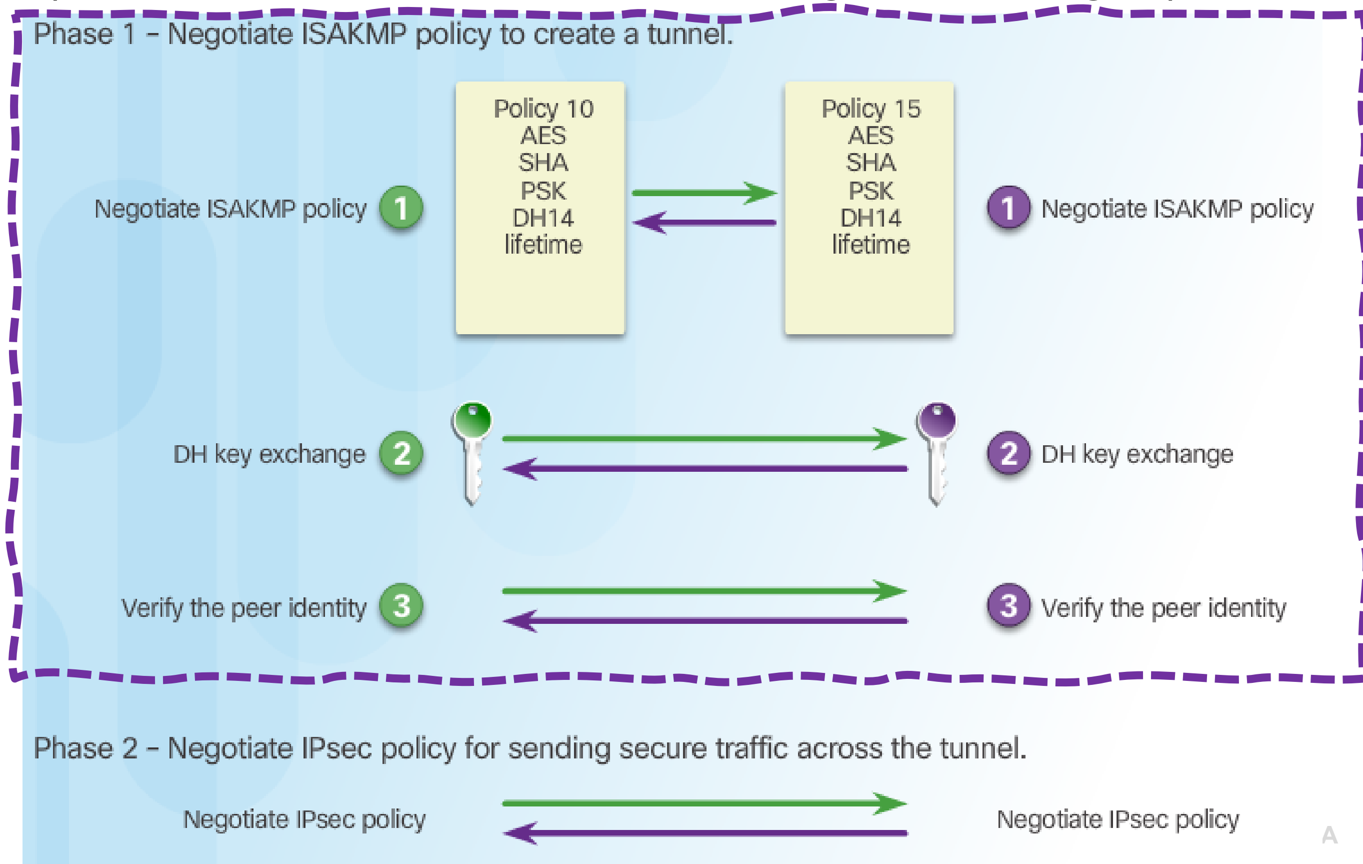
- It is necessary to realize
  - => We set up an encrypted IPsec VPN over an insecure Internet with a remote "unknown" gateway
- VPN gateways therefore need to resolve number of issues:
  - How do I know that a distant neighbor is the one to be and is not a stranger or a stranger?
    - Handled by authentication
  - How do we exchange encryption keys to encrypt data over the insecure internet?
    - We don't have a secure channel yet
  - What symmetric algorithm to use for data encryption?
    - What password to use for encryption / decryption
  - Which traffic will be encrypted and which will not?
  - What IPsec protocol and mode of operation will we use for VPN ?
  - And many others....

Establishing of a connection between IPsec neighbors



Connection creation: IKE phase 1 (IKE SA)

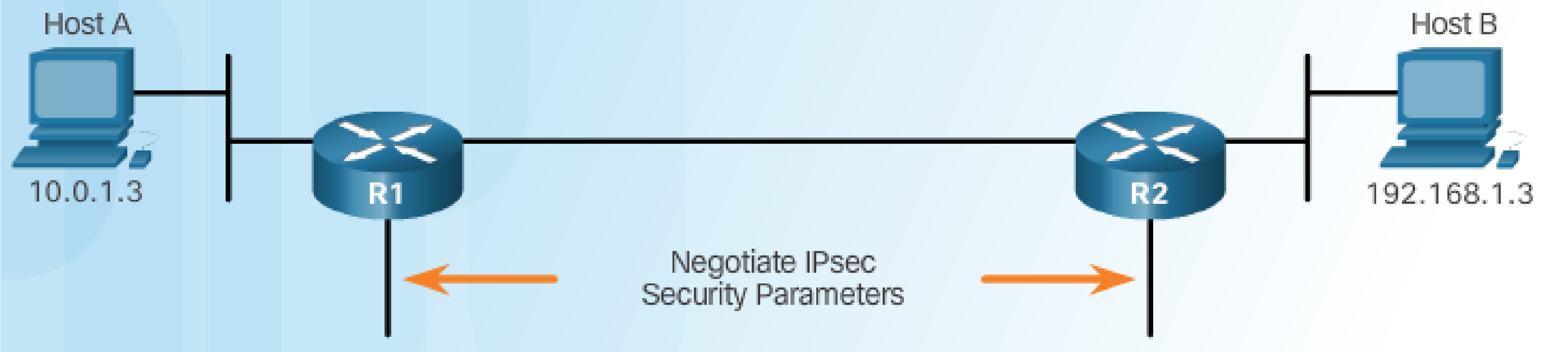
- IKE phase 1
  - Authenticates neighbors and manage ISAMP policies (ISAKMP Security Associations)
  - Creates a secure channel for IKE Phase 2 (It does not negotiates the characteristics of the IPsec tunnel itself|
  - Has two modes, **Main and aggressive** (uses different number of exchanged messages)
- IKE phase 1 has three steps:
  - ISAKMP policy agreement
  - Cipher / hash key exchange using Diffie-Hellman algorithm
  - Verification of neighbors' identity
- What are ISAKMP policies?
  - What encryption algorithm? (confident.)
  - What hashing algorithm? (integr.)
  - What Diffie-Hellman group?
  - What way to verify your identity?(auth.)
  - Lifetime
- Identity verification
  - According to the method agreed in the first step



Establishing a connection between IPsec neighbors

- The creation of an IPsec tunnel is not done in advance
- Tunneling always triggers the arrival of a first packet (so called packet of interest) transmitted from the one network to another
- Upon arrival of such a packet (identified by the ACL)
  - Both phases will take place and connection security associations will be established
  - Their use is tied to the lifetime determined by the configuration
  - IPsec is terminated after a period of inactivity and lifetime
    - And created till after

Connection creation: IKE phase 2

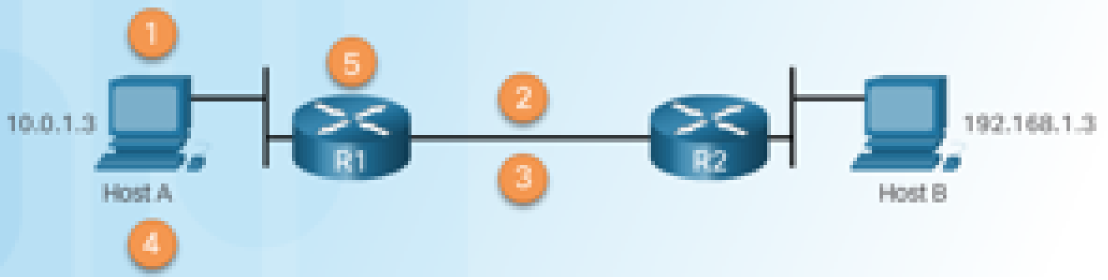


- IKE Phase 2 is responsible for arranging how IPsec will select packets for encryption and how packets is going to be encrypted between neighbors
  - What IPsec protocol - AH, ESP, AH + ESP?
  - Which mode - tunnel or transport?
  - What encryption algorithm?
  - What hashing mechanism?

  transformation set

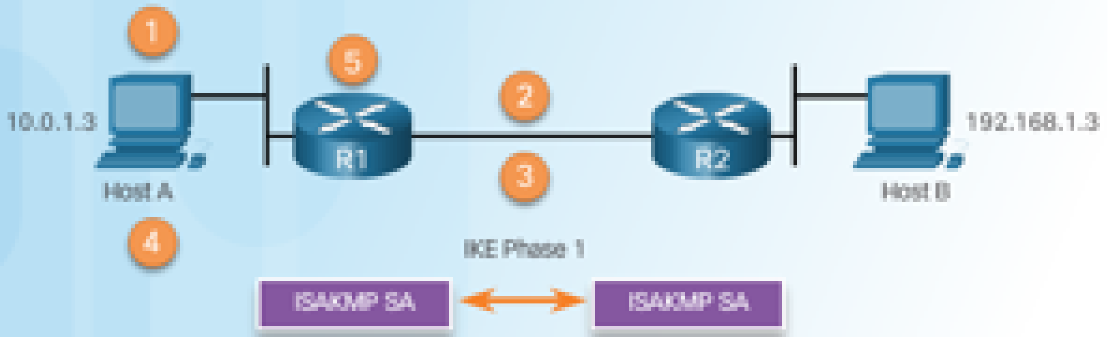
  - What encryption keys ? (DH)
  - What will be the lifetime of the agreed information?

IPsec Negotiation

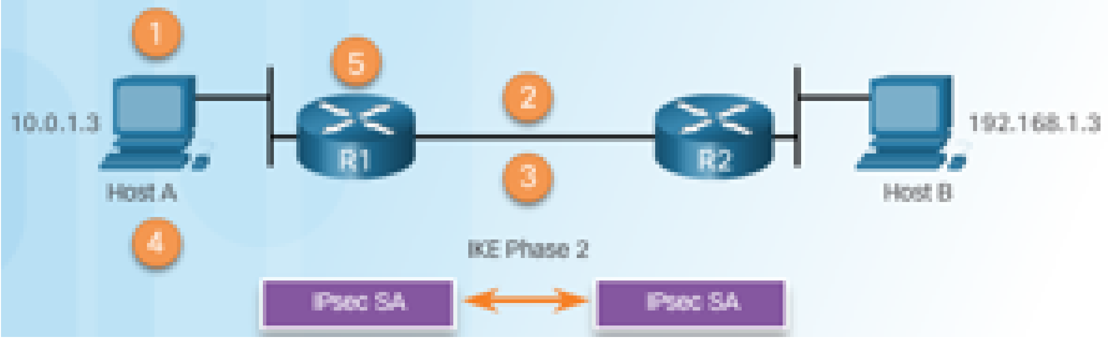


IPsec VPN Negotiation: Step 1 - Host A sends interesting traffic to Host B.

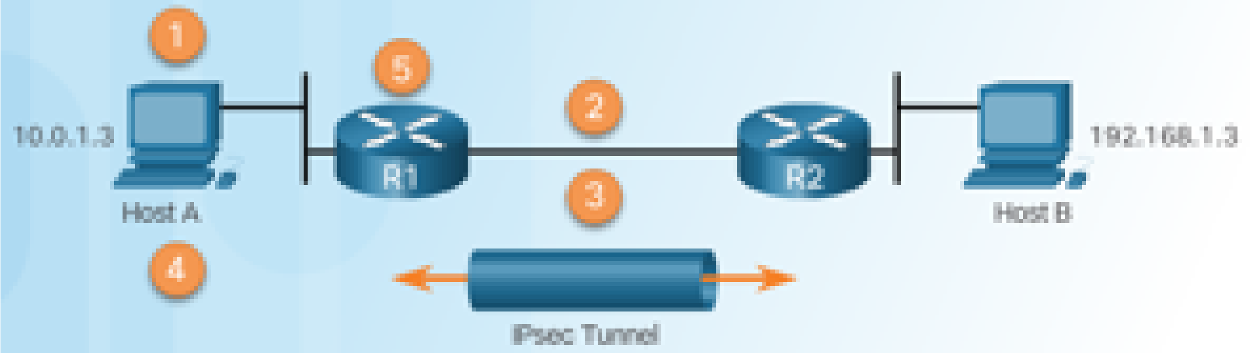
IPsec VPN Negotiation: Step 2 - R1 and R2 negotiate an IKE Phase 1 session.



IPsec VPN Negotiation: Step 3 - R1 and R2 negotiate an IKE Phase 2 session.

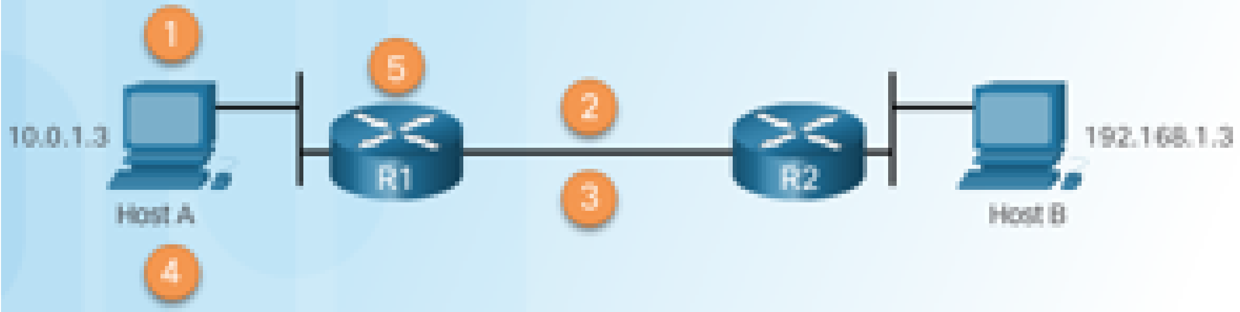


IPsec Negotiation



IPsec VPN Negotiation: Step 4 - Information is exchanged via IPsec tunnel.

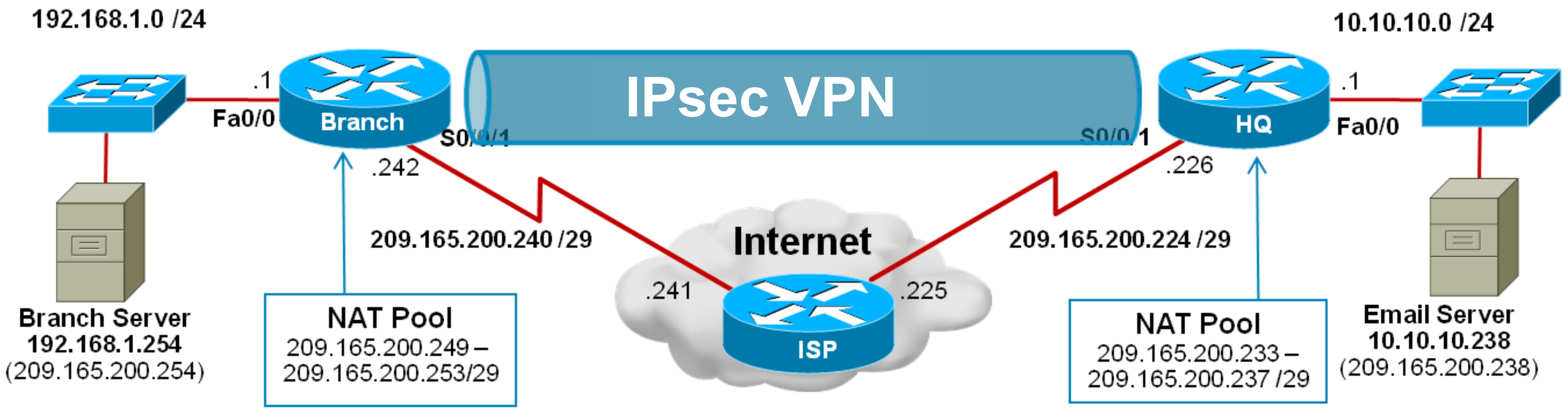
IPsec VPN Negotiation: Step 5 - The IPsec tunnel is terminated.



Site-to-Site IPsec VPN configuration (with IKEv1)

IPsec configuration steps

- IPsec configuration procedure
  - Establish at least one ISAKMP policy for phase 1
  - Create at least one transformation set for phase 2
  - Create an ACL that specifies what to provide using IPsec
    - When the arrival of the packet starts IPSec processing
  - Create an encryption map that specifies what the ACL should provide with the ACL and how
    - Till the arrival of the packet starts IPSec processing
  - Apply an encryption map to the output interface
- Note:
  - In the example, the Internet is used only as a backup connection for a private WAN



Complete IPsec VPN Branch Router configuration

```
Branch# conf t
Branch(config)# crypto isakmp policy 1                ❶
Branch(config-isakmp)# encryption aes 256
Branch(config-isakmp)# hash sha
Branch(config-isakmp)# lifetime 3600
Branch(config-isakmp)# authentication pre-share
Branch(config-isakmp)# group 24
Branch(config-isakmp)# exit
Branch(config)# crypto isakmp key cisco123 address 209.165.200.226
! Specifikuj IPSec transformacnu sadu
Branch(config)# crypto ipsec transform-set MOJA_TR_SADA esp-aes esp-sha-hmac esp-3des   ❷
Branch(cfg-crypto-trans)# exit
! Specifikuj prevadzku, ktora bude sifrovana
Branch(config)# access-list 110 permit ip 192.168.1.0 0.0.0.255 10.10.10.0 0.0.0.255
Branch(config)#
Branch(config)#
Branch(config)# crypto map MOJA_MAPA 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
! Mapa spoji kroky dokopy, t.j. na akeho suseda aku Tr.Sadu + ake ACL
! Moze mat viac blokov pre viac susedov
Branch(config-crypto-map)# set transform-set MOJA_TR_SADA
Branch(config-crypto-map)# set peer 209.165.200.226
Branch(config-crypto-map)# match address 110
Branch(config-crypto-map)# exit
Branch(config)# int s0/0/1
Branch(config-if)# crypto map MOJA_MAPA
Branch(config-if)# ^Z
Branch#
```

❶ **ISAKMP Policy**
Specifies the initial VPN security details

❷ **IPsec trans. set**
Specifies how the IPsec packet will be encapsulated

❸ **Crypto ACL**
Specifies the traffic that will trigger the VPN to activate

❹ **VPN Tunnel Information**
Creates the crypto map that combines the ISAKMP policy, IPsec transform set, VPN peer address, and crypto ACL

❺ **Apply the Crypto Map**
Identifies which interface is actively looking to create a VPN

Default ISAKMP Policy

- ISAKMP policy
  - Zariadenie má default politiku s predefinovanými vlastnosťami
  - Konfiguráciou vytvárame vlastné
    - Môžeme použiť rôzne per suseda

```
R1# show crypto isakmp default policy

Default IKE policy
Default protection suite of priority 65507
          encryption algorithm:   AES - Advanced Encryption Standard (128 bit keys).
          hash algorithm:         Secure Hash Standard
          authentication method:  Rivest-Shamir-Adleman Signature
          Diffie-Hellman group:   #5 (1536 bit)
          lifetime:               86400 seconds, no volume limit
Default protection suite of priority 65508
          encryption algorithm:   AES - Advanced Encryption Standard (128 bit keys).
          hash algorithm:         Secure Hash Standard
          authentication method:  Pre-Shared Key
          Diffie-Hellman group:   #5 (1536 bit)
```

IPsec Transform Set

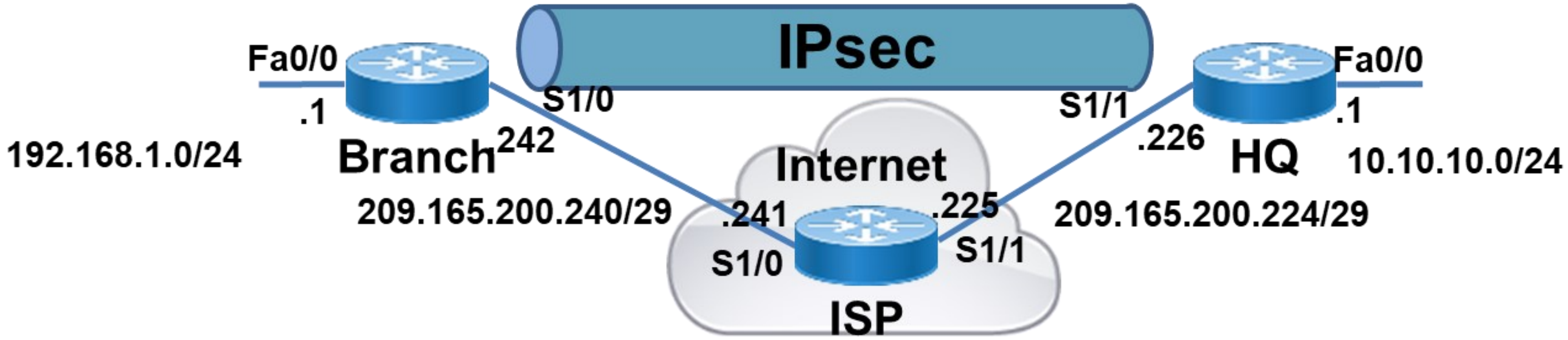
- Konfig množiny šifrovacích a hašovacích algoritmov
  - Môže obsahovať viac kombinácii
    - pre lepšiu zhodu so susedom a vyššiu úroveň bezpečnosti


```
R1(config)# crypto ipsec transform-set ?
  WORD  Transform set tag

R1(config)# crypto ipsec transform-set R1-R2 ?
  ah-md5-hmac      AH-HMAC-MD5 transform
  ah-sha-hmac      AH-HMAC-SHA transform
  ah-sha256-hmac   AH-HMAC-SHA256 transform
  ah-sha384-hmac   AH-HMAC-SHA384 transform
  ah-sha512-hmac   AH-HMAC-SHA512 transform
  comp-lzs         IP Compression using the LZS compression algorithm
  esp-3des         ESP transform using 3DES(EDE) cipher (168 bits)
  esp-aes          ESP transform using AES cipher
  esp-des          ESP transform using DES cipher (56 bits)
  esp-gcm          ESP transform using GCM cipher
  esp-gmac         ESP transform using GMAC cipher
  esp-md5-hmac     ESP transform using HMAC-MD5 auth
  esp-null         ESP transform w/o cipher
  esp-seal         ESP transform using SEAL cipher (160 bits)
  esp-sha-hmac     ESP transform using HMAC-SHA auth
  esp-sha256-hmac  ESP transform using HMAC-SHA256 auth
  esp-sha384-hmac  ESP transform using HMAC-SHA384 auth
  esp-sha512-hmac  ESP transform using HMAC-SHA512 auth
```

IPsec configuration example (ISR routers)

```
! BRANCH
ena
conf t
crypto isakmp policy 1
          encryption aes 256
          hash sha
          authentication pre-share
          group 24
          exit
crypto isakmp key cisco123 address 209.165.200.226
crypto ipsec transform-set MOJA_TR_SADA esp-aes esp-sha256-
hmac
access-list 110 permit ip 192.168.1.0 0.0.0.255 10.10.10.0
0.0.0.255
crypto map MOJA_MAPA 10 ipsec-isakmp
          set transform-set MOJA_TR_SADA
          set peer 209.165.200.226
          match address 110
          exit
int s 1/0
          crypto map MOJA_MAPA
          end
wr mem
```

```
ena
conf t
crypto isakmp policy 1
          encryption aes 256
          hash sha
          authentication pre-share
          group 24
          exit
crypto isakmp key cisco123 address 209.165.200.242
crypto ipsec transform-set MOJA_TR_SADA esp-aes esp-
sha256-hmac
access-list 110 permit ip 10.10.10.0 0.0.0.255
192.168.1.0 0.0.0.255
crypto map MOJA_MAPA 10 ipsec-isakmp
          set transform-set MOJA_TR_SADA
          set peer 209.165.200.242
          match address 110
          exit
int s 1/1
          crypto map MOJA_MAPA
          end
wr mem
```



Verification

- Displays configured ISAKMP policies
  - **`Show crypto isakmp policy`**
- Display PSK key
  - **`sh crypto isakmp key`**
- Display IKE phase 1 SA
  - It can be seen only when phase `1` is over
  - **`Sh crypto isakmp sa`**
- View config and status of Sapre Ipsec
  - **`Sh crypto ipsec sa`**
- Show crypto map
  - **`Show crypto map`**

  - **`Show crypto session`**

Verification

```
Branch# sh crypto isakmp policy

Global IKE policy
Protection suite of priority 1
        encryption algorithm:   AES - Advanced Encryption Standard (256 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #24 (2048 bit, 256 bit subgroup)
        lifetime:               86400 seconds, no volume limit
```

```
Branch# sh crypto isakmp key
Keyring      Hostname/Address                                            Preshared Key

default      209.165.200.226                                             cisco123
```

```
Branch#sh crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
209.165.200.226 209.165.200.242 QM_IDLE           1001 ACTIVE

IPv6 Crypto ISAKMP SA
```

Verification

```
Branch#sh crypto isakmp sa detail
Codes: C - IKE configuration mode, D - Dead Peer Detection
       K - Keepalives, N - NAT-traversal
       T - cTCP encapsulation, X - IKE Extended Authentication
       psk - Preshared key, rsig - RSA signature
       renc - RSA encryption
IPv4 Crypto ISAKMP SA

C-id  Local           Remote          I-VRF  Status Encr Hash   Auth DH Lifetime Cap.

1001  209.165.200.242 209.165.200.226        ACTIVE aes  sha    psk  24 23:54:29
       Engine-id:Conn-id =  SW:1

IPv6 Crypto ISAKMP SA
```

```
Branch#sh crypto session
Crypto session current status

Interface: Serial1/0
Session status: UP-ACTIVE
Peer: 209.165.200.226 port 500
  Session ID: 0
  IKEv1 SA: local 209.165.200.242/500 remote 209.165.200.226/500 Active
  IPSEC FLOW: permit ip 192.168.1.0/255.255.255.0 10.10.10.0/255.255.255.0
        Active SAs: 2, origin: crypto map
```

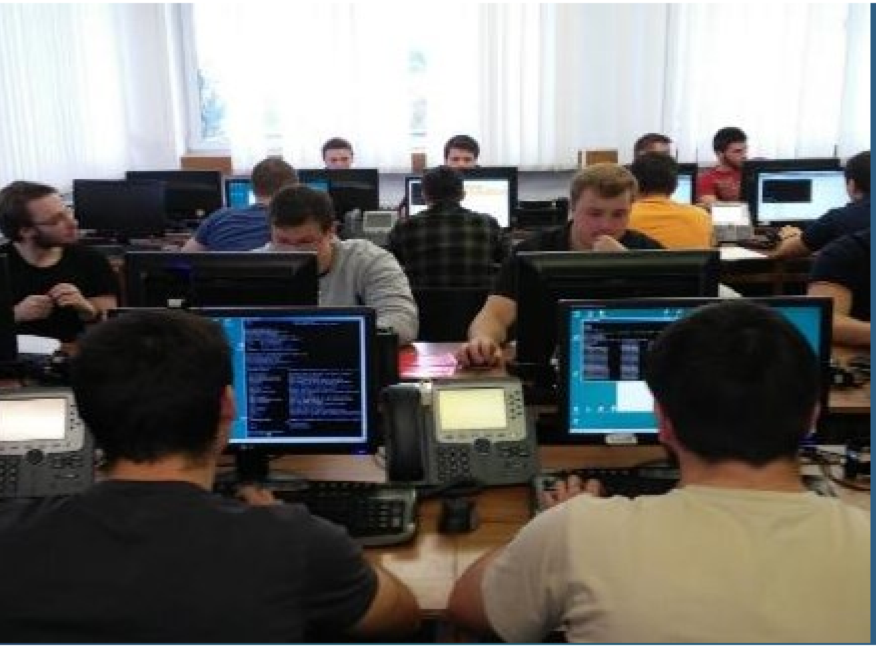
Verification

```
Branch#sh crypto ipsec sa

interface: Serial1/0
    Crypto map tag: MY_MAP, local addr 209.165.200.242

   protected vrf: (none)
   local  ident (addr/mask/prot/port):
(192.168.1.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port):
(10.10.10.0/255.255.255.0/0/0)
   current_peer 209.165.200.226 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 14, #pkts encrypt: 14, #pkts digest: 14
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify: 10
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 209.165.200.242, remote crypto endpt.:
209.165.200.226
     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb
Serial1/0
     current outbound spi: 0x3486AE69(881241705)
     PFS (Y/N): N, DH group: none
```

```
 inbound esp sas:
      spi: 0x96FAE6C7(2533025479)
        transform: esp-aes esp-sha256-hmac ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, sibling_flags 80004040, crypto
map: MY_MAP
        sa timing: remaining key lifetime (k/sec): (4270878/3185)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x3486AE69(881241705)
        transform: esp-aes esp-sha256-hmac ,
        in use settings ={Tunnel, }
        conn id: 2, flow_id: SW:2, sibling_flags 80004040, crypto
map: MY_MAP
        sa timing: remaining key lifetime (k/sec): (4270877/3185)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:
```

IPsec: Final notes

- If we have ACLs on the devices, ports must be open for Ipsec
  - ESP: UDP/50
  - ISAKMP: UDP/500



Simple GRE over IPsec configuration

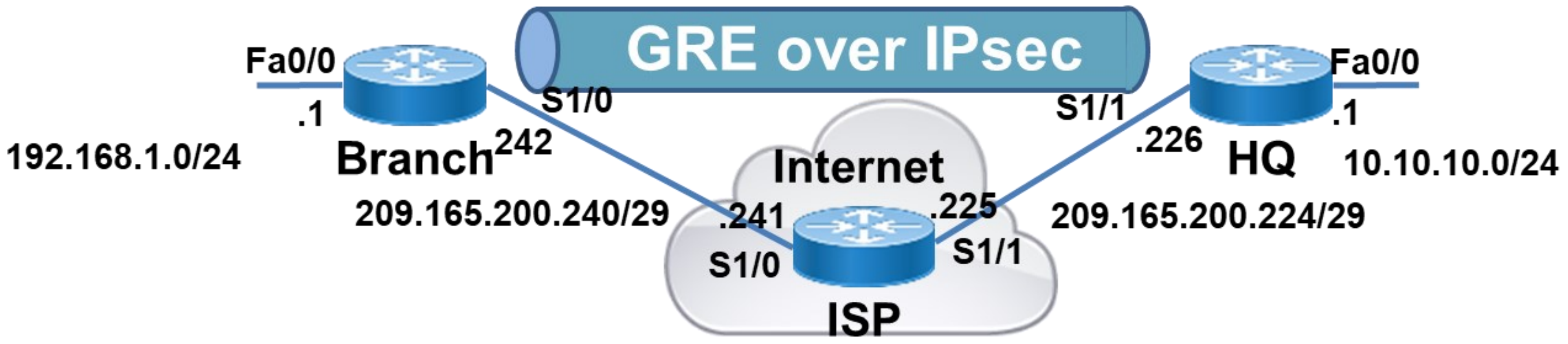
Simple GRE over IPsec configuration

- Used if dynamic routing is required to operate exchange network prefixes between sites
  - Otherwise, static routing is required
- Requires
  - Configure GRE
  - Configure IPsec
    - ... however, it need keep in mind that plain packets of the transmitted protocol no longer leave the output interface, but GRE packets
    - Command **set peer** in the cryptomap must match the address given
      In command **tunnel destination** on the Tunnel interface
    - ACLs in the cryptomap must select **GRE** packets whose source matches the **tunnel source** command and the destination of the **tunnel destination** command

Router config - GRE

```
! Branch
ena
conf t
int tunnel 0
        tunnel source s 1/0
        tunnel destination 209.165.200.226
        tunnel mode gre ip
        ip add 172.16.1.1 255.255.255.0
router ospf 1
        network 192.168.1.0 0.0.0.255 area 0
        network 172.16.1.0 0.0.0.255 area 0
```

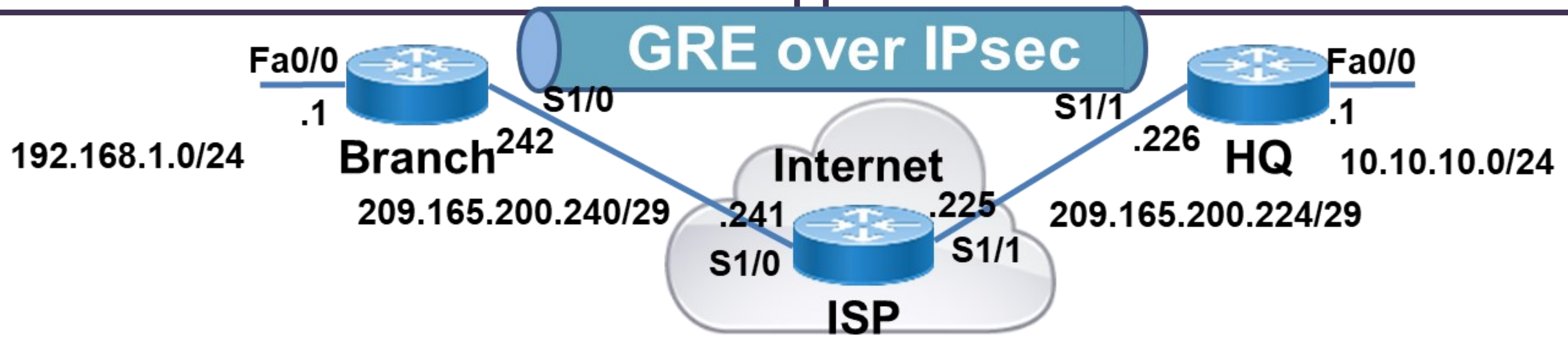
```
!HQ
ena
conf t
int tunnel 0
        tunnel source s 1/1
        tunnel destination 209.165.200.242
        tunnel mode gre ip
        ip add 172.16.1.2 255.255.255.0
router ospf 1
        network 10.10.10.0 0.0.0.255 area 0
        network 172.16.1.0 0.0.0.255 area 0
```



Router config - preparation

```
! Branch
ena
conf t
crypto isakmp policy 1
         encryption aes 256
         hash sha
         authentication pre-share
         group 24
         exit
crypto isakmp key cisco123 address 209.165.200.226
crypto ipsec transform-set MY_TR_SET esp-aes esp-
sha256-hmac
access-list 110 permit gre host 209.165.200.242 host
209.165.200.226
crypto map MY_MAP 10 ipsec-isakmp
         set transform-set MY_TR_SET
         set peer 209.165.200.226
         match address 110
         exit
int s 1/0
         crypto map MY_MAP
         end
wr mem
```

```
ena
conf t
crypto isakmp policy 1
         encryption aes 256
         hash sha
         authentication pre-share
         group 24
         exit
crypto isakmp key cisco123 address 209.165.200.242
crypto ipsec transform-set MY_TR_SET esp-aes esp-
sha256-hmac
access-list 110 permit gre host 209.165.200.226 host
209.165.200.242
crypto map MY_MAP 10 ipsec-isakmp
         set transform-set MY_TR_SET
         set peer 209.165.200.242
         match address 110
         exit
int s 1/1
         crypto map MY_MAP
         end
wr mem
```



IPsec in PT

- Implementation of IPSec in P.T 7.3 using 1841 / 19xx requires activation of the Security feature in IOS as follows:
  - Enter the show version and if is the security feature **disable**

```
-----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      disable       None           None
data          disable       None           None
```

  - Enter the command in config mode

```
Router(config)#license boot module c1900 technology-package securityk9
```

  - And then

```
ACCEPT? [yes/no]: yes
Router(config)# End
Router# Copy run startup
Router# reload
```

  - After restart, verify that Security is enabled

Configuration of Site-to-Site IPsec PSK VPN with IKEv2

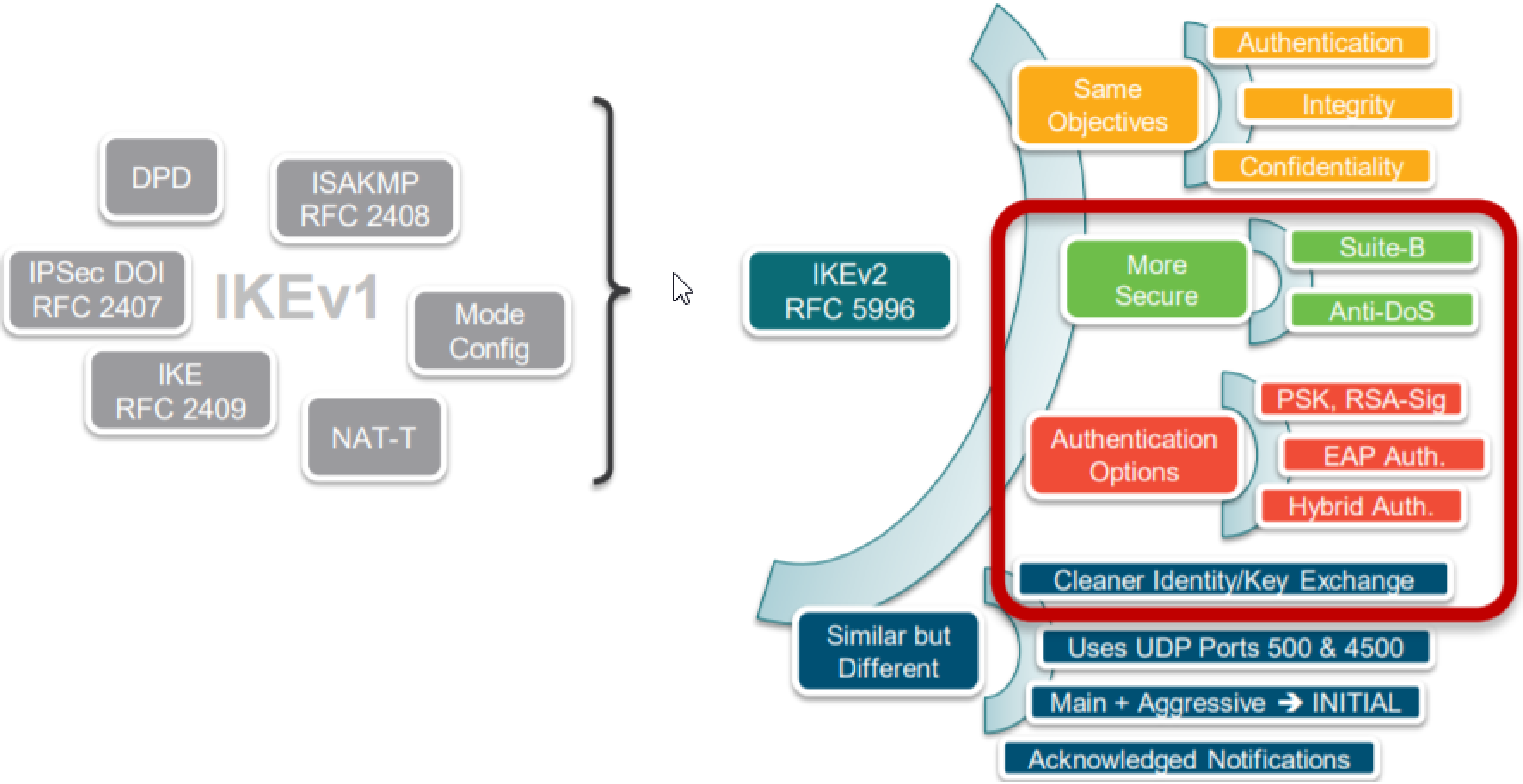
New

* CSR1000v

IKEv2 benefits

- Dead Peer Detection and Network Address Translation-Traversal
  - Internet Key Exchange Version 2 (IKEv2) provides built-in support for Dead Peer Detection (DPD) and Network Address Translation-Traversal (NAT-T)
- Denial of Service Attack Resilience
- EAP Support
  - IKEv2 allows the use of Extensible Authentication Protocol (EAP) for authentication
  - And hybrid methods
- Multiple Crypto Engines
- Certificate URLs
  - Certificates can be referenced through a URL and hash, instead of being sent within IKEv2 packets, to avoid fragmentation.
- Reliability and State Management (Windowing)
  - IKEv2 uses sequence numbers and acknowledgments to provide reliability, and mandates some error-processing logistics and shared state management.
- Several possibilities to identify neighbors
  - IP address, DNS name, URL
- Several possibilities to apply IPsec
  - Using crypto map
  - Gre tunnel protection
  - ...

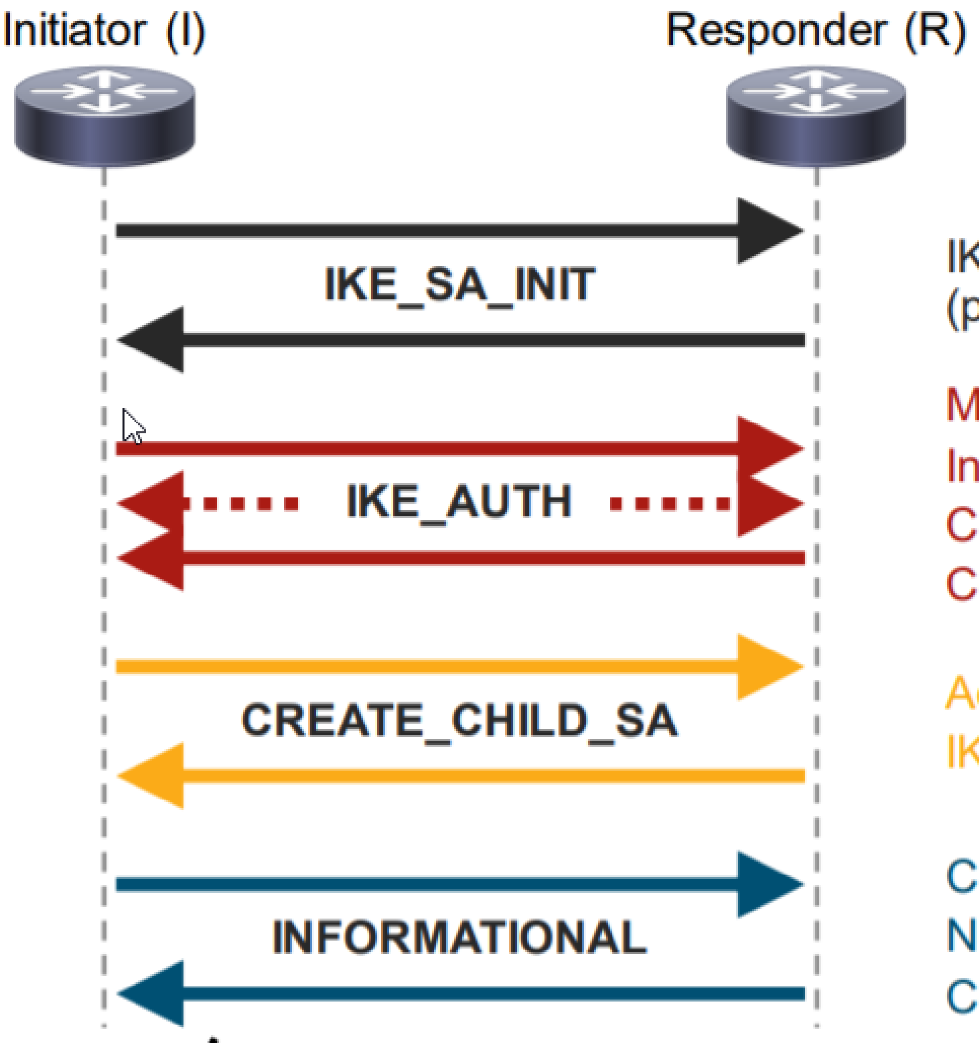
Comparing IKEv1 & IKEv2



IKEv1 vs IKEv2

	IKEv1	IKEv2
Auth messages	6 max	Open ended
First IPsec SA	6-9 messages	~ 4-6 messages minimum
Authentication	pubkey-sig, [pubkey-encr], PSK	pubkey-sig, PSK, EAP, asymmetrical authentication
Security	Vulnerable to DOS attacks	Anti-clogging, Suite-B Support, ...
IKE rekey	Requires re-auth (expensive)	No Re-auth
Notifies	Fire & Forget	Acknowledged
Message Segmentation	None, relies on IP fragmentation	Protocol built in
NG Cryptography	**Support is stopped***	**Yes**

IKEv2 exchanges



IKEv2 Security Association (SA) establishment (proposal selection, key exchange)

Mutual authentication & identity exchange
Initial IPSec SAs establishment
Certificate exchange (optional)
Configuration exchange (optional)

Additional IPSec SAs establishment
IKEv2 & IPSec SA rekey

Can be (I → R) with ACK or (R → I) with ACK
Notifications (SA deletion, liveness check, ...)
Configuration exchange (one or both ways)

Internet Key Exchange Version 2 - CLI Constructs

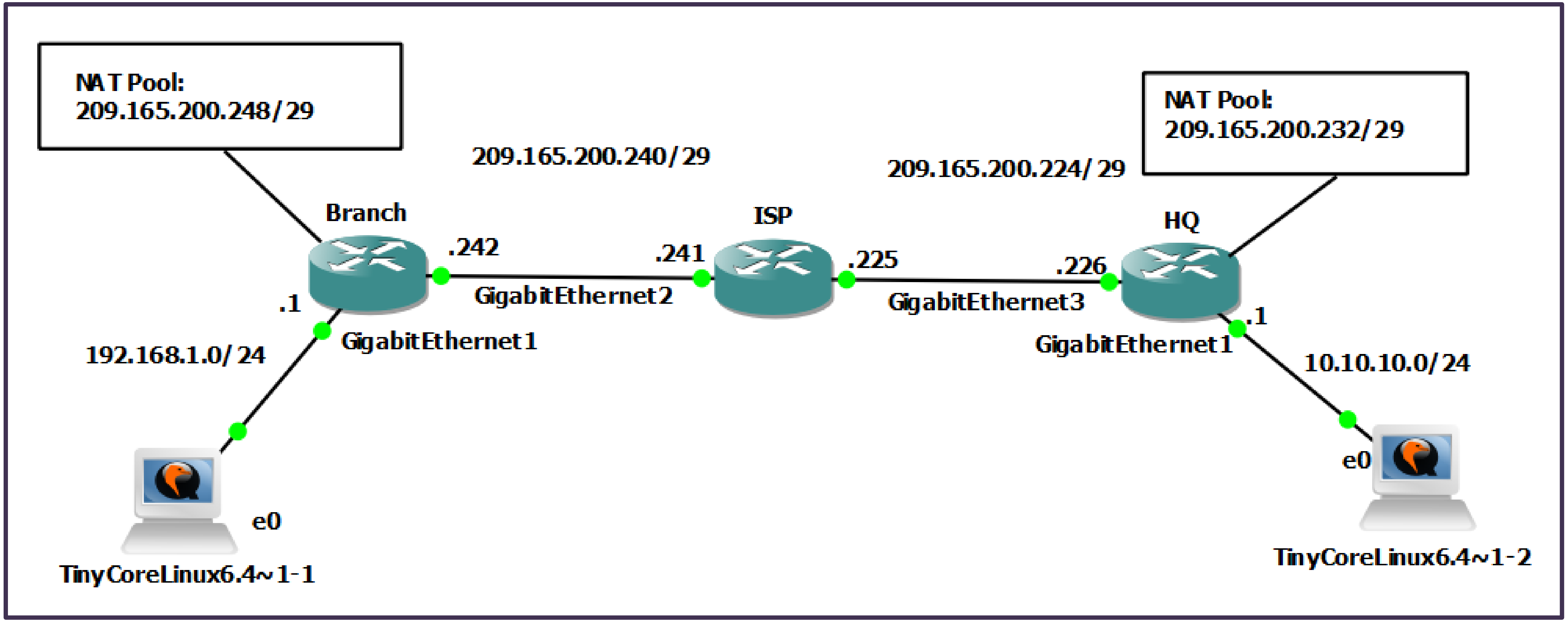
- **IKEv2 Key Ring**
  - An IKEv2 keyring is a repository of symmetric and asymmetric preshared keys and is independent of the IKEv1 key ring.
  - The IKEv2 keyring is associated with an IKEv2 profile and hence supports a set of peers that match the IKEv2 profile.
  - The IKEv2 key ring gets its VPN routing and forwarding (VRF) context from the associated IKEv2 profile.
  - Defined per a peer neighbor
- **IKEv2 Proposal**
  - A collection of transforms used in the negotiation of Internet Key Exchange (IKE) **security associations** (SAs) as part of the IKE_SA_INIT exchange.
  - The transform types used in the negotiation are as follows:
    - Encryption algorithm
    - Integrity algorithm
    - Pseudo-Random Function (PRF) algorithm
    - Diffie-Hellman (DH) group
- **IKEv2 Policy**
  - Contains proposals that are used to negotiate the encryption, integrity, PRF algorithms, and DH group in the IKE_SA_INIT exchange.
  - It can have match statements, which are used as selection criteria to select a policy during negotiation.
  - Applied per a peer neighbor

Internet Key Exchange Version 2 - CLI Constructs

- **IKEv2 Profile**
  - An IKEv2 profile is a repository of nonnegotiable parameters of the IKE SA, such as local or remote identities, authentication methods and services that are available to authenticated peers that match the profile.
  - An IKEv2 profile must be attached to either a crypto map or an IPSec profile on the initiator.
  - An IKEv2 profile is not mandatory on the responder.
  - Applied per a peer neighbor
  - *Note: In my case, I'm using the IP address as the identity of my peers.*
- **IPsec transform set**
  - Transform Set is used to define how the data traffic between IPSec peers is going to be operated and protected.
- **ACL**
  - Crypto ACL is just an ACL created to identify interesting traffic that starts the IPsec tunnel initialization.
- **Crypto Map**
  - Crypto Maps are used to connect all the pieces of IPSec configuration together. A Crypto Map consists of one or more entries as an ACL, Transform Set, Remote Peer, the lifetime of the data connections etc

https://nil.uniza.sk/site-to-site-ikev2-ipsec-vpn-using-pre-shared-key-authentication-simple-configuration-example-on-cisco-routers/

Example



Example – router's init config

```
!BRanch
ena
conf t
hostname BRANCH
int g1
ip add 192.168.1.1 255.255.255.0
ip nat inside
no shut
int g2
ip add 209.165.200.242
255.255.255.248
ip nat outside
no shu
exit
ip access-list extended NAT
10 deny ip 192.168.1.0 0.0.0.255
10.10.10.0 0.0.0.255 log
20 permit ip 192.168.1.0 0.0.0.255
any log
ip nat inside source list NAT int g2
overload
ip route 0.0.0.0 0.0.0.0 g2
209.165.200.241
ip dhcp pool LAN
network 192.168.1.0 /24
default-router 192.168.1.1
line con 0
logging synchronous
```

```
!HQ
ena
conf t
hostname HQ
int g1
ip add 10.10.10.1 255.255.255.0
ip nat inside
no shut
int g3
ip add 209.165.200.226
255.255.255.248
ip nat outside
no shu
exit
ip access-list extended NAT
10 deny ip 10.10.10.0 0.0.0.255
192.168.1.0 0.0.0.255 log
20 permit ip 10.10.10.0 0.0.0.255 any
log
ip nat inside source list NAT int g3
overload
ip route 0.0.0.0 0.0.0.0 g3
209.165.200.225
ip dhcp pool LAN_HQ
network 10.10.10.0 /24
default-router 10.10.10.1
Line con 0
logging synchronous
```

```
!ISP
ena
conf t
hostname ISP
int g2
ip add 209.165.200.241
255.255.255.248
no shut
int g3
ip add 209.165.200.225
255.255.255.248
no shu
exit
Line con 0
logging synchronous
end
wr mem
```

Configuring IKEv2 keyring

```
BRANCH(config)#crypto ikev2 keyring KEYRING_1
! thare can be several peers identified several ways,
! i'm using peer IP address
BRANCH(config-ikev2-keyring)# peer HQ_ROUTER
BRANCH(config-ikev2-keyring-peer)# address 209.165.200.226
BRANCH(config-ikev2-keyring-peer)# pre-shared-key MY_PASS_cisco123
```

```
HQ(config)# crypto ikev2 keyring KEYRING_1
HQ(config-ikev2-keyring)# peer BRANCH_ROUTER
HQ(config-ikev2-keyring-peer)# address 209.165.200.242
HQ(config-ikev2-keyring-peer)# pre-shared-key MY_PASS_cisco123
```

Configuring IKEv2 proposal

```
!proposal
BRANCH(config)#crypto ikev2 proposal MY_IKEV2_PROPOSAL
IKEv2 proposal MUST either have a set of an encryption algorithm other than
aes-gcm, an integrity algorithm and a DH group configured or
 encryption algorithm aes-gcm, a prf algorithm and a DH group configured
BRANCH(config-ikev2-proposal)#encryption aes-gcm-256
BRANCH(config-ikev2-proposal)#prf sha512
BRANCH(config-ikev2-proposal)#group 21
```

```
HQ(config)#crypto ikev2 proposal MY_IKEV2_PROPOSAL
IKEv2 proposal MUST either have a set of an encryption algorithm other than
aes-gcm, an integrity algorithm and a DH group configured or
encryption algorithm aes-gcm, a prf algorithm and a DH group configured
HQ(config-ikev2-proposal)#encryption aes-gcm-256
HQ(config-ikev2-proposal)#prf sha512
HQ(config-ikev2-proposal)#group 21
```

IKEv2 Policy

```
BRANCH(config)#crypto ikev2 policy BRANQ_TO_HQ_POLICY
IKEv2 policy MUST have atleast one complete proposal attached
BRANCH(config-ikev2-policy)#proposal MY_IKEV2_PROPOSAL
```

```
HQ(config)#crypto ikev2 policy HQ_TO_BRANCH_POLICY
IKEv2 policy MUST have atleast one complete proposal attached
HQ(config-ikev2-policy)# proposal MY_IKEV2_PROPOSAL
```

Configuring IKEv2 Profile

```
BRANCH(config)#crypto ikev2 profile IKE_BRANCH_TO_HQ_PROFILE
IKEv2 profile MUST have:
   1. A local and a remote authentication method.
   2. A match identity or a match certificate or match any statement.
BRANCH(config-ikev2-profile)#match address local 209.165.200.242
BRANCH(config-ikev2-profile)# match identity remote address 209.165.200.226 255.255.255.248
BRANCH(config-ikev2-profile)#authentication local pre-share
BRANCH(config-ikev2-profile)#authentication remote pre-share
BRANCH(config-ikev2-profile)#keyring local KEYRING_1
```

```
HQ(config)#crypto ikev2 profile IKE_HQ_TO_BRANCH_PROFILE
IKEv2 profile MUST have:
   1. A local and a remote authentication method.
   2. A match identity or a match certificate or match any statement.
HQ(config-ikev2-profile)# match address local 209.165.200.226
HQ(config-ikev2-profile)# match identity remote address 209.165.200.242 255.255.255.248
HQ(config-ikev2-profile)# authentication remote pre-share
HQ(config-ikev2-profile)# authentication local pre-share
HQ(config-ikev2-profile)# keyring local KEYRING_1
```

IPsec transform set

```
BRANCH(config)#crypto ipsec transform-set IPSEC_TR_SET esp-aes 256
BRANCH(cfg-crypto-trans)#mode tunnel
```

```
HQ(config)#crypto ipsec transform-set IPSEC_TR_SET esp-aes 256
HQ(cfg-crypto-trans)# mode tunnel
```

ACL

```
ip access-list extended ACL
  remark Preotect flows form Branch to HQ
  permit ip 192.168.1.0 0.0.0.255 10.10.10.0 0.0.0.255
```

```
ip access-list extended ACL
  remark Preotect flows form HQ to Branch
  permit ip 10.10.10.0 0.0.0.255 192.168.1.0 0.0.0.255
```

Crypto Map

```
BRANCH(config)#crypto map MY_MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
and a valid access list have been configured.
BRANCH(config-crypto-map)# set peer 209.165.200.226
BRANCH(config-crypto-map)# set transform-set IPSEC_TR_SET
BRANCH(config-crypto-map)# set ikev2-profile IKE_BRANCH_TO_HQ_PROFILE
BRANCH(config-crypto-map)# match address ACL

!apply
interface g2
  crypto map MY_MAP
```

```
HQ(config-if)#crypto map MY_MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
and a valid access list have been configured.
HQ(config-crypto-map)# set peer 209.165.200.242
HQ(config-crypto-map)# set transform-set IPSEC_TR_SET
HQ(config-crypto-map)# set ikev2-profile IKE_HQ_TO_BRANCH_PROFILE
HQ(config-crypto-map)# match address ACL

!Apply
interface g3
  crypto map MY_MAP
```

Verification

```
BRANCH#sh crypto ikev2 ?
  authorization Author policy
  certificate-cache Show certificates in
ikev2 certificate-cache
  client Show Client Status
  cluster Show Cluster load
  diagnose Shows ikev2 diagnostic
  policy Show policies
  profile Shows ikev2 profiles
  proposal Show proposals
  sa Shows ikev2 SAs
  session Shows ikev2 active session
  stats Shows ikev2 sa stats
```

```
BRANCH#sh crypto ipsec ?
  out-sa-hash IPsec Outbound SA Hash for
VESEN
  policy Show IPSEC client policies
  profile Show ipsec profile information
  sa IPSEC SA table
  security-association Show parameters for
IPSec security associations
  spi-lookup IPSEC SPI table
  transform-set Crypto transform sets
```

```
BRANCH#sh crypto session ?
  active Shows HA-enabled crypto sessions in the active state
  brief brief output
  detail detailed output
  fvrf Front-door VRF
  groups show all connected groups usage
  interface Show crypto sessions on the interface
  isakmp Show crypto sessions using the isakmp profile or group
  ivrf Inside VRF
  local Show crypto sessions for a local crypto endpoint
  remote Show crypto sessions for a remote IKE peer
```

sh crypto session detail

```
BRANCH#sh crypto session detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation
R - IKE Auto Reconnect, U - IKE Dynamic Route Update
S - SIP VPN

Interface: GigabitEthernet2
Profile: IKE_BRANCH_TO_HQ_PROFILE
Uptime: 00:37:20
Session status: UP-ACTIVE
Peer: 209.165.200.226 port 500 fvrf: (none) ivrf: (none)
      Phase1_id: 209.165.200.226
      Desc: (none)
  Session ID: 1
  IKEv2 SA: local 209.165.200.242/500 remote 209.165.200.226/500 Active
          Capabilities:U connid:1 lifetime:23:22:40
  IPSEC FLOW: permit ip 192.168.1.0/255.255.255.0 10.10.10.0/255.255.255.0
        Active SAs: 2, origin: crypto map
        Inbound:  #pkts dec'ed 40 drop 0 life (KB/Sec) 4607995/1362
        Outbound: #pkts enc'ed 40 drop 0 life (KB/Sec) 4607996/1362
```

UNIVERSITY OF ŽILINA
Faculty of Management Science and Informatics

KATEDRA INFORMAČNÝCH SIETÍ * FAKULTA RIADENIA A INFORMATIKY ŽU *

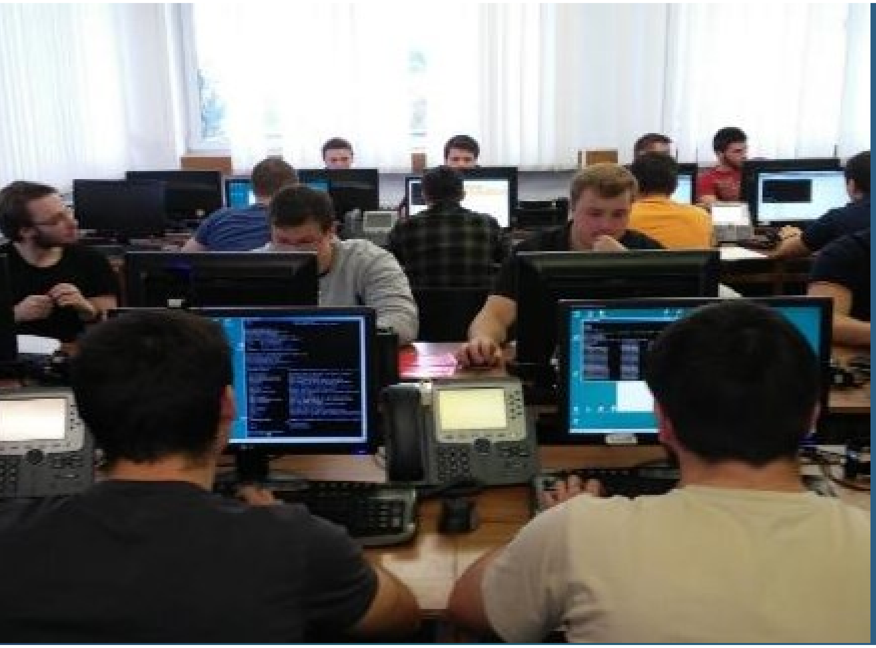
CISCO Networking Academy

Thank you for your attention, the following pictures are only for "network-knowledge-geedy ones" - CCNA 3 v7.0 does not cover

Rate our CNA Academy on google :

- https://goo.gl/maps/BAnFvQKYCBpffcEX7

--



Dynamic Multipoint VPN (DMVPN)

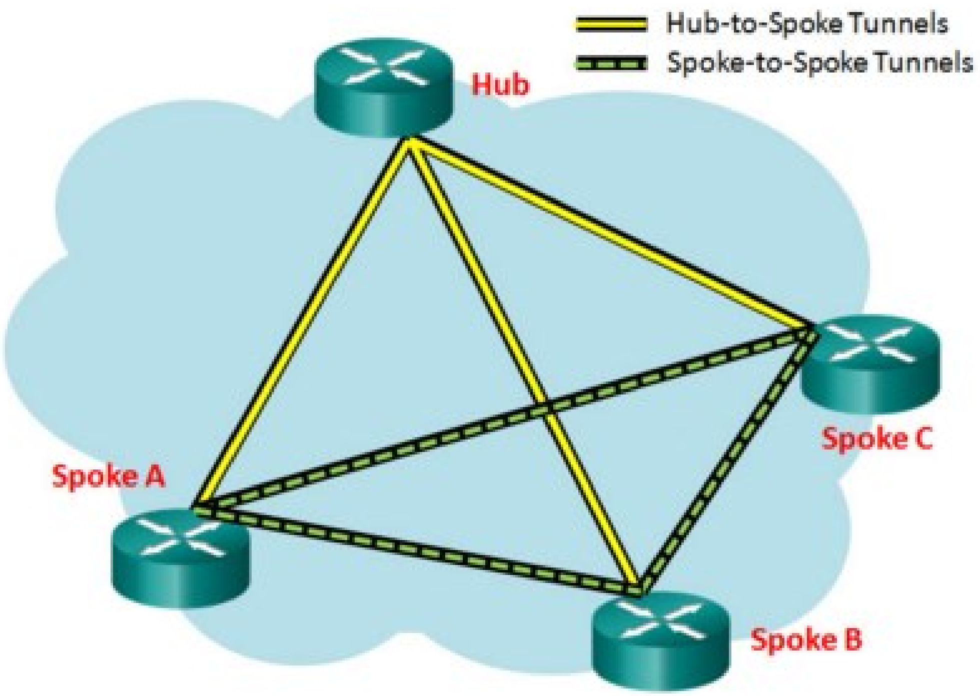
By MarMarc Khayat, CCIE #41288

Technical Manager, Cisco Networking Academy

- A little updated

Why DMVPN?

- To have efficient spoke-to-spoke communication in a hub-and-spoke topology.
- Dynamic tunneling
  - No more static configuration of separated p-t-p tunnels is required
  - Spoke-spoke
  - Hub-spoke

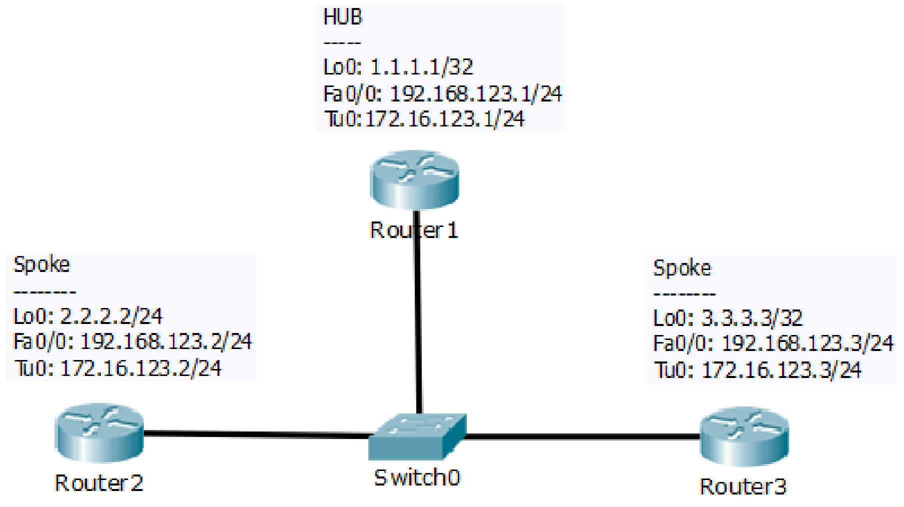


How are these tunnels built?

- Next Hop Resolution Protocol (NHRP)
- Multipoint Generic Routing Encapsulation (mGRE) tunnels
- IP Security (IPsec) encryption

Config Tasks

1. NHRP: set the hub as the server, allow multicast to flow to it.
2. mGRE tunnel config.
3. Enable IPSec encryption on the tunnels.



Hub and Spoke configuration example

```
!  Spoke config
crypto isakmp policy 1
 encr aes
 hash md5
 authentication pre-share
 group 2
crypto isakmp key MYKEY address 0.0.0.0
!
crypto ipsec transform-set MYSET esp-aes esp-md5-hmac
!
crypto ipsec profile MGRE
 set security-association lifetime seconds 86400
 set transform-set MYSET
!
interface Tunnel0
 ip address 172.16.123.1 255.255.255.0
 no ip split-horizon eigrp 10
 ip nhrp authentication CISCO
 ip nhrp map multicast dynamic
! Identifiy DMVPN net
! Have to be same on hub and spookes
 ip nhrp network-id 1
 tunnel source FastEthernet0/0
 tunnel mode gre multipoint
 tunnel protection ipsec profile MGRE
! No explicit tunnel destination required
!
router eigrp 10
 network 1.0.0.0
 network 172.16.0.0
```

```
! Hub konfig
crypto isakmp policy 1
 encr aes
 hash md5
 authentication pre-share
 group 2
crypto isakmp key MYKEY address 0.0.0.0
!
crypto ipsec transform-set MYSET esp-aes esp-md5-hmac
!
crypto ipsec profile MGRE
 set security-association lifetime seconds 86400
 set transform-set MYSET
!
interface Tunnel0
 ip address 172.16.123.2 255.255.255.0
 ip nhrp authentication CISCO
 ip nhrp map multicast dynamic
! the HUB tunnel address
 ip nhrp nhs 172.16.123.1

! Map tunnel address of Hub to its real and globally
! reachable IP address
 ip nhrp map 172.16.123.1 192.168.123.1
 ip nhrp map multicast 192.168.123.1
 ip nhrp network-id 1
 tunnel source FastEthernet0/0
 tunnel mode gre multipoint
 tunnel protection ipsec profile MGRE
!
router eigrp 10
 network 2.0.0.0
 network 172.16.0.0
```

Verification

```
Show dmvpn
! Not from the topology above
! Just an example

R1# show dmvpn
...

Tunnel0, Type:Hub, NHRP Peers:3,
 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
 1     172.16.25.2     192.168.0.2    UP 00:02:28 D
 1     172.16.35.2     192.168.0.3    UP 00:02:26 D
 1     172.16.45.2     192.168.0.4    UP 00:02:25 D
```