Composants fondamentaux d'une architecture sécurisée. Tunnels, VPN et IPSec **TLS-SEC**
Fiche de révision, examen 16/12/2022 10h00 N7, une feuille A4 papier autorisée

Tunnels : Encapsulation d'un protocole (généralement de niv. 2 / 3) dans un protocole de niv. 3
Deux méthodes classiques de tunnel :
- IP in IP : *prévu dans Ipv4/6, mélange Ipv4/6, deux en-tête Ips qui se suivent*
- Generic Routing Encapsulation (GRE) : *en-tête entre deux en-têtes IP*
  - *N'importe quel protocole peut être encapsulé*
  - *Permet un chiffrement, pas très robuste ni polyvalent*

Exemple : Relier 2 réseaux privés 192.168.X.X via un WAN 172.16.X.X (routeur A seulement) :
```
# ip tunnel add montunnel gre remote 172.16.2.1 local 172.16.1.1
# ip link set montunnel up
# ip addr add 10.0.0.1 dev montunnel
# ip route add 192.168.2.0/24 dev montunnel
```
Il est aussi possible de mettre des tunnels sur tout et n'importe quoi pour s'accommoder d'une infrastructure (e.g. IP over DNS).

VPN : Interconnexion de réseaux locaux via un tunnel sécurisé
Protocoles dédiés : *PPPoE*, *PPTP*, *L2TP*, *IPSec*, *MPLS*

PPPoE (Point to Point Protocol over Ethernet)
PPP : Gestion de l'initialisation et du contrôle de connexion
Authentifications : PAP CHAP EAP → PAP & CHAP claqué, EAP(SRP-SHA1) à préférer
Chiffrement : Microsoft Point-to-Point Encryption protocol (MPPE)
→ RC4 (très mauvais), 40 ou 128 bits, attaque bit flip, pas cassé mais attaqué

PPTP (Point-to-Point Tunneling Protocol)
→ Canal de contrôle de connexion (TCP 1723) + tunnel PPP
→ On envoie les messages PPP via GRE
Sécurité : Initialement PPP (cassé), couche ajoutée (cassé aussi (dommage, bien tenté))

PPP over SSH : Un VPN fait maison de papy
→ pppd + sshd : plutôt simple, mais IP over TCP pas terrible

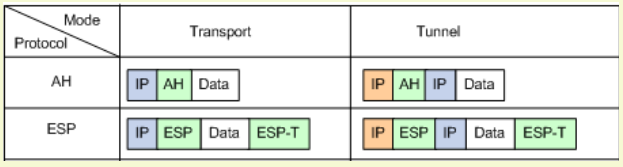
Alternative **VPN de OpenSSH** (attention technique de barbu, également vu en cours SSH) :
```
PermitRootLogin yes & PermitTunnel yes > sshd config                (serveur)
Tunnel yes & TunnelDevice any:any > ssh_config                      (client)
# ssh root@172.20.2.3 -w 0:0 -N                                     (client)
# ifconfig tun0 10.2.2.1 netmask 255.255.255.252                    (client)
# ifconfig tun0 10.2.2.2 netmask 255.255.255.252                    (serveur)
# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE              (serveur)
# route add -net reseauB tun0                                       (client)
```
Autres options : **OpenVPN** ou **IPSec** (super transition carlos)

IPSec

Sécurité
- Authentification : mano / PSK (test), X509 ou ECDSA avec PKI (prod), Kerberos (windows)
- Chiffrement : DES (claqué), 3DES (lent), AES-[CBC|GCM]-[128|192|256]. GCM += intégrité
- Intégrité : MD5 (claqué), SHA1/256/384 (HMAC), AES-[GCM|GMAC]-*. GCM fait les deux.
- Échange de clés : DH ou ECDH (Elliptic Curve Diffie Hellman)

Protocoles
- Security Association (SA) : choix des algos, auth. et échange de clés
  - Internet Security Association and Key Management Protocol (ISAKMP) → un framework
  - Pre-shared keys, IKE (v1/2), Kerberized Internet Negotiation of Keys et autres
- Authentication Headers (AH) : ajout d'entêtes aux paquets IP pour authentification
  - Intégrité, authentification et anti-rejeux
  - Sur l'ensemble du paquet → n'aime pas le NAT
- Encapsulated Security Payload (ESP)
  - Chiffrement, authenticité intégrité et anti-rejeux
  - Ne concerne que la payload, le header IP n'est pas concerné

Modes de fonctionnement
Mode transport : ajout d'un en-tête (AH ou ESP) entre l'en-tête IP et le payload
→ Chiffrement possible du payload initial
→ Ne rajoute que quelques octets
Mode tunnel : Encapsulation du paquet IP original
→ Meilleur protection et intégrité



TP - Commandes et autres
```
#!/usr/sbin/setkey -f
flush; spdflush;
add 192.168.1.1 192.168.2.1 ah 0x200 -A hmac-sha256 0xef[...]92;          # AH
add 192.168.2.1 192.168.1.1 ah 0x300 -A hmac-sha256 0xef[...]92;          # AH
spdadd 192.168.1.1 192.168.2.1 any -P out ipsec ah/transport//require;    # AH
spdadd 192.168.2.1 192.168.1.1 any -P in ipsec ah/transport//require;     # AH
add 192.168.1.1 192.168.2.1 esp 0x201 -E aes-ctr 0xaa[..]2e;              # ESP
add 192.168.2.1 192.168.1.1 esp 0x301 -E aes-ctr 0xaa[..]2e;              # ESP
spdadd 192.168.1.1 192.168.2.1 any -P out ipsec esp/transport//require;   # ESP
spdadd 192.168.2.1 192.168.1.1 any -P in ipsec esp/transport//require;    # ESP
```

Firewalls

VLANs : Isolation niveau 2 (plage ports/MACs/Ips)

PVLANs : Isolation niveau 1 (isolated port)

Firewalls : plus précis, pas que du tout ou rien

Firewalls de type routeur

Stateless : regarde tous les paquets un à un

Statefull : prend en compte les paquets passés, regarde les établissements de connexions

→ transmission UDP, ICMP, fragmentation, port knocking (prononcer le k)

Applicatif : vérification du protocole

Problèmes

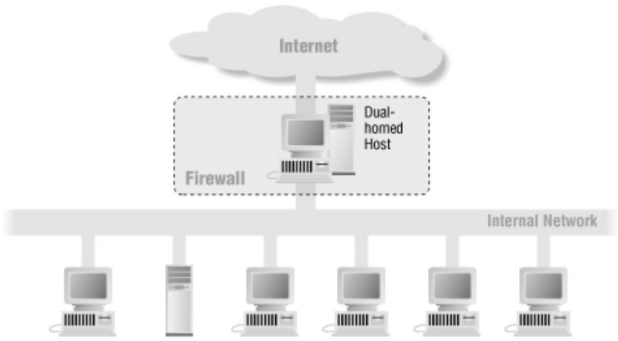
→ authentification basée sur l'IP = claqué au sol

→ aucun filtre sur les tunnels SSH / HTTP CONNECT

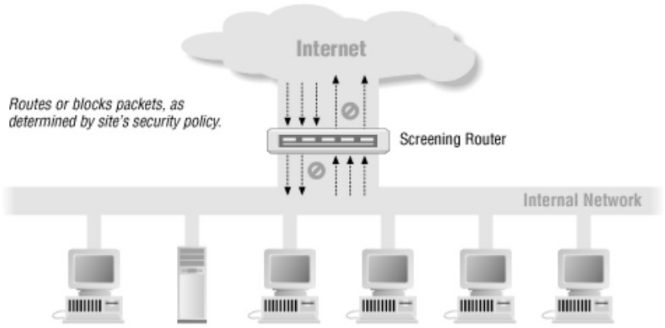
→ pas de défense en profondeur : les paquets originaux sont routés

⇒ utilisation de relais applicatif

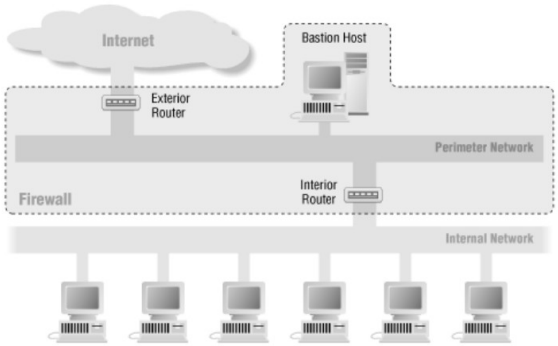
Architectures



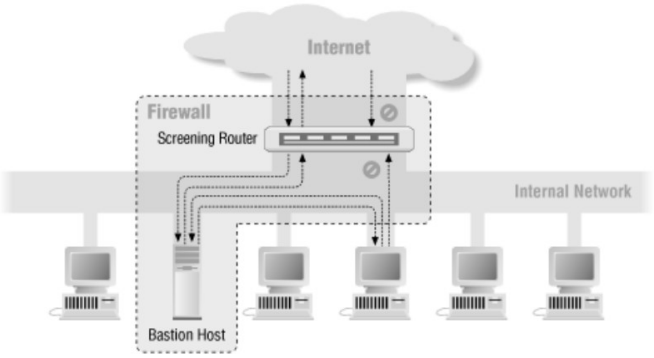
Bastion relais
(ne route pas les paquets)



Screening router
(filtre, bof bof)



DMZ *(le mieux, pas trop de pb. si les serveurs tombent)*



Screening router + bastion
(plus modulable et protège le bastion)

TP

\# Politique DMZ

ITF_INT=enp1s0f1 ; ITF_DMZ=enp1s0f3 ; IP_INT=195.168.2.0/24 ; IP_DMZ=195.168.1.0/24

iptables -X ; iptables -F ## Flush

Default: DROP, accept established

iptables -t filter -P FORWARD DROP

iptables -t filter -A FORWARD -m state --state ESTABLISHED -j ACCEPT

6.3. Filtrage ingress/egress (règles anti-spoofing)

iptables -t filter -A FORWARD -i $ITF_INT ! -s $IP_INT -j DROP

iptables -t filter -A FORWARD -i $ITF_DMZ ! -s $IP_DMZ -j DROP

6.1. Flux vers la DMZ : SMTP

iptables -t filter -A FORWARD -i $ITF_INT -s $IP_INT -o $ITF_DMZ -d $IP_DMZ -p tcp --destination-port 25 -m state --state NEW --syn -j ACCEPT

iptables -t filter -A FORWARD -i $ITF_DMZ -s $IP_DMZ -o $ITF_INT -d $IP_INT -p tcp --destination-port 25 -m state --state NEW --syn -j ACCEPT

...

IDS (Intrusion Detection System)

- NIDS (Network)
- HIDS (Host) (e.g. un antivirus juste observateur)

IPS (Intrusion Prevention System)

→ Ne bloque pas flux malveillant en soit, mais ajoute des règles de FW ou autre

NDR ⇒ Network Detection and Response, proche de l'IDS

EDR ⇒ Endpoint Detection and Response, j'ai ça sur moi, comment je dois agir ?

XDR ⇒ eXtended Detection and Response (NDR + EDR sous stéroïdes)

SIEM ⇒ Security Information Event Management (condensateur et management)