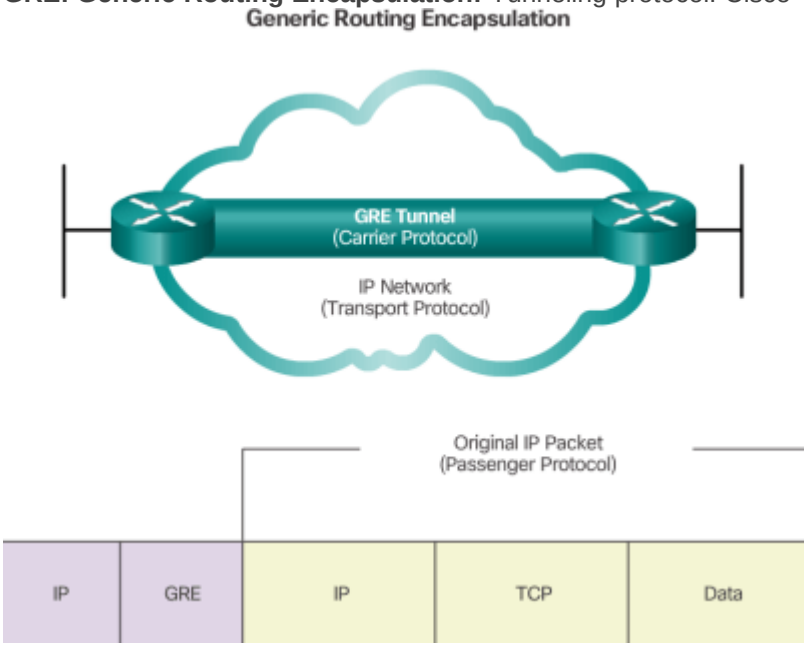
Post 9

Thursday, January 24, 2019 11:26 PM

SECURING SITE-TO-SITE CONNECTIVITY

1st VPNs: Strictly IP tunnels: Didn't auth/encrypt data
GRE: Generic Routing Encapsulation: Tunneling protocol: Cisco



- Encapsulates lots of network layer protocol packet types inside IP tunnels
- Creates virtual p-2-p link to rtr at remote points over internetwork

VPN GW: Necessary: Rtr/FW/Cisco ASA: Combines: FW/VPN concentrator/intrusion prevention into 1 SW img
Benefits:

Cost	3rd-party transport: Connect remote offices/usrs to main site • Eliminates $$ dedicated WAN links/modem banks • Reduced connectivity $: Increased remote connection BW
Scalability	Infrastructure w/in ISPs/devices: Easy to add usrs
Compatibility broadband	Mobile workers take advantage of: Access to org networks • Broadband: Flexibility/efficiency: Cost-effective solution for connecting remote offices
Sec	Sec mechanisms: Highest lvl sec using advanced encryption/auth protocols

2 types of VPN networks:

1. Site-to-site
2. Remote access

Site-to-Site	**Devices on both sides of VPN aware of VPN config in advance** • Remains static • Internal hosts: NO knowledge VPN exists • End hosts send/receive normal TCP/IP traffic through VPN GW **VPN gateway:** Encapsulates/encrypts outbound traffic for all traffic from particular site • GW sends through VPN tunnel over Internet to peer VPN GW at target site **Receipt:** • Peer VPN GW strips headers

	• Decrypts content • Relays packet to target host inside private network Extension of classic WAN network: Past: Leased line/Frame Relay connection was req to connect sites
Remote-Access	**When VPN info not statically set: Dynamic changing info: Enable/disable** • Telecommuters/extranet/consumer-to-business traffic • Support client/server arch • VPN client: Remote host: Gains sec access to enterprise via VPN server device at network edge **Used to connect individual hosts:** • Must access company network sec over Internet • SW may need install on mobile usrs stuff ○ Encrypted data sent over Internet to VPN GW at edge of target network **Receipt:** Behaves as site-to-site

GRE: Generic Routing Encapsulation
Basic/non-sec site-to-site VPN tunneling protocol

- Dev: Cisco: Can encapsulate wide variety of protocol packet types inside IP tunnels
- Virtual p-t-p link to Cisco rtrs at remote points over IP internetwork

Manages transportation of multiprotocol/IP multicast traffic bet 2/more sites:

- May only have IP connectivity
- Can encapsulate multiple protocol packet types inside IP tunnel

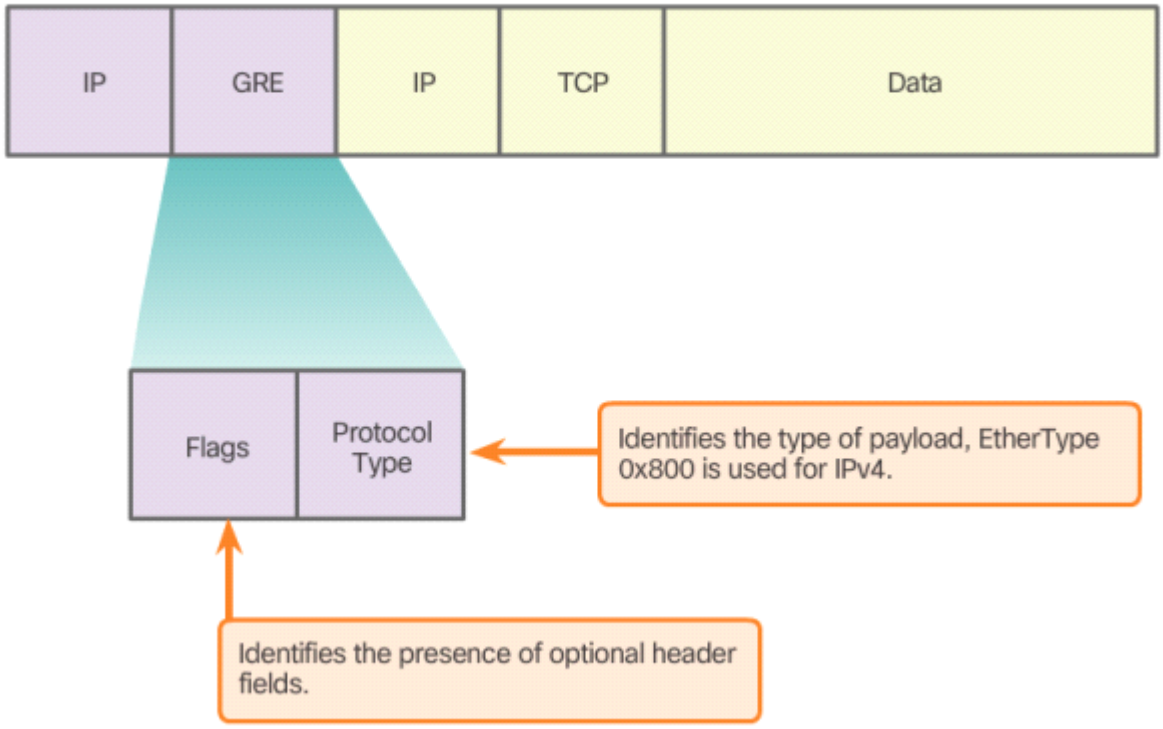
Tunnel int supports header for each:

- Encapsulated/passenger protocol: IPv4/IPv6/AppleTalk/DECnet/IPX
- Encapsulation protocol/carrier: GRE
- Transport delivery protocol: IP [protocol that carries encapsulated protocol]

GRE characteristics: IETF standard: RFC 2784

- **Outer IP header:** 47 used in protocol field to indicate GRE header will follow
- Encapsulation uses protocol type field in GRE header to support encapsulation of any L3 protocol
  - Protocol Types: RFC 1700 “EtherTypes”
- **Stateless: Default: Doesn’t include any flow-control mech**
- Doesn’t include any strong sec to protect payload
- **Header & tunneling IP header:** At least 24 bytes of addl overhead for tunneled packets

Header for GRE Encapsulated Packet Header



GRE Tunnel Config: Learn IP of endpoints:

1. Create tunnel int using **int tunnel number** cmd
2. Specify tunnel source/dest IP
3. Config IP for tunnel int
4. [Optional] Specify GRE tunnel mode as tunnel int mode
    - GRE tunnel mode: Default tunnel int mode for IOS

IP subnet must also be config to provide IP connectivity across link

R1(config)# int Tunnel0
R1(config-if)# tunnel mode gre ip
R1(config-if)# ip address 192.168.2.1 255.255.255.0
R1(config-if)# tunnel source 209.165.201.1
R1(config-if)# tunnel destination 198.133.219.87
R1(config-if)# router ospf 1
R1(config-router)# network 192.168.2.0 0.0.0.255 area 0

tunnel mode gre ip	Specifies mode of tunnel int is GRE over IP
ip address IP	IP addr of tunnel int

GRE Tunnel Verification
sh ip int br Determine whether tunnel int up/down
sh int tunnel Verify state of GRE tunnel: Line protocol on GRE tunnel int up as long as route to tunnel dest
sh ip ospf neighbor Verify OSPF has neighbors/also config/adjacency established
GRE: Considered VPN b/c priv network created by tunneling over public network
Advantages of GRE: Can be used to tunnel non-IP traffic over IP network: Allowing network expansion

- Connects multiprotocol subnetworks across single-protocol backbone env
- Supports IP multicast tunneling: Routing protocols can be used across tunnel:
    - Enables dynamic exchange of routing info in virtual network
- Common practice to create IPv6 over IPv4 GRE tunnels: IPv6 is encapsulated protocol/IPv4 is transport protocol

Downside: No encryption/sec: If sec data comm needed: IPsec/SSL VPNs should be config
R1# sh ip int br | include Tunnel
R1# sh int Tunnel 0
IPsec: Flexible/scalable connectivity

- Info from private network sec transported over public network
- Forms virtual network instead of using dedicated L2 connection
- Traffic encrypted to keep data confidential

IPsec: Framework of open standards that spells out rules for sec comms

- Not bound to any specific encryption/auth/sec alg/keying tech
- Relies on existing algs to implement sec comm
- Allows newer/better algs to be implemented w/out amending existing IPsec standards

Works at the network layer:

- Protecting/auth IP packets bet participating devices: AKA: Peers
- Sec path bet pair of GW's/hosts or GW/AND/host
- Can protect virtually all app traffic b/c protection can be implemented from L4-L7

All implementations of IPsec:

- Plaintext L3 header: No issues w/routing
- Functions over all L2 protocols: Ethernet/ATM/Frame Relay

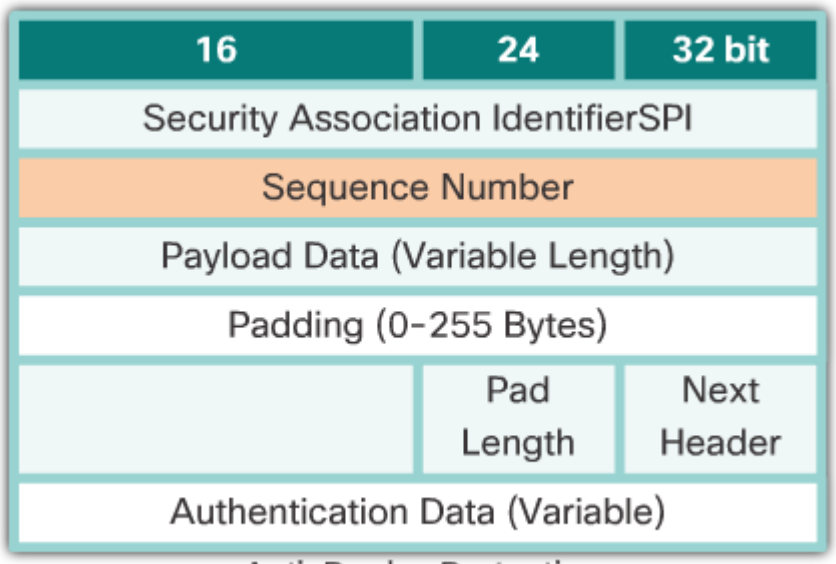
Characteristics:

- Framework of open standards that is alg-independent
- Data confidentiality/integrity/origin auth
- Acts at network layer: Protecting/auth IP packets

Sec Services:

Confidentiality	E**ncryption: Private data travels over public network** • Data confidentiality vital: Attained by encrypting data before transmitting across network • Strong encryption algs
Data Integrity	**Receiver can verify data was transmitted through Internet w/out change** • Just as impt to verify not changed in transit • Mech to ensure encrypted portion of packet/entire header/data portion of packet not changed • Ensures integrity by using checksums: Redundancy check • If tampering detected: Packet dropped
Auth	**Verify ID of src of data sent** • Guards against # of attacks that depend on spoofing • Auth ensures connection made w/desired comm partner • Receiver can auth packet: Certify src of info • **IKE: Internet Key Exchange** to auth usrs/devices that can carry out comm independently ○ Uses 7 types of auth: usrname/passwd/OTP/biometrics/PSK: Pre-Shared Key: Dig certs
Anti-Replay Protection	**Ability to detect/reject replayed packets: Helps prevent spoofing** • Verifies each packet unique/not duplicated • Protected by comparing seq # of received packets w/sliding win on dest host/sec GW • Packet that has seq # before sliding win considered to be late/duplicate: Dropped

CIA: Confidentiality/Integrity/Auth



Anti-Replay Protection

Confidentiality: VPN traffic: Confidential w/encryption

- Plaintext data transported over Internet can be intercepted/read
- Both sender/receiver must know rules used to transform original msg into coded form

Encryption Algs: DES/3DES: Not considered sec: Recommended AES used for IPsec encryption: Cisco: 256-bit AES

- Recommended using 2048-bit keys w/RSA if used during auth phase of IKE

Symmetric Encryption: AKA: Secret-key encryption: Symmetric key crypto: Encryption/decryption use same key

- Used to encrypt content of msg: DES/3DES/AES

Asymmetric Encryption: Public key crypto: Encryption/decryption use diff key: Used in digital cert/key mgmt: RSA

Diffie-Hellman Key Exchange: Data Integrity

- DH: Not encryption mech: Not typically used to encrypt data
- Method to sec exchange keys that encrypt data
- Algs allow 2 parties to establish shared secret key used by encryption/hash algs
- DH part of IPsec standard

OAKLEY: Protocol: Uses DH alg: Used by IKE: Part of framework: ISAKAMP Internet Sec Association/Key Mgmt Protocol

Integrity w/Hash Algs: Integrity/auth of VPN traffic handled by hash algs

- Ensures unauth persons don't tamper w/transmitted msgs
- VPNs use msg auth code to verify integrity/authenticity of msg w/out addl mechs

HMAC: Hash-based Msg Auth Code: Mech for msg auth using hash functions

- Keyed HMAC: Data integrity alg that guarantees integrity of msg

2 params:

1. Msg input
2. Secret key known only to msg owner/intended receivers

2 common HMAC algs:

MD5	128-bit shared secret key: Var-length msg/shared secret combined/HMAC-MD5 hash alg • Output 128-bit hash: Appended to original msg/fwded to remote end
SHA	-1: 160-bit secret key: Var-length msg/160-bit shared key combined/HMAC-SHA1 hash alg • Output 160-bit hash: Appended to original msg/fwded to remote end

IPsec Auth: Device on other end of VPN tunnel must be auth before comm path considered sec

2 peer auth methods:

PSK	**Secret key shared bet 2 parties using sec chan before used** • Pre-shared keys: Symmetric key crypto algs • PSK entered into each peer manually/used to auth peer • At each end: PSK combined with other information to form the authentication key.
RSA sigs	**Digital certs exchanged to auth peers: Local device derives hash/encrypts it w/private key** • encrypted hash/digital sig attached to msg/fwded to remote end

	• At remote end: encrypted hash decrypted using public key of local end • If decrypted hash matches recomputed hash: sig good

IPsec uses RSA (public-key cryptosys) for auth in context of IKE

- RSA sig method uses digital sig setup in which each device digitally signs a set of data/sends it to other party
- RSA sigs use CA to generate unique-id digital cert assigned to each peer for auth
- ID digital cert similar in function to PSK: Provides much stronger sec
- Each initiator/responder to IKE session using RSA sigs sends its own ID/digital cert/RSA sig value
- Consists of variety of IKE values: Encrypted by negotiated IKE encryption method (ex. AES)
- DSA: Digital Sig Alg: Another option

IPsec Protocol Framework: Describes msging to sec comm: Relies on existing algs

2 main IPsec protocols:

AH	**AH: Auth Header:** Protocol when confidentiality not req/perm • Provides data auth/integrity for IP packets passed bet 2 sys • AH doesn't provide data confidentiality of packets: Encryption • Txt transported plaintext
ESP	**Encapsulation Sec Payload:** Provides confidentiality/auth by encrypting packet • Conceals data/ID's src/dest • Auths inner IP/ESP header • Auth provides data origin auth/data integrity • Although both encryption/auth optional: Min: 1 must be selected

Building blocks IPsec framework must be selected:

IPsec	When config IPsec GW to provide sec services: IPsec protocol must be selected • Choices some combo of ESP/AH • ESP/ESP+AH options almost always selected b/c AH itself doesn't provide encryption
Confidentiality	If IPsec implemented w/ESP: AES strongly recommended w/AES-GCM providing greatest sec
Integrity	Guarantees content hasn't been altered in transit: Implemented through use of hash algs: MD5/SHA
Auth	Represents how devices on either end of VPN auth: 2 methods: PSK/RSA
DH alg group	How a shared key established bet peers: 7 options: DH24 provides greatest sec

Types of Remote-access VPNs

2 methods for deploying remote-access VPNs:

1. SSL
2. IPsec

Cisco SSL VPN: IOS SSL VPN is 1st rtr-based SSL VPN solution: Offers "anywhere" connectivity

SSL supports various crypto algs for ops:

- Examples: Auth server/client to each other/transmitting certs/establishing session keys

Remote-access connectivity features:

- Customized remote access
- Protection against viruses/worms/spyware/hackers on VPN connection: Reduces cost/mgmt
- Use of single device for SSL VPN/IPsec VPN
    - Reduces cost/mgmt: Robust remote access/site-to-site VPN services from single platform w/unified mgmt

Remote access by using browser/native SSL encryption: Remote access using Cisco AnyConnect Sec Mobility Client SW

ASA provides 2 modes found in Cisco SSL VPN:

AnyConnect Sec Mobility Client w/ SSL	Req Cisco AnyConnect Client
Sec Mobility Clientless SSL VPN	Req browser
ASA	Config to support SSL VPN connection

AnyConnect Sec Mobility Client w/SSL

Client-Based SSL VPN: Auth usrs w/LAN-like, full network access to corporate resources

- Remote devices req client app: Cisco VPN Client/AnyConnect client installed on usr device

ASA config: Full tunneling/remote access SSL VPN

- Remote usrs use AnyConnect to establish SSL tunnel w/Cisco ASA

- ASA establishes VPN w/remote user: Can fwd traffic into SSL tunnel
- AnyConnect Client creates virtual network int to provide func

Cisco Sec Mobility Clientless SSL VPN

- Provide access to corporate resources even when device not corporately-managed
- ASA used as proxy device to network resources: Provides web portal int for devices to navigate using port-fwding
- ASA clientless SSL VPN: Remote usrs employ standard browser to establish SSL session w/ASA
    - ASA presents usr w/portal over which usr can access internal resources
    - Usr can access only some services

Cisco Easy VPN: Flexibility/scalability/ease of use for both site-to-site/remote access

Consists of 3 components:

Easy VPN Server	IOS rtr/ASA FW acting as VPN head-end device in site-to-site/remote
Easy VPN Remote	IOS rtr/ASA FW acting as remote VPN client
VPN Client	App supported on PC used to access VPN server

- Negotiates tunnel params
- Establishes tunnels according to set params
- Auth usrs by usrnames/group names/passwds
- Manages sec keys for encryption/decryption
- Auth/encrypt/decrypt data through tunnel

Comparing IPsec/SSL: IPsec/SSL VPN tech offer access to virtually any network app/resource

	SSL	**IPsec**
Applications	Web-enabled apps/file sharing/email	All IP-based apps
Encryption	Moderate-Strong: Key lengths 40-256bits	Strong: Key lengths from 56-256bits
Auth	Moderate: 1-way/2-way auth	Strong: 2-way auth using shared secrets/dig certs
Connection Complexity	Low: Only web browser	Medium: Can be challenging to nontech usrs
Connection Options	Any device can connect	Only specific devices w/specific configs