VPN TLS avec OpenVPN

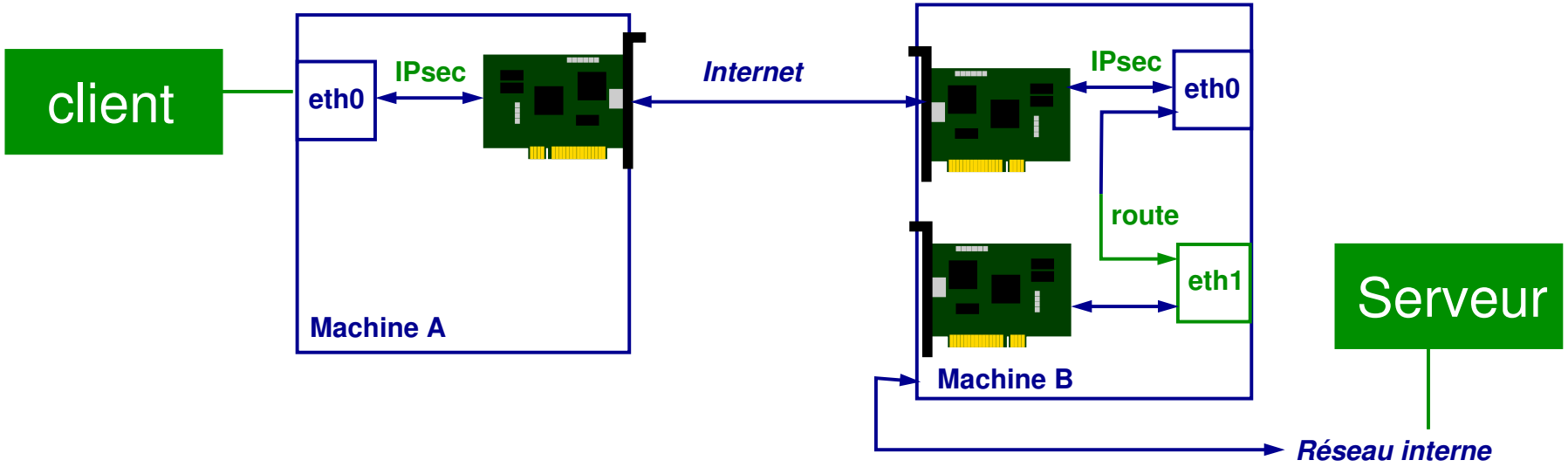
Matthieu Herrb

14 Mars 2005

Pour en finir avec IPSec

IPSec : sécurisation au niveau réseau.
- développé avec IPv6,
- protocoles spécifiques AH & ESP,
- modes tunnel et transport,
- protocole d'échange de clés (IKE) isakmpd/racoon (UDP),
- authentification au niveau machine,
- IPSec n'est pas un VPN,
- VPNs avec IPSec -> ajout de L2TP + PPP (Cisco+Microsoft),
- NAT-T pour le support de NAT - protocole supplémentaire,
- Trop complexe - pas de solution ouverte (B. Schneier),
- Problèmes d'interopérabilité,
- Déployement difficile...

Connexion IPsec



- Pas d'adresses IP dédiées.
- Comment reconnaitre un client utilisant IPsec ?

TLS

TLS (SSLv3) : sécurisation au niveau applicatif.
Technologie qui s'est imposée pour la sécurisation des échanges sur l'internet (**https**).

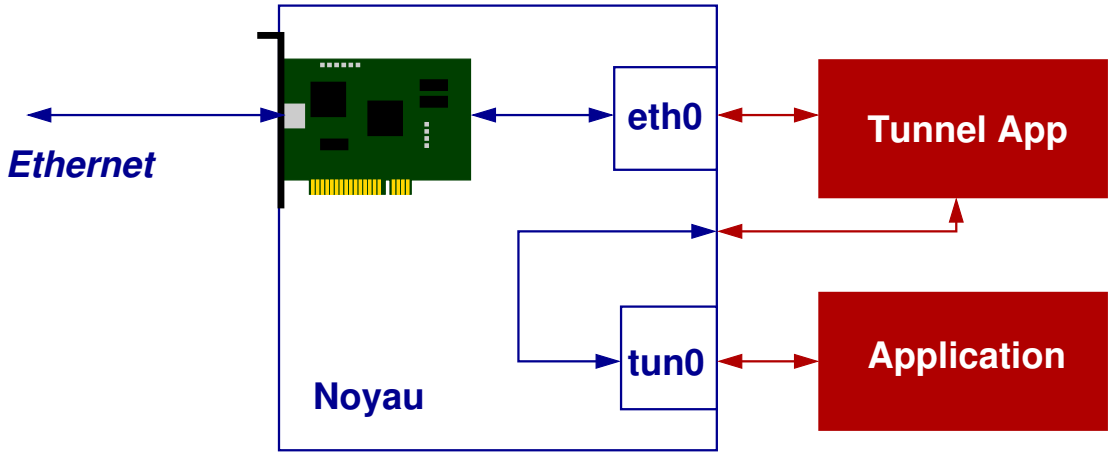
- au dessus des protocoles TCP ou UDP,
- authentification par certificats X509,
- négociation des protocoles et échange des clés lors de la phase de connexion.

Interfaces réseau virtuelles

TUN/TAP : interfaces réseau particulières :

- apparaissent comme des interfaces réseau au noyau,
- envoient et recoivent le trafic vers un programme utilisateur plutot que vers un support physique externe,
- TUN : niveau 3 - interface point à point similaire à PPP,
- TAP : niveau 2 - interface ethernet virtuelle (adresse MAC, protocole ARP, etc.)

Drivers disponibles sur la plupart des systèmes.



Réseaux privés virtuels SSL/TLS

Les réseaux privés virtuels SSL/TLS sont nés de la rencontre de ces deux technologies.

Principe :
- établir une connexion sécurisée par SSL (remplace IKE),
- créer à chaque extrémité une interface réseau virtuelle,
- encapsuler le traffic IP du réseau virtuel dans la connexion SSL,
- utiliser de préférence UDP comme protocole de base (éviter TCP dans TCP).

Plusieurs implémentations (incompatibles) : OpenVPN, vtun, solutions propriétaires,...

OpenVPN

http://openvpn.net/

- solution libre,
- multi-platformes (Windows, Linux, *BSD, MacOS X, Solaris),
- utilise OpenSSL pour la cryptographie,
- implémentation sécurisée (réduction des privilèges, chroot, ...),
- compression optionnelle avec la bibliothèque LZO,
- fonctionne en mode TUN (routage) ou TAP (pont),
- scriptable + interface de management,
- développement dynamique,
- version 2.0 $\rightarrow$ mode client/serveur avec plusieurs clients.

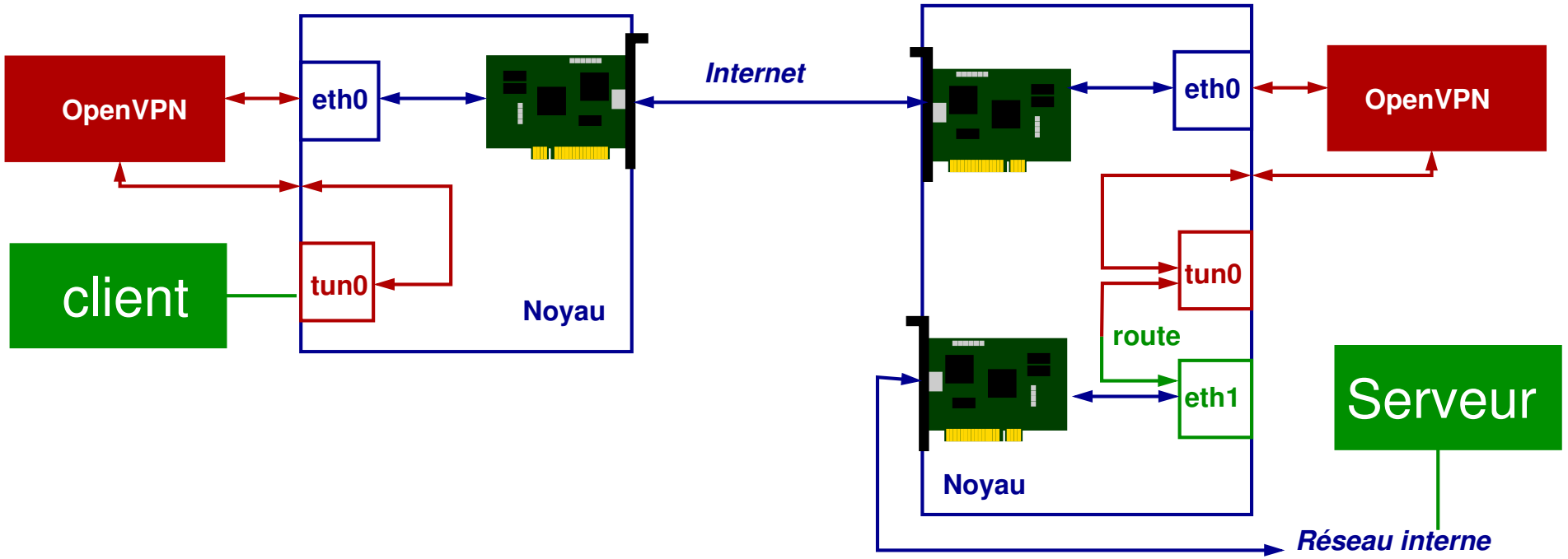
Plus :

- relativement simple à mettre en oeuvre,
- compatible avec les certificats CNRS.

OpenVPN : fonctionnalités

- Mode point à point ou client/serveur,
- Authentification mutuelle par certificats,
- Possibilité de secret partagé pour debug,
- Scripts pluggables lors de chaque étape (connexion, authentification, vérification certificat, déconnexion...),
- Le serveur peut se comporter comme un serveur DHCP pour envoyer des infos (DNS, routeur par défaut) aux clients,
- Possibilité de 'push' de commandes de configuration vers les clients,
- interface de controle (socket) pour créer des interfaces utilisateur,
- Adapte le MTU automatiquement,
- Passe le NAT.

Serveur OpenVPN routeur



eth0 adresse IP externe

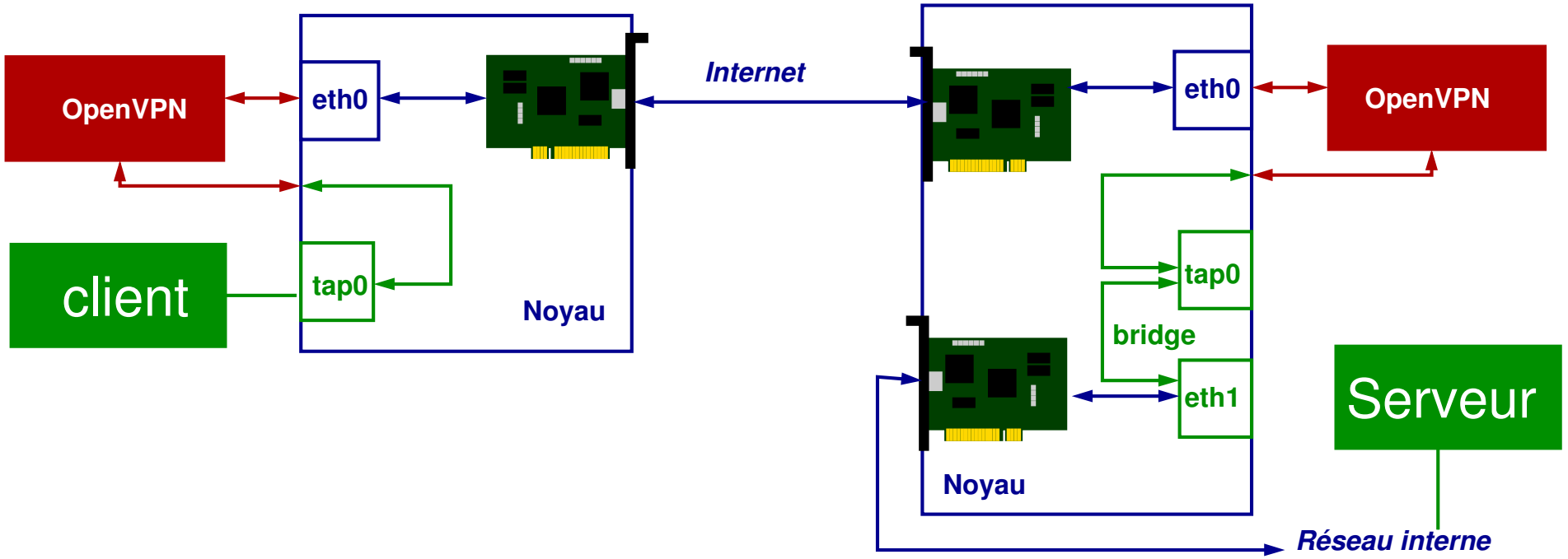
tun0 adresse IP interconnexion

eth0 adresse IP externe

tun0 adresse IP interconnexion

eth1 adresse IP interne

Serveur OpenVPN pont



eth0 adresse IP externe

tap0 adresse IP interne

eth0 adresse IP externe

tap0 adresse IP interne

eth1 adresse IP interne

Routeur ou pont ?

Bridge :
- un seul réseau IP,
- les broadcast sont diffusés dans le VPN - rend possible l'utilisation de protocoles tels que NetBIOS ou NIS,
- fonctionne avec tous les protocoles au dessus d'ethernet (IPX, AppleTalk,...
- simplifie les controles d'accès basés sur adresses IP,
- pas de routage à configurer.

Mais :
- moins efficace que le routage,
- ne s'adapte pas à l'échelle de très nombreux clients.

Dans les 2 cas, il vaut mieux que le serveur VPN soit aussi routeur.

Scénario 1 - réseau expérimental virtuel

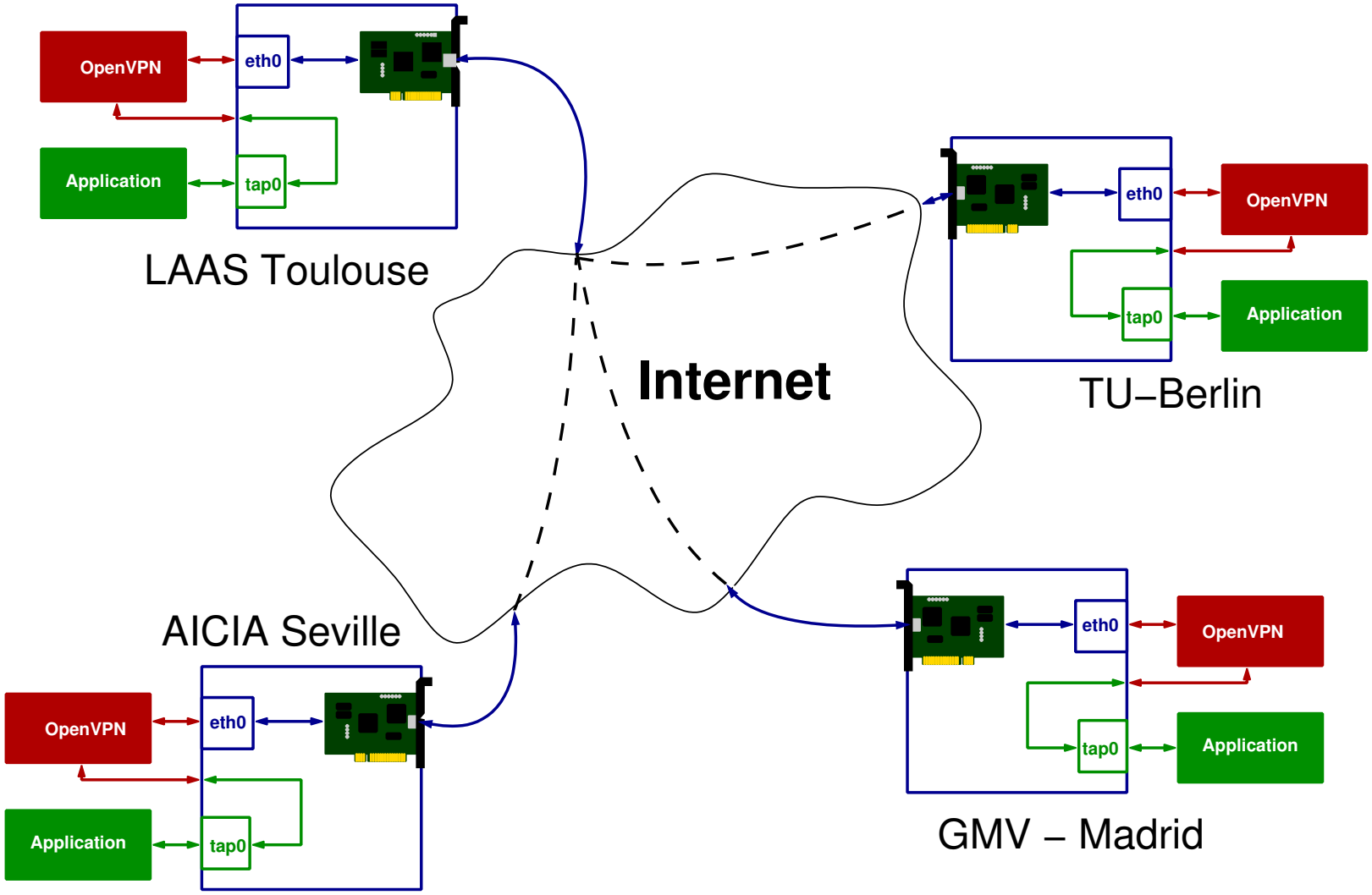
Quatre partenaires dans un projet européen
(COMETS http://www.comets-uavs.org).

Un réseau défini pour les manips d'intégration logicielles, utilisé lors des réunions physiques (192.168.100.0/24).

Besoin de tests via internet sans reconfigurer les logiciels. Passer simplement à travers les firewalls des partenaires. Machines Linux et Windows.

OpenVPN 2.0, mini-ca auto-signée, mode bridge, pas de routage vers l'extérieur du VPN.

Scénario 1 - architecture

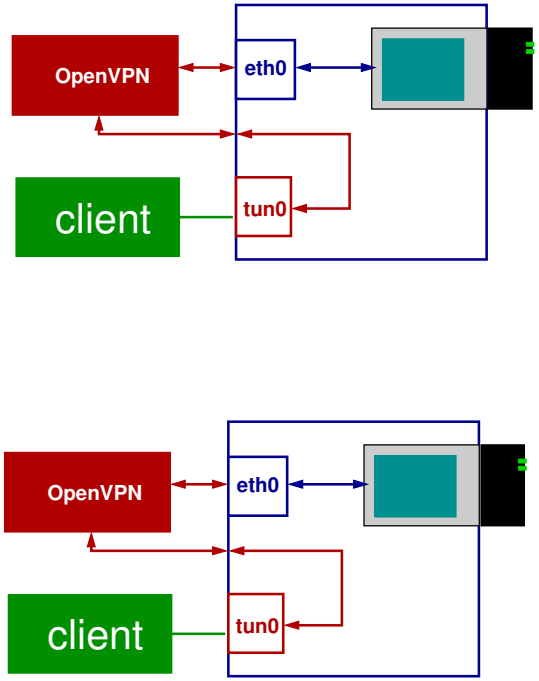


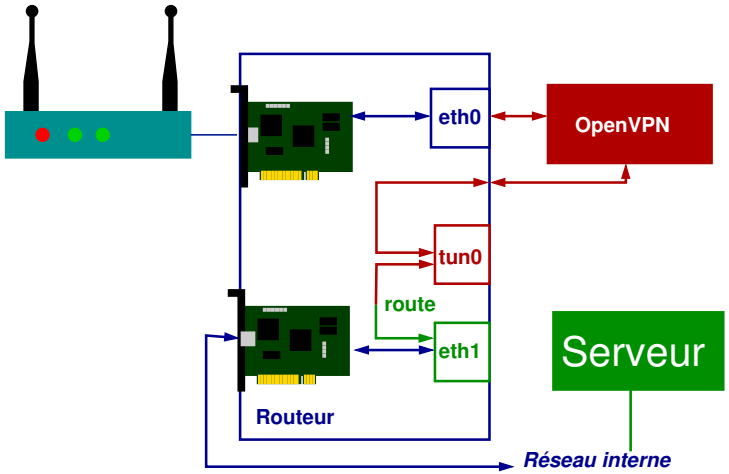
Scénario 2 - sécurisation d'un réseau Wifi

WEP, IEEE802.1X, WPA, Radius, etc, trop compliqués, pas fiable.

Remplacement par une passerelle wifi VPN.

Mode routage + éventuellement NAT sur la passerelle.



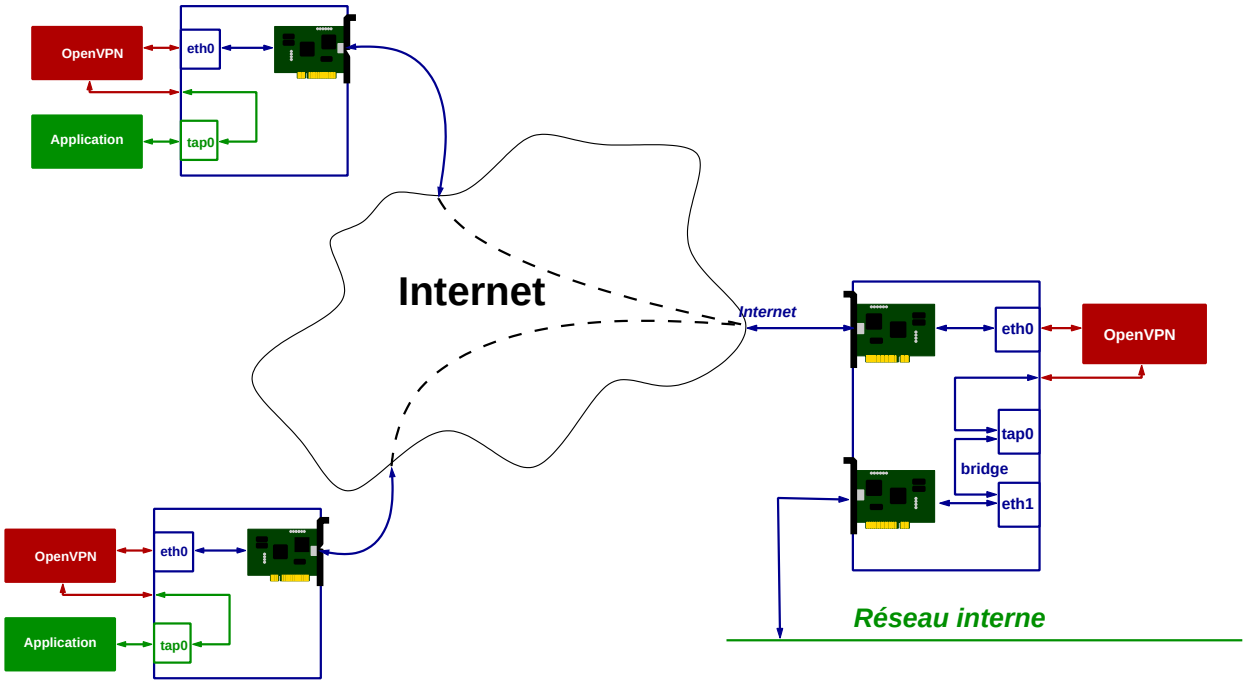


Scénario 3 - connexion de postes nomades

Le « graal » des VPN.

Mode bridge - les clients ont une adresse IP du réseau du laboratoire, voient les broadcasts.

Ne pas oublier de créer le bridge entre tap0 et l'interface interne.



Scénario 3 - serveur

```
port 1194
proto udp
dev tap
ca /etc/openvpn/ca.crt
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key
dh /etc/openvpn/dh1024.pem
mode server
tls-server
ifconfig 140.93.250.254 255.255.0.0
server-bridge 140.93.250.254 255.255.0.0 140.93.250.1 140.93.250.32
comp-lzo
user nobody
group nobody
chroot /var/empty
persist-key
persist-tun
ping 30
```

Scénario 3 - client

```
client
dev tap
proto udp
remote fogerty-dmz.laas.fr 1194
nobind
user nobody
group nobody
chroot /var/empty
persist-key
persist-tun
ca /home/matthieu/ssl/ca.crt
cert /etc/openvpn/cortez.herrb.com.crt
key /etc/openvpn/cortez.herrb.com.key
comp-lzo
```

Points durs

- nécessite de pouvoir passer du traffic UDP sur le port 1194
- sécurisation des postes clients...
- politique de routage sur les postes clients ?
- expiration/révocation/renouvellement des certificats...
- pas encore de support IPv6 (:-),
  (mais le mode bridge transporte l'IPv6 interne)
- reste malgré tout une solution propriétaire.

Expérimentations en cours au LAAS. Pour l'instant on utilise plutôt :

- webmail ou Thunderbird + certificats pour la messagerie
- SSH/SCP (WinSCP ou Fugu) pour l'accès aux fichiers
- SSH pour l'accès interactif

Bibliographie

- A cryptographic evaluation of IPsec, N. Ferguson and B. Schneier.
  http://www.schneier.com/paper-ipsec.html
- Installation d'une passerelle Linux/Configuration du service OpenVPN
  http://linbox.free.fr/chapitre9.html
- Building VPNs on the cheap, T. Greene
  http://www.nwfusion.com/news/2004/122004cheapvpn.html
- Meet OpenVPN H.C. Speel, Linux Journal
  http://www.linuxjournal.com/article/7949