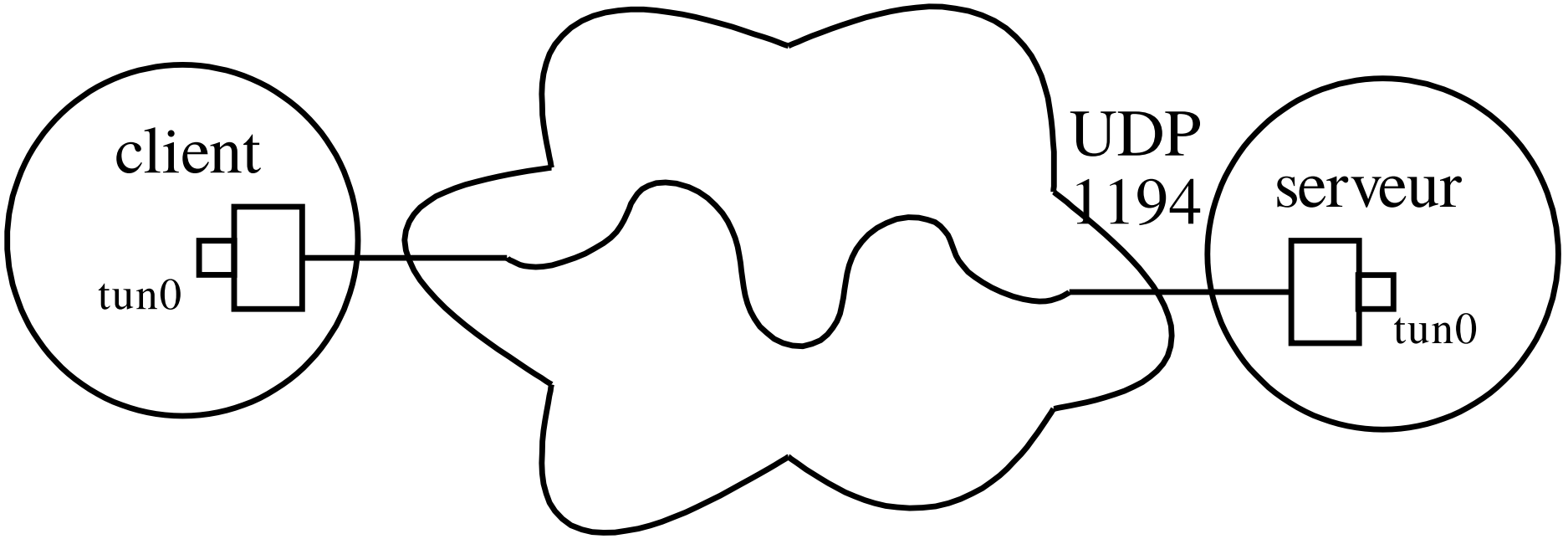
VPN

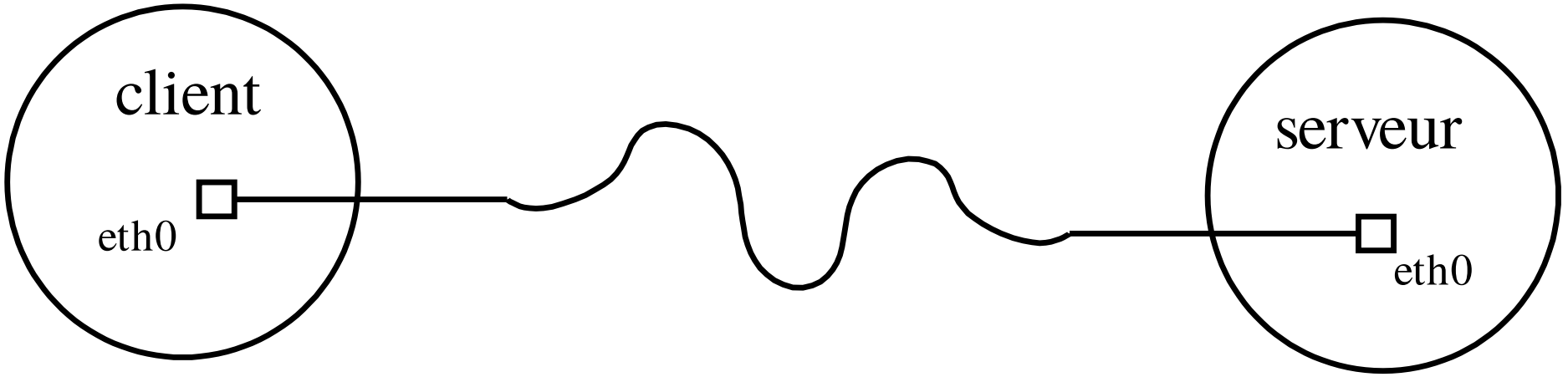
Mise en œuvre
Retour d'expérience FDN/Aquilenet

Slides sur www.ffdn.org/wiki ,
section Doc&outils
Lien Configuration OpenVPN

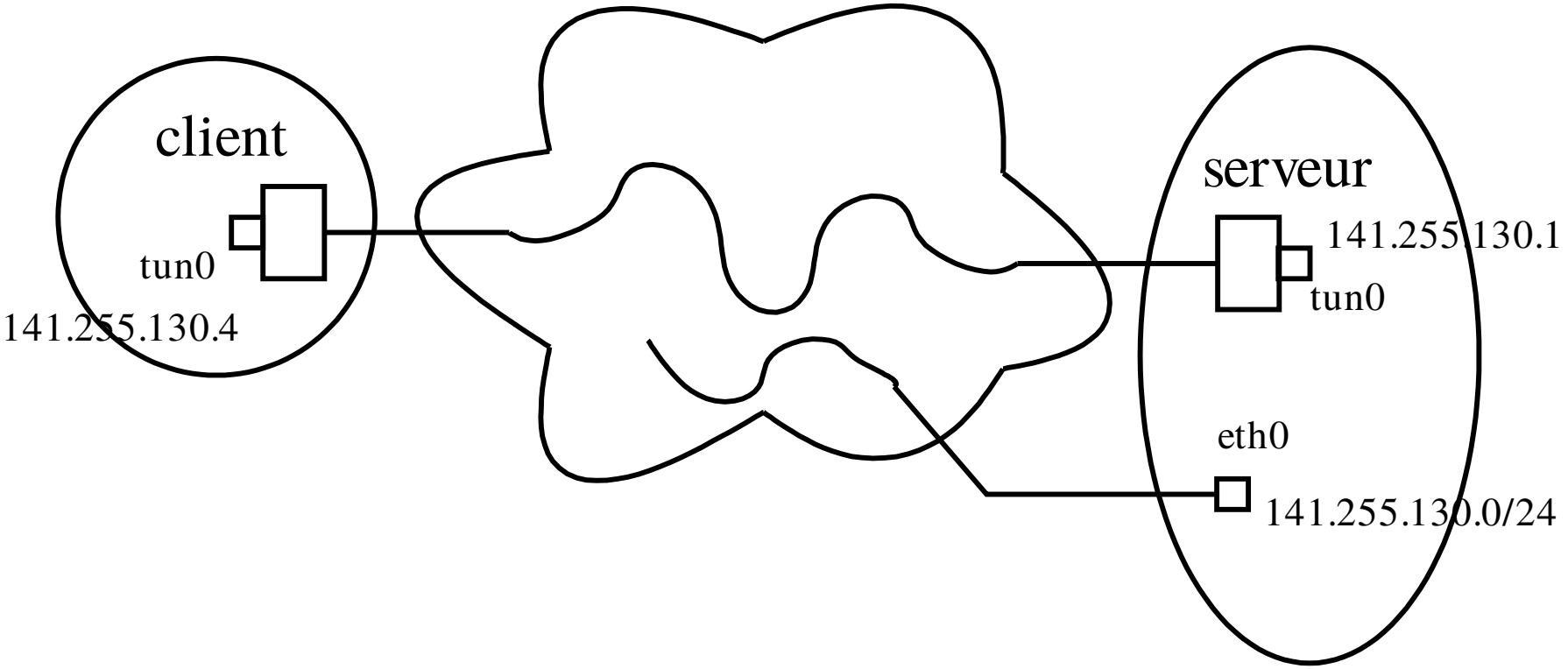
Rappel du principe de VPN



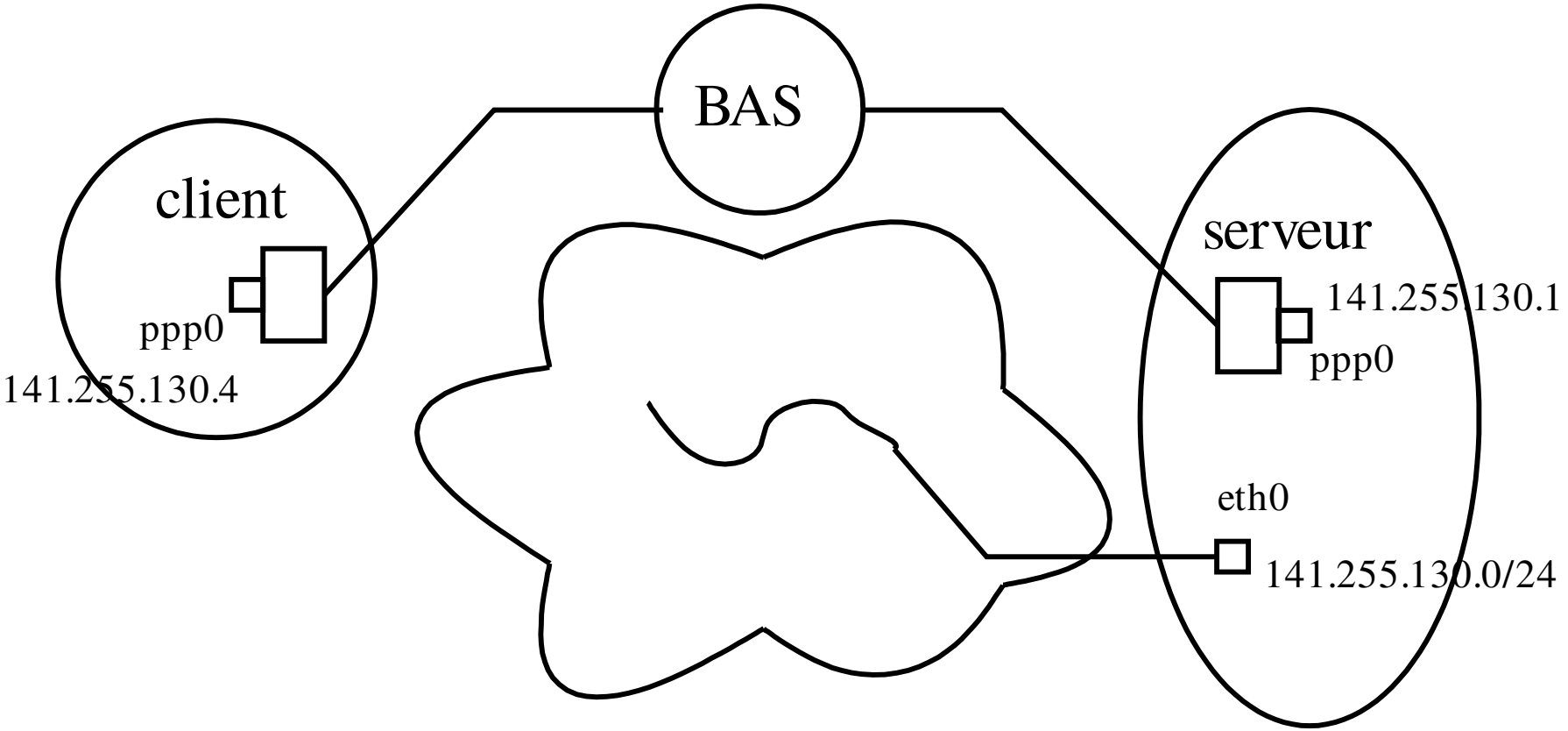
Rappel du principe de VPN



Rappel du principe de VPN



Rappel du principe de VPN



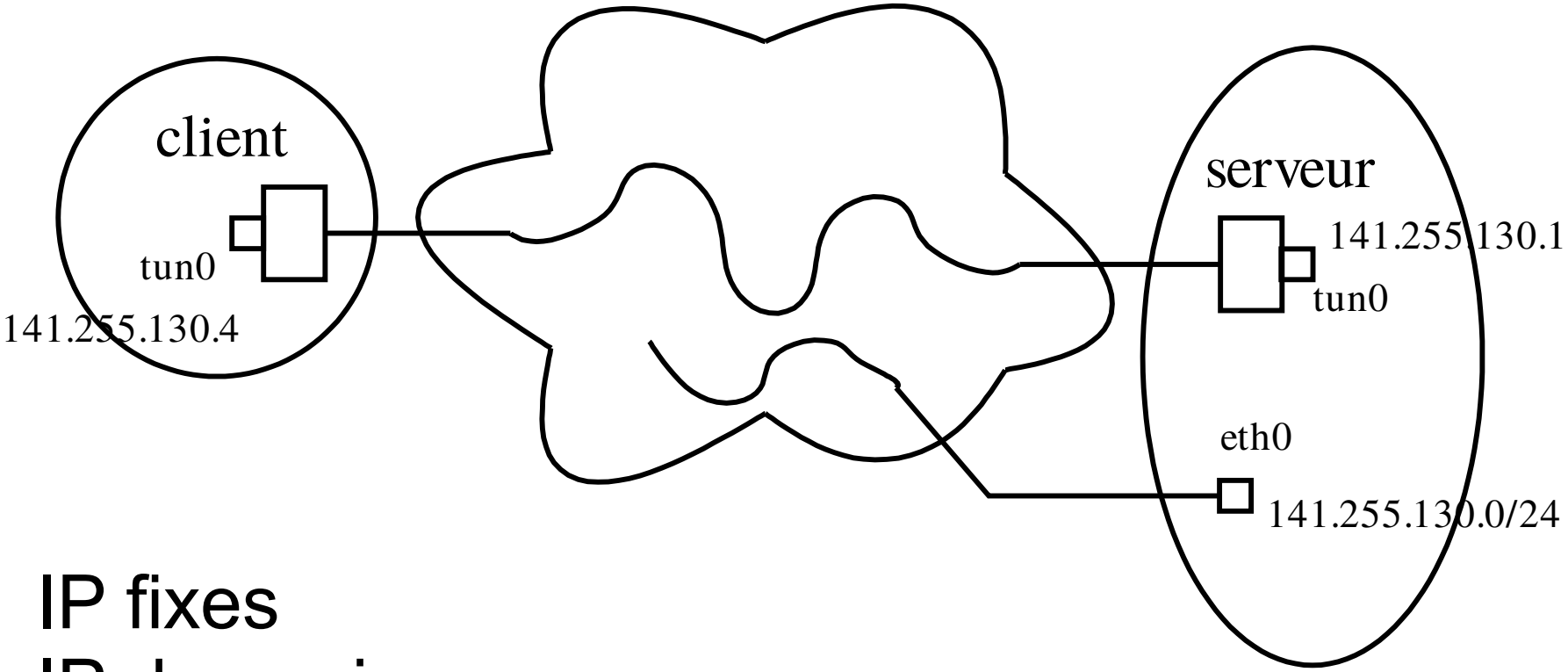
VPN comme un FAI, donc

- Trafic encapsulé dans un tunnel
  - L2TP pour ADSL
- Marge bien plus grande que l'ADSL
  - Mais attention, le trafic coûte deux fois !

Différents protocoles VPN

- PPTP
- L2TP
- tinc
- iodine
- …
- Tous incompatibles bien sûr :)
- Ici, OpenVPN

Attribution des IPs



IP fixes
IP dynamiques
Les deux

UDP et TCP

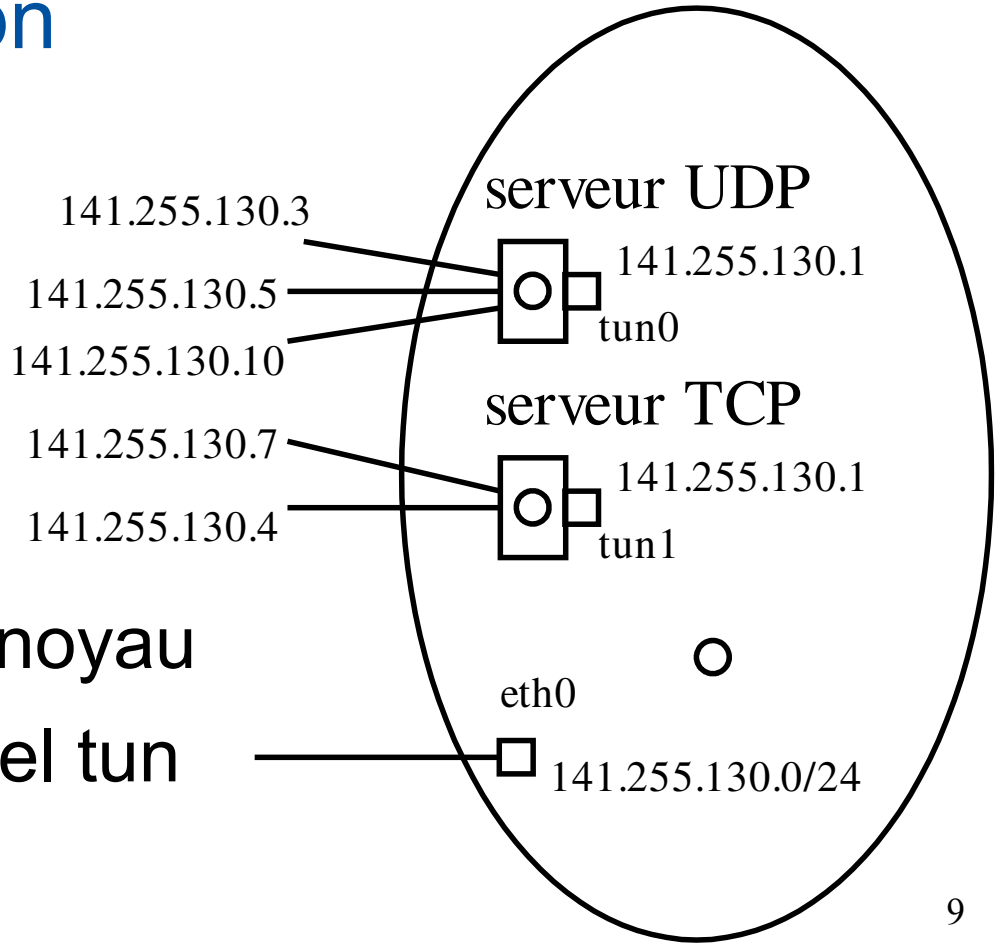
OpenVPN ne gère pas les deux à la fois

→ deux démons openvpn

Chacun a sa table de routage interne

Script client-connect

- Ajouter la route au noyau
- Qu'il sache vers quel tun



Attention Ips dynamiques

- Plusieurs démons openvpn
- Séparer les tranches où piocher les Ips
  - Sinon conflit d'allocation potentiel

IPv4/IPv6

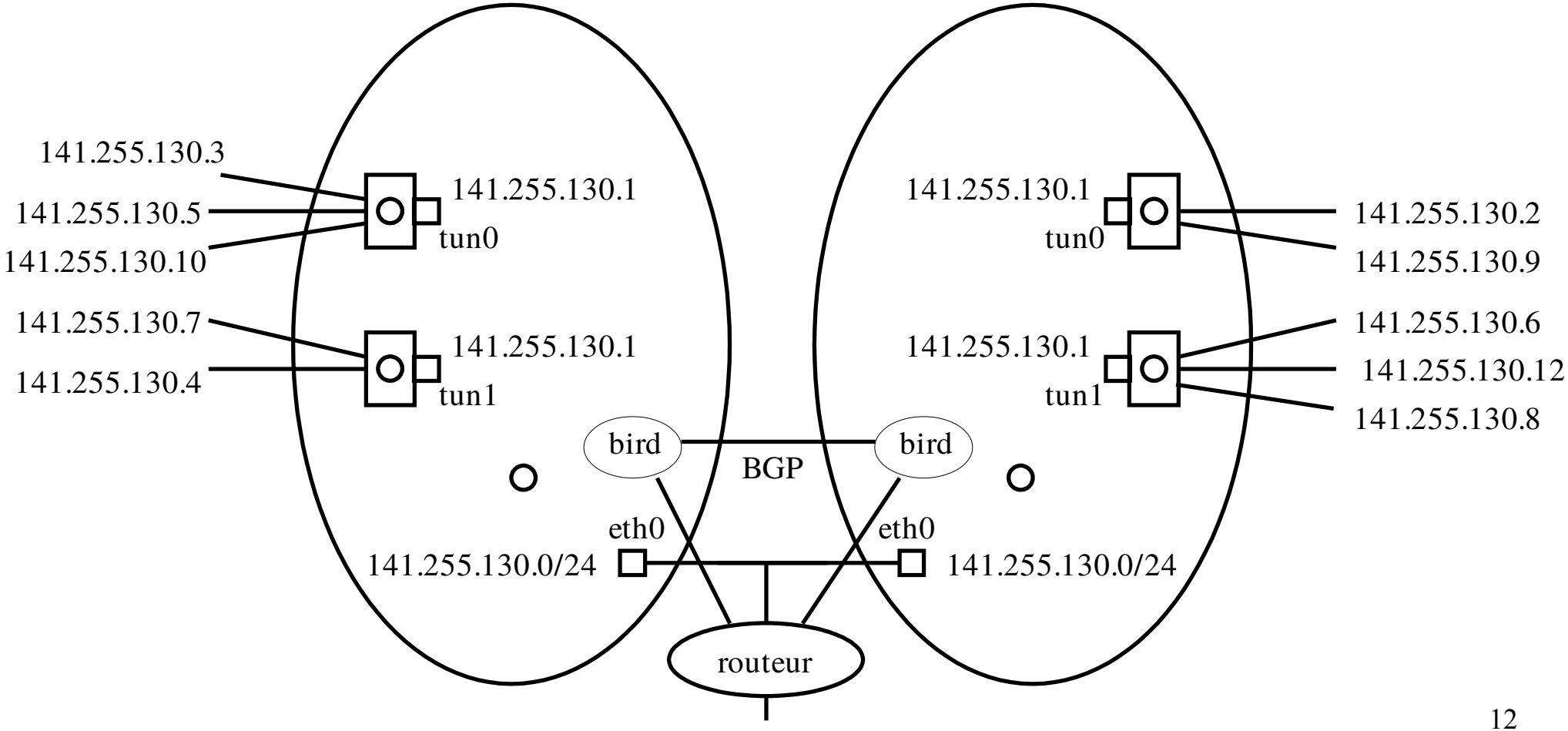
- OpenVPN gère les deux en même temps
- Un seul démon par UDP / TCP

```
proto udp6

proto tcp6-server
```

Redondance

Échange des routes en BGP



Redondance côté client

Diverses méthodes

- vpn.fdn.fr résoud en .45 et .57
  - Avec Round-robin DNS
  - Facile d'en ajouter
- Directement dans la conf client

  <connection>remote vpn1.fdn.fr</connection>

  <connection>remote vpn2.fdn.fr</connection>
  - Pas facile d'en ajouter…
- IP virtuelle
- Éventuellement `remote-random-hostname`

vpn-rw

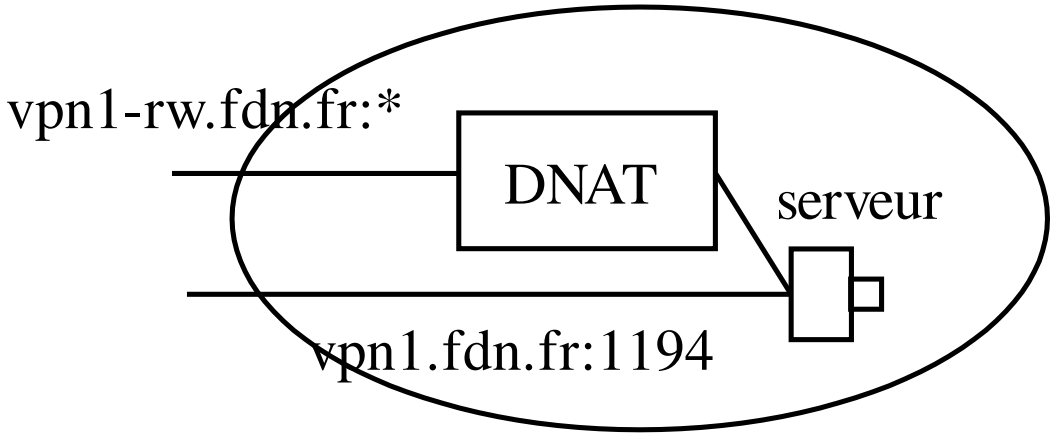
Écouter sur tous les ports

```
iptables -t nat -A PREROUTING -p udp -d vpn1-rw
  -j DNAT --to-destination vpn1:1194
```

Idem TCP

- Éventuellement, 22222 NATé vers 22 avant

Idem ip6tables avec kernel ≥ 3.7 (Debian Jessie)



Choix du port, TCP/UDP

- En principe UDP port 1194
  - Mais environnements hostiles pas neutres
- Préférer UDP !
  - TCP over TCP doublonne les retransmissions, ça tue les chatons !
- TCP parfois nécessaire
  - Accès Internet ~~~~= TCP port 80 et 443
  - Parfois proxy transparent (avec man in the middle donc) sur 443 :(
  - Proxy non transparent, option openvpn **`http-proxy`**

Choix du port

VPN everywhere

- Projet de programmation U-Bordeaux
- Interface non-tech compliant
- Tester automatiquement les ports
  - Essai dernier port utilisé
  - Scan des ports classiques en parallèle
  - Scan aléatoire de tous les ports en parallèle
- Tests sur neutral-echos.aquilenet.fr
  - Sinon fail2ban réagit

Authentification / autorisation

Un fichier passwd

- Script maison `checkpass`
- Configuration .ccd pour les IPs fixes et routes

Authentification / autorisation

Un script client-connect / disconnect

- Echo d’options OpenVPN

Authentification / autorisation

RADIUS

- Authentification login/pass
- Remontée des IPs et routes
  - `Framed-IP-Address 141.255.130.4`
  - `Framed-IP-Netmask 255.255.255.0`
  - `Framed-IPv6-Route 2a01:474:4::/56`
  - `Framed-IPv6-Address 2a01:474:4:ffff::1`
    - Nécessite openvpn-auth-radius patché

Authentification / autorisation

LDAP

- Adresses et routes des objets dans la base
- Voir avec Illyse les schémas qu'ils ont défini

Certificats

Être sûr de se connecter au bon serveur

- Ne pas donner son pass à n'importe qui !

- Le serveur a sa paire clé privée/publique
- Le client
  - Vérifie la signature de la clé publique
    - Utilise une CA
      - Conf client <ca></ca>
  - Challenge le serveur

Comment faire son certificat ?

- Utiliser CA connue
  - Utiliser `verify-x509-name *.fdn.fr name`
  - Pas vraiment nécessaire en fait
    - On met le CA dans la conf client
- Utiliser CA perso
  - Attention à la date d'expiration
- Éviter de changer de CA
  - Éviter de faire changer la conf client

Qualité, débit

- En principe, pas de perte ?
  - Dépassement CPU
- Latence supplémentaire
  - Quelques 10ms
- Débit atteint
  - Obtenu au moins 70Mbps depuis Renater
  - En TCP, limité par pertes du lien Internet
  - Limité par temps CPU pour le chiffrement

Chiffrement

- Paralléliser avec plusieurs démons openvpn
- Dépend algorithme négocié
- Accélération matérielle
  - /proc/cpuinfo : au moins flags ssse3 et aes
  - VM : utiliser `-cpu host` ou `-cpu qemu64,+ssse3,+aes` (attention ganeti)
  - Si possible, `pclmulqdq ssse3 sse4_1 sse4_2 aes avx bmi1 avx2 bmi2`
- Chez FDN,
  - sans ssse3 & aes, saturation 1 cœur ~100Mbps

Limitation bande passante

open.fdn.fr

```
# Virer ce qui existe

tc qdisc del dev $dev root 2> /dev/null || true

# Mettre un htb tout en haut, qui balance les paquets au
  sfq 1:2 et limite quand ils remontent

tc qdisc add dev $dev root handle 1: htb default 2

tc class add dev $dev parent 1: classid 1:1 htb rate $BW

tc class add dev $dev parent 1:1 classid 1:2 htb rate $BW

# Mettre un sfq pour repartir equitablement

tc qdisc add dev $dev parent 1:2 handle 2: sfq perturb 10
```

Discussion avec abonnés

- Un VPN « proprifie » un accès ADSL/fibre classique
  - Plus facilement acceptable
- Adresse IP fixe / dynamique
  - On peut vouloir l'un ou l'autre
- Rappeler qu'un VPN n'anonymise pas
  - Obligation de donner informations, e.g. IP source du client VPN
- Rappeler que nos VPNs n'ont pas de firewall
  - Certains OS présupposent réseau privé

Patcher openvpn

- Version patchée pénible à maintenir
- Bricoler une autre solution pas viable à long terme
- Contribuez à faire intégrer les patchs upstream