

Les techniques de tunnels VPN

Roland Dirlewanger

CNRS - Délégation Aquitaine-Limousin
Esplanade des Arts et Métiers
33402 TALENCE CEDEX

Roland.Dirlewanger@dr15.cnrs.fr

Sommaire

- Généralités
- Trois solutions
 - Tunnels SSH par port dans SSHv3
 - Tunnels Ipsec
 - Tunnels OpenVPN
- Bilan d'utilisation
- Bibliographie

Les enjeux de la mobilité

- Le point de vue de l'utilisateur
 - On peut se raccorder à l'Internet depuis n'importe où
 - Je peux donc accéder aux ressources informatiques de mon unité d'où je veux, quand je veux
- Le point de vue de l'administrateur systèmes et réseaux
 - Je dois protéger ces ressources informatiques
- Comment concilier les deux ?

Les différents cas de figure

- Connexion sur un réseau tiers
- salle libre service d'une conférence
 - visite dans un autre labo
 - réseau WiFi d'hôtel, de gare, ...

Utilisation d'un poste tiers booté par une clé USB, un DVD, etc.



Réseau du labo

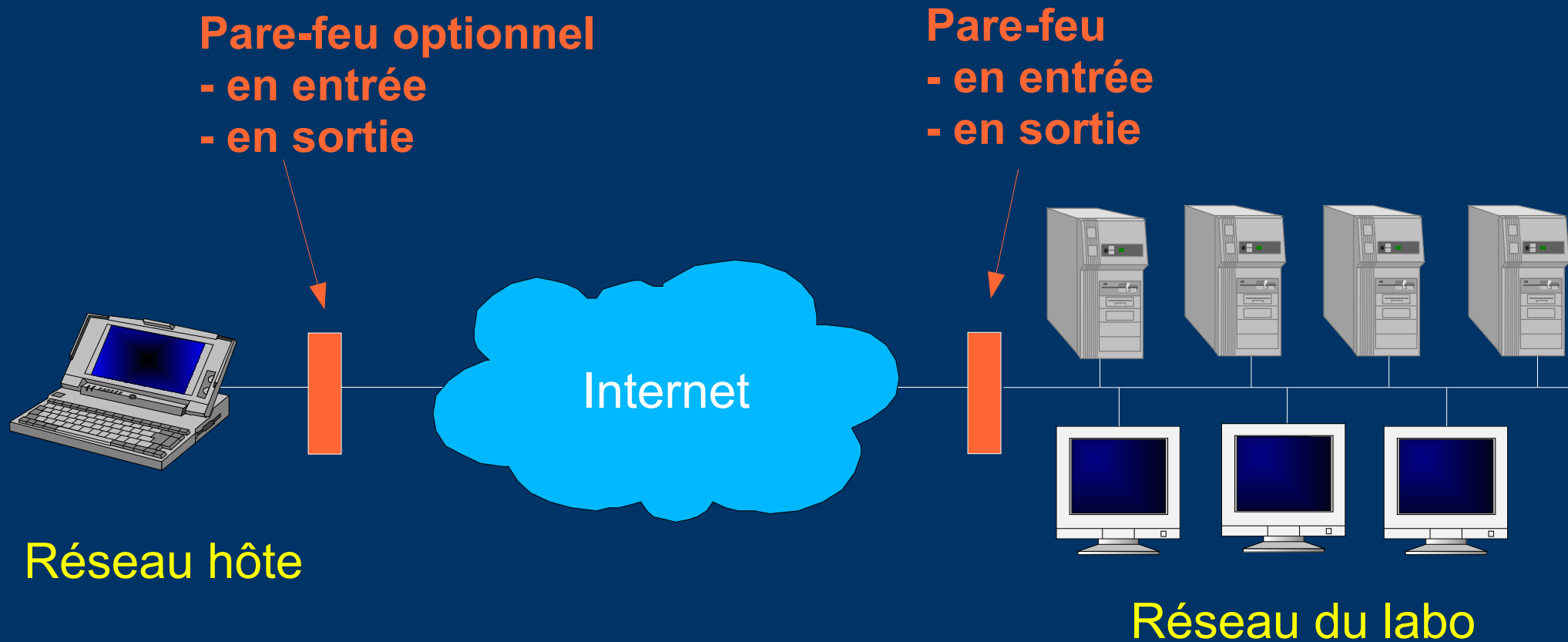
- Connexion à domicile
- ADSL
 - Téléphone

- Utilisation d'un poste tiers
- cybercafé
 - dans un autre labo

- Connexion sans fil
- via opérateur
 - via téléphone portable

Les différents cas de figure

- Ces cas de figure se ramènent à :

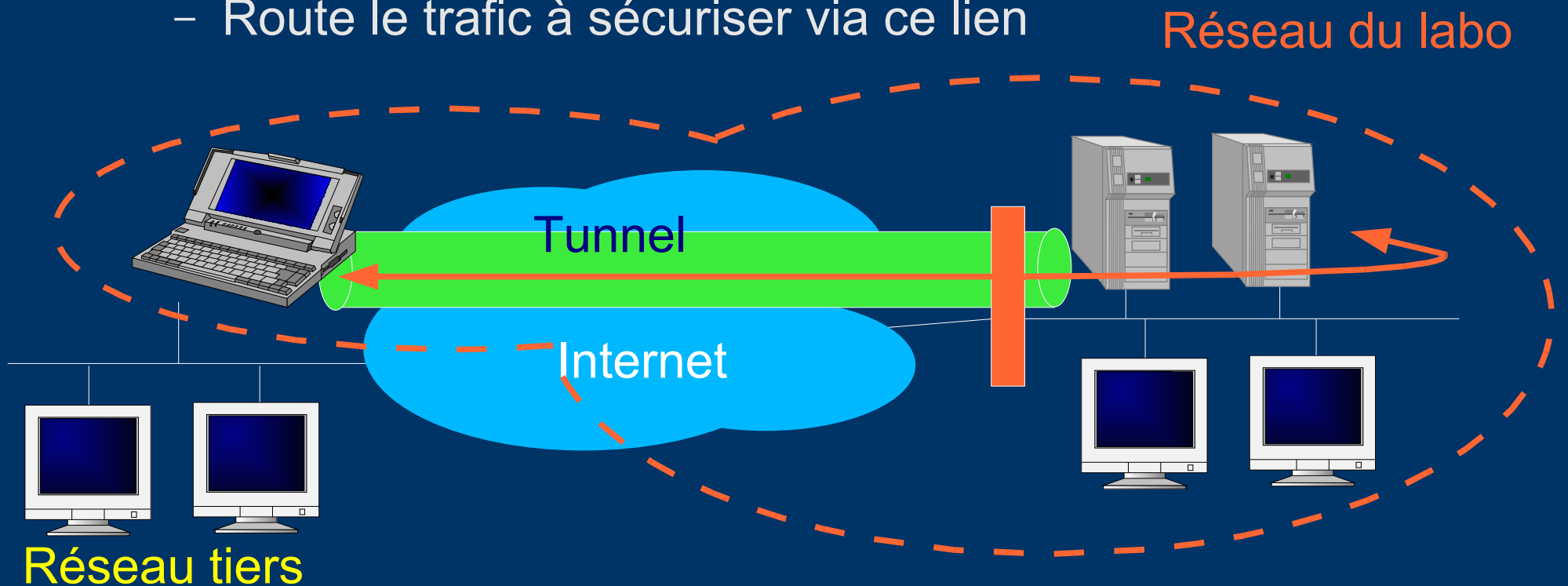


Les différents cas de figure

- Les besoins se ramènent le plus souvent à :
 - Accès à la messagerie : lecture, envoi
 - Accès intranet WWW
 - Accès aux serveurs de fichiers
 - Accès à des bases de données
 - Accès à des applications
 - Accès à une bulle protégée dans un réseau “hostile”
- On peut répondre avec une solution différente pour chaque besoin et chaque situation : c'est complexe
- Un réseau privé virtuel est une réponse homogène et transparente pour l'ensemble

Idée générale

- Mettre en place une solution qui :
 - Crée un lien virtuel point à point entre le poste mobile et un équipement du labo.
 - Route le trafic à sécuriser via ce lien



Qu'est-ce qui doit transiter par le tunnel ?

Deux solutions :

- Uniquement le trafic vers le réseau du labo, le reste est routé normalement
 - Pas de transit inutile via le réseau du labo
 - Possibilité d'utiliser des ressources (imprimantes) du réseau local sur lequel est connecté le poste mobile
- Tout le trafic passe par le tunnel
 - Certaines solutions (Checkpoint, Cisco) rejettent tout trafic entrant qui ne provient pas du tunnel
 - Le poste mobile profite de la protection du FW du labo
 - Le poste mobile peut accéder à des services authentifiés par adresse IP

Les réseaux privés virtuels

- VPN = Virtual Private Network
- Technologie réseau permettant de construire un réseau privé à l'intérieur d'une infrastructure publique.
 - Privé : les échanges transitant par ce réseau sont confidentiels pour les autres utilisateurs du réseau public.
 - Virtuel : le réseau privé ainsi créé n'est pas matérialisé par des liens physiques.

Le problème de TCP dans TCP

- Solution naturelle :
 - Bâtir un tunnel avec un protocole sécurisé (SSH, SSL) qui s'appuie sur une connexion TCP
- TCP dans TCP :
 - Les connexions vers des services (HTTP, SMTP, IMAP) utilisent elles aussi des connexions TCP qui vont passer dans le tunnel. On parle de TCP dans TCP.
 - Sur des réseaux non saturés, tout se passe bien

Le problème de TCP dans TCP

- Le problème :
 - La connexion TCP du tunnel peut détecter des pertes de paquets, des congestions, etc.
 - Elle ralentit le trafic en rallongeant les délais de retransmission.
 - Si une connexion TCP (exemple, l'envoi d'un message via SMTP) est demandée à travers le tunnel, elle démarre avec des délais de retransmission courts. Elle n'a aucun ACK dans ces délais.
 - Elle génère des demandes de retransmission à un rythme plus rapide que la connexion TCP du tunnel peut absorber
 - Le résultat est que la connexion TCP du tunnel est noyée de demandes de retransmission. La part de trafic utile diminue fortement et donne l'impression que le tunnel est coupé.

Tunnels SSH v3

- Le protocole SSH
 - mis au point par Tatu Ylönen qui fonde une société du même nom en 1995
 - but : remplacer par des équivalents plus sûrs les commandes telnet, ftp, rlogin, rcp, etc.
 - authentification par login/mot de passe ou clé privée
clé, publique
 - confidentialité (chiffrement), intégrité (HMAC)
 - transport du protocole X11 (déport d'affichage sur Unix)
 - tunnels par ports

SSH – Authentification du serveur

- Chaque serveur doit disposer d'un bi-clé
- Client et serveur échangent un secret (via Diffie-Hellmann) dont sont dérivées les clés de session
- Le serveur transmet sa clé publique et signe le dialogue initial
- Le client doit vérifier le lien entre la clé publique et le serveur
 - utilisation d'une base locale
 - utilisation de certificats
 - échec de la vérification => avertir l'utilisateur

SSH – Authentification du client

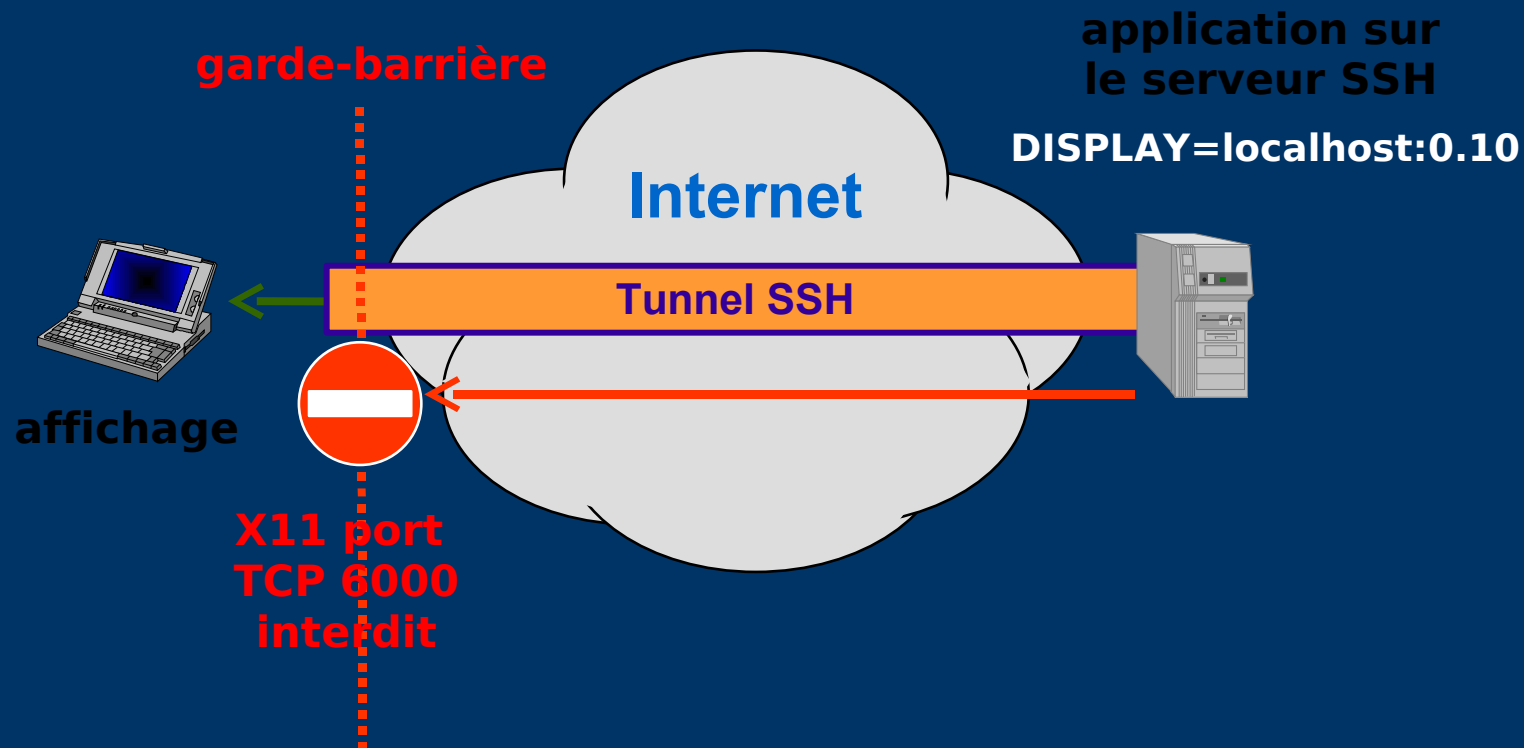
- L'authentification du client
 - Plusieurs méthodes : mot de passe, clé publique, nom de machine, Kerberos (1.5)
 - Par mot de passe :
 - le plus simple à mettre en œuvre
 - remplacement transparent de telnet, ftp, etc. par ssh, sftp, etc.
 - Par clé publique :
 - action explicite de l'utilisateur pour rajouter sa clé publique dans la configuration de SSH

Tunnels SSH

- X11 forwarding
 - définition d'un serveur X11 virtuel sur le serveur distant :
DISPLAY=localhost:0.10
 - transmission via la connexion SSH de toute requête sur ce serveur X11 vers le serveur X11 de la machine locale
- Port forwarding
 - association entre un port local (IP 127.0.0.1 port tcp/x) et une destination quelconque (A.B.C.D port tcp/y)
 - toute connexion sur le port x local entraîne une connexion du serveur distant vers A.B.C.D port y
 - transmission via la connexion SSH du trafic 127.0.0.1:x vers A.B.C.D:y

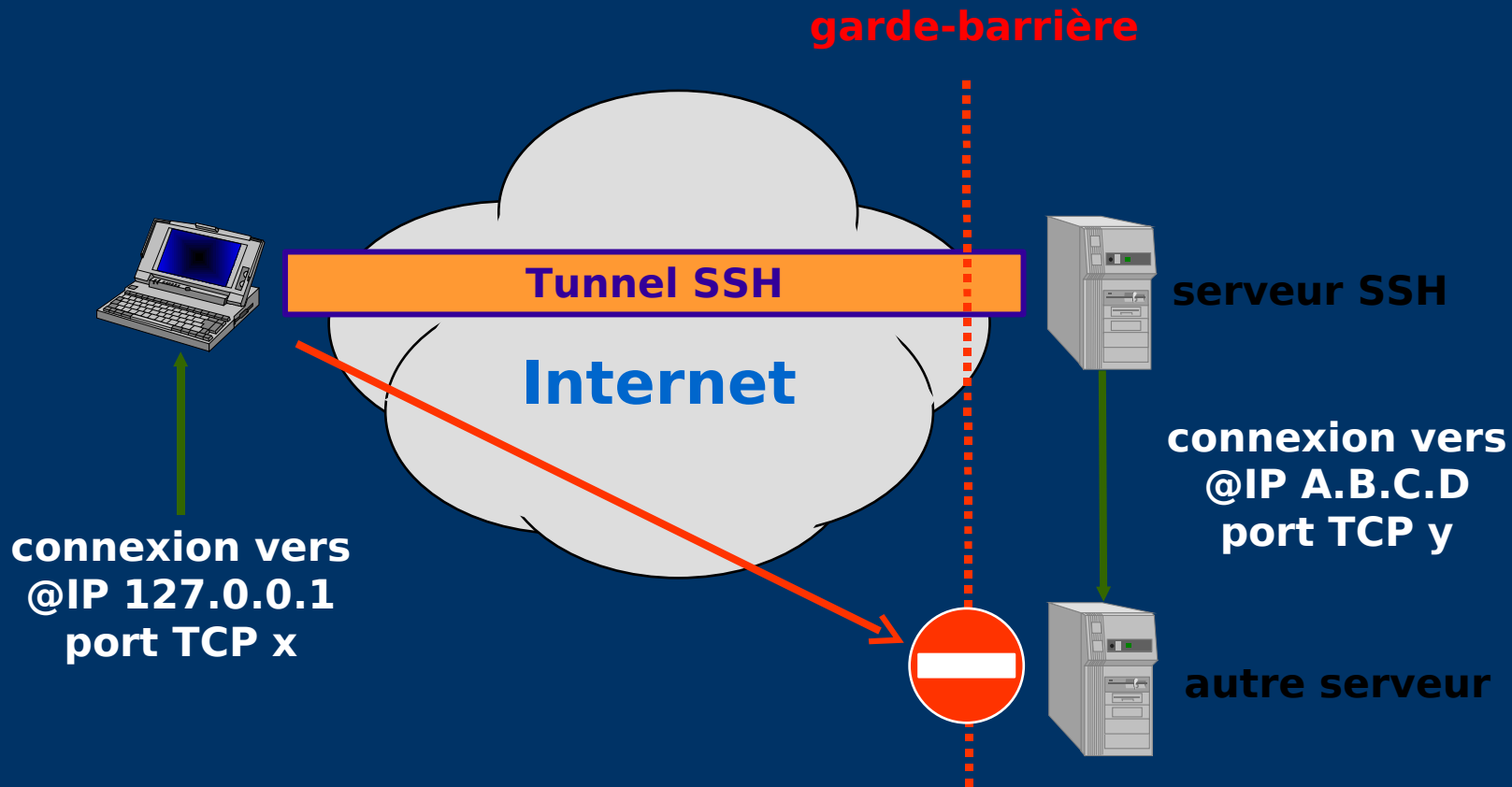
SSH – Tunnels par ports

- X11 forwarding : déport d'affichage à travers SSH



SSH – Tunnels par ports

Port forwarding



SSH – Principales implémentations

- Serveurs
 - Version commerciale : www.ssh.fi
 - Version libre : www.openssh.org
 - mais aussi sur des équipements réseaux (cisco, ...)
 -
- Clients
 - openssh
 - putty

SSH – Bilan d'utilisation

- Avantages :
 - Très simple à mettre en oeuvre
 - Grande interopérabilité
 - Mais ... il faut faire l'effort d'imposer une authentification par clé publique
- Inconvénients :
 - Ne fonctionne que pour une liste de ports connue à l'avance
 - Ne fonctionne pas pour le partage de fichiers, le bureau à distance, les serveurs HTTP virtuels, etc.
 - Pas vraiment transparent pour l'utilisateur

Tunnels IPsec

- IPsec = IP Security Protocol.
 - Couche sécurité développée par l'IETF.
- Ensemble de mécanismes destinés à protéger le trafic au niveau IP.
 - Transparence pour les couches supérieures
- Un système conforme à IPsec peut :
 - Choisir les protocoles de sécurité.
 - Choisir les algorithmes utilisés.
 - Utiliser des clefs cryptographiques/certificats

Les principaux composants d'IPsec

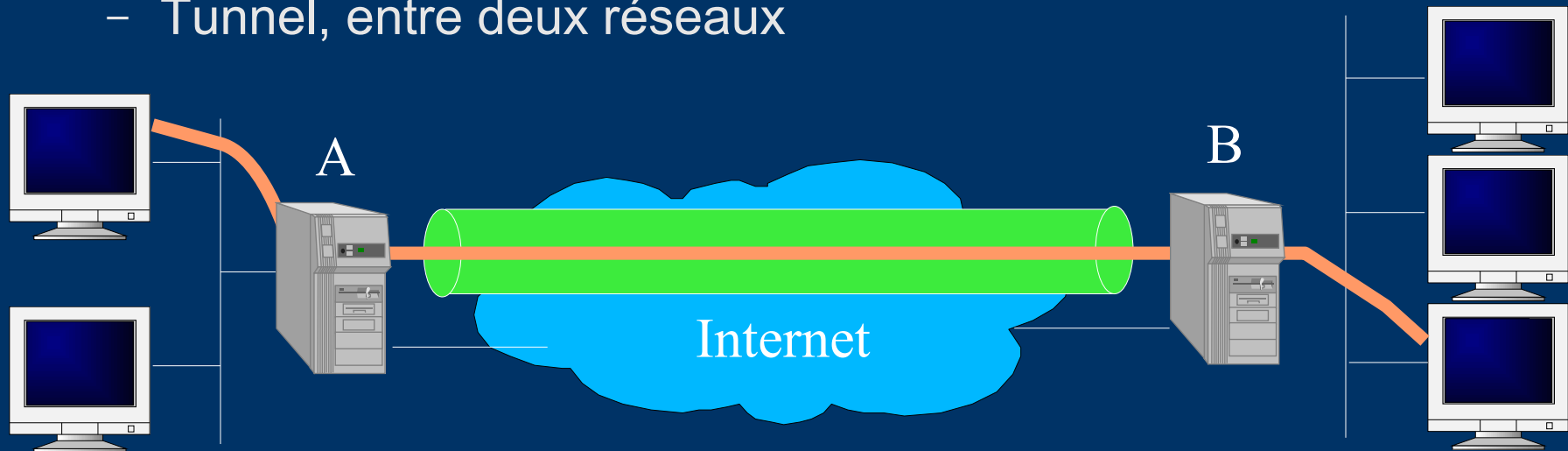
- ISAKMP, UDP port 500
 - Négociation des protocoles de sécurité
 - Echanges de clés
 - Authentification par PSK, certificats, bi-clés
- ESP = Encapsulated Security Payload, protocole 50
 - Chiffrement
- AH = Authentication Header, protocole 51
 - Authentification, intégrité
- Politique de sécurité
 - Permet d'indiquer quel trafic sécuriser ou pas en fonction des adresses et des numéros de ports.

IPsec – AH et ESP

- Deux modes d'utilisation :
 - Transport, entre deux machines



- Tunnel, entre deux réseaux



IPsec - Principales implémentations

- Linux, NetBSD, FreeBSD :
 - Natif + ipsec-tools (racoon, implémente IKE)
 - FreeSwan pour les noyaux Linux 2.4
- Windows
 - Natif
- Cisco

IPsec et NAT

- IPsec n'est pas nativement prévu pour fonctionner en environnement NAT-té.
- Protocole AH :
 - Impossibilité car il authentifie des champs modifiés par du NAT (adresses sources et destination).
- Protocole ESP :
 - Cache les ports TCP/UDP utilisés pour multiplexer/démultiplexer les connexions NAT-tées.
- Support du NAT décrit dans 2 RFC:
 - RFC 3948 : “UDP Encapsulation of ESP IPsec Packets”.
 - RFC 3947 : “Negotiation of NAT-Traversal in the IKE”.

IPsec et NAT

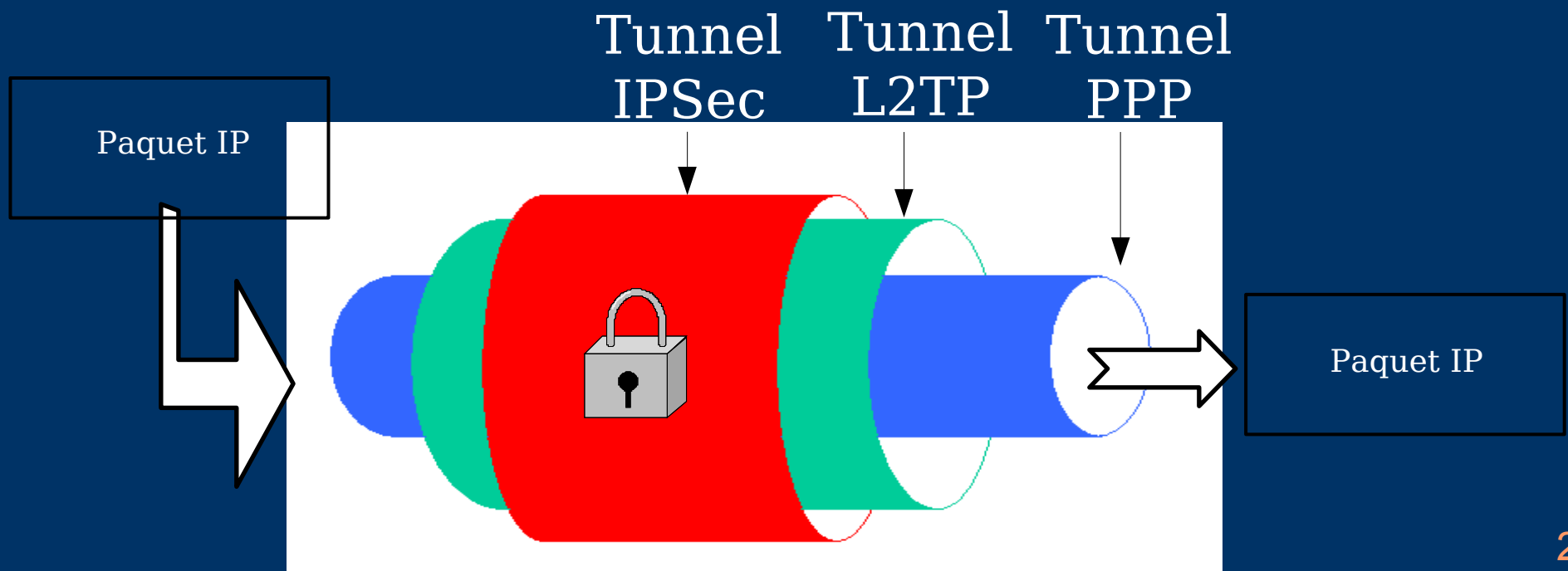
- Principes du NAT-Traversal :
 - Encapsulation de la charge ESP dans un datagramme UDP.
 - Ajout au protocole ISAKMP pour :
 - Détecter les systèmes NAT sur le chemin réseau entre les 2 tiers en présence.
 - négocier l'accord sur la norme de NAT-Traversal utilisé.
 - Utilise le port 4500/udp (ISAKMP NAT).

IPsec – Le modèle Windows

- But :
 - D'abord, authentifier la machine via un certificat
 - Puis, authentifier l'utilisateur via ses paramètres de connexion Windows
- Utilise PPP
 - Authentification de l'utilisateur
 - Création d'interfaces virtuelles
- Utilise L2TP
 - PPP s'appuie sur la couche liaison (niveau 2)
 - L2TP offre un niveau 2 au dessus d'IP

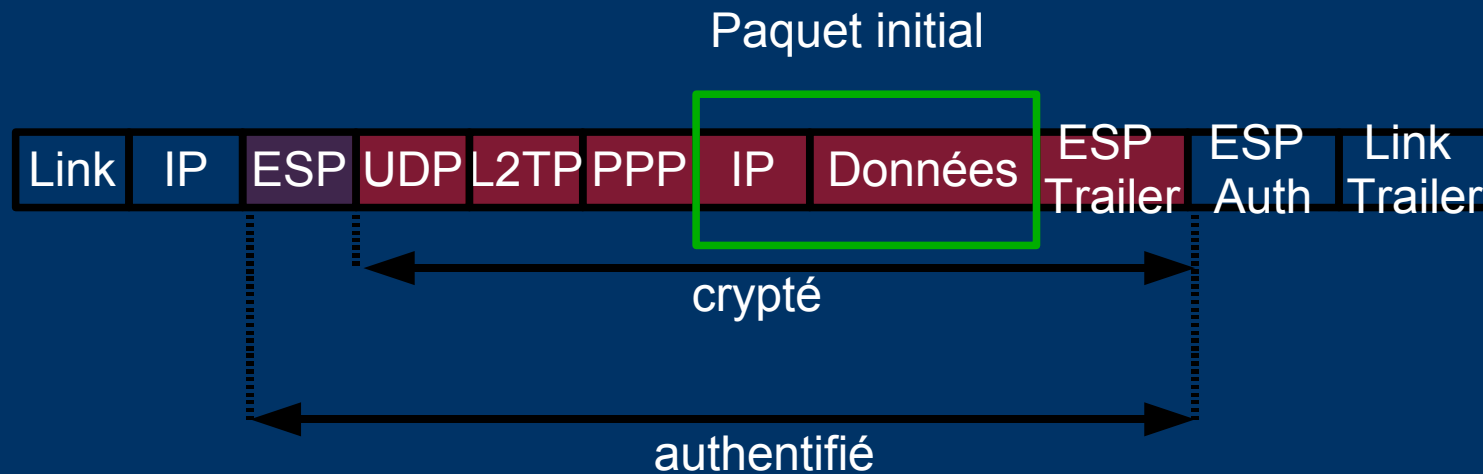
IPsec – Le modèle Windows

- Imbrication des tunnels IPsec, L2TP et PPP.
 - IPsec = protection des échanges (chiffrement).
 - L2TP = transport des trames PPP.
 - PPP = Authentification, réadressage de la machine cliente, transport du trafic VPN.



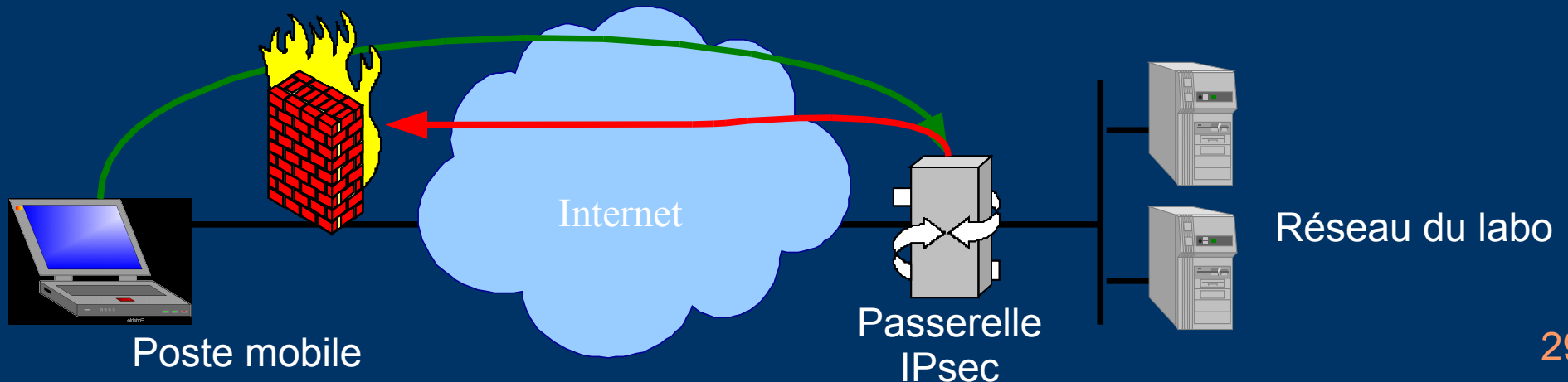
IPsec – Le modèle Windows

- Imbrication des tunnels IPsec, L2TP et PPP.



IPsec – Bilan d'utilisation

- Complexité du modèle Windows
- Problème de NAT
- Problèmes d'accès :
 - Les paquets UDP 500, 4500, ESP, AH sont souvent arrêtés par le FW du réseau qui héberge le poste mobile



Les tunnels SSL

- Au départ, SSL (*Secure Socket Layer*) est destiné à sécuriser les connexions TCP entre un client et un serveur. Exemples :
 - HTTP / SSL entre un navigateur et un serveur WWW
 - IMAP / SSL entre un client et un serveur de messagerie
 - SMTP / TLS entre deux serveurs de messagerie
- Stunnel (www.stunnel.org) :
 - Permet d'utiliser SSL lorsque l'application sur le client et/ou le serveur ne sait pas faire du SSL
 - Sécuriser des accès à des bases de données
 - Sécuriser rsync, LDAP, VNC, ...

Tunnels SSL

- Autre idée : utiliser SSL en lieu et place d'IPsec pour faire des VPN
- Exemple : OpenVPN (www.openvpn.org)
 - Produit OpenSource, licence GPL
 - Utilise les interfaces virtuelles TUN/TAP
 - Encapsule le trafic dans une connexion SSL
 - Utilise UDP ou TCP (port paramétrable = 1194 par défaut)

OpenVPN

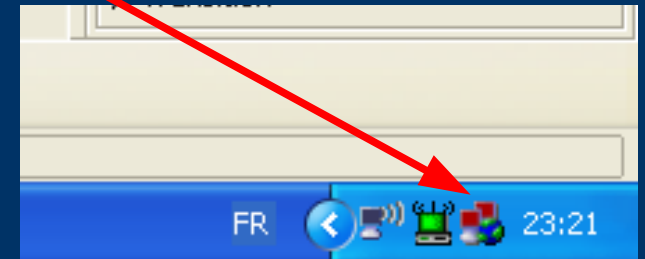
- Deux modes de fonctionnement :
 - Mode tunnel : niveau 3 (routage)
 - Le client obtient une adresse IP de la part du serveur OpenVPN. Cette adresse est dans un réseau privé.
 - Le trafic entre le client, le serveur, le réseau de l'unité est routé via le tunnel
 - Mode pont : niveau 2
 - Le client distant est vu comme étant directement connecté au réseau local de l'unité
 - Obtient une adresse IP de la même façon que les postes directement connectés au réseau local (DHCP, statique)
 - Les broadcasts sont acheminés dans le tunnel

OpenVPN - Authentification du client

- Par certificats
 - Accepte des certificats CNRS serveurs ou utilisateurs
 - Fait appel à un script (verify-dn) qui permet de programmer des droits d'accès aussi fins que l'on veut
- Par login / mot de passe
- Sur Linux, par plugin PAM

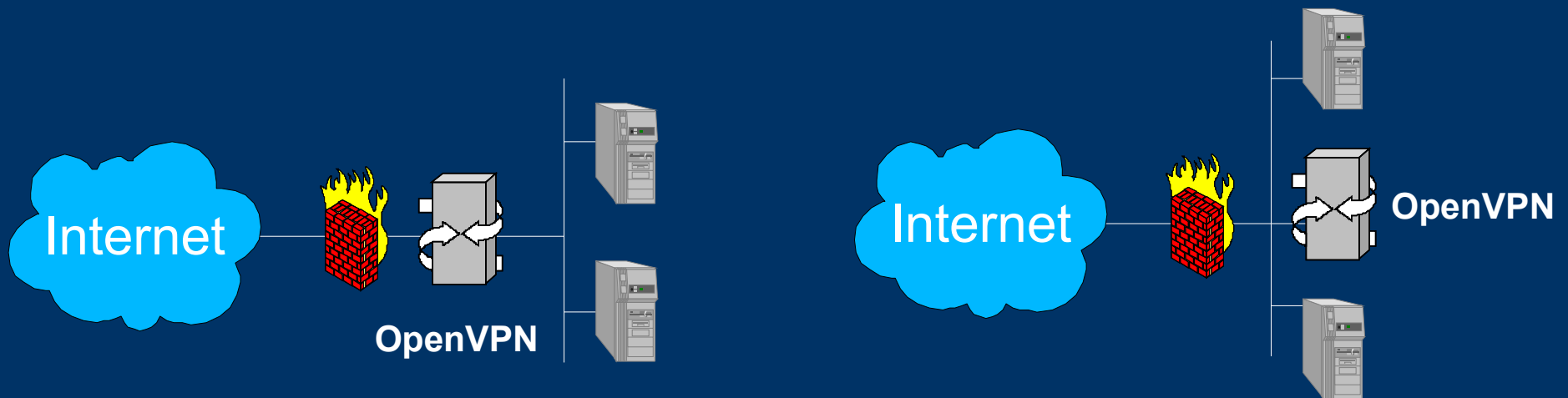
OpenVPN - Configuration, utilisation

- Utilise des fichiers de configuration :
 - Syntaxe identique pour client et pour serveur
 - Syntaxe indépendante de la plateforme (Linux, MacOS, Windows)
 - Configuration simple (~ 12 lignes) et bien documentée
- Une GUI de lancement pour client Windows
 - Via bouton dans le coin inférieur droit
 - Téléchargeable sur
 - openvpn.se



OpenVPN – trafic du tunnel ... dans le tunnel

- OpenVPN suppose que la passerelle VPN dispose d'une adresse externe et d'une adresse dans le réseau du labo. En pratique, elle n'a qu'une interface, dans le réseau interne.



Architecture avec une passerelle OpenVPN à deux interfaces

Architecture avec une passerelle OpenVPN à une interface

OpenVPN – trafic du tunnel ... dans le tunnel

- Bug (?) : OpenVPN en mode tunnel sur une passerelle à une interface (@IP a.b.c.d) établit le tunnel vers cette interface puis configure les tables de routage pour que tout le trafic à destination du réseau interne, y compris a.b.c.d, passe par le tunnel.
- Deux possibilités pour contourner le problème :
 - Script sur le client pour mettre une route spécifique de a.b.c.d vers le routeur par défaut du réseau qui héberge le client
 - Configurer OpenVPN sur le serveur pour qu'il « pousse » vers le client des routes individuelles pour chaque serveur du réseau interne.

OpenVPN - Bilan d'utilisation

- Avantages
 - Une seule syntaxe client/serveurs pour toutes plateformes
 - Simplicité de la mise en oeuvre
 - Technologies maîtrisées (SSL, certificats)
- Inconvénients
 - Mode tunnel : trafic du tunnel dans le tunnel
 - Mode pont : difficilement utilisable sur liaison lentes à cause des broadcasts

Bilan général d'utilisation

- Un éventail de solutions robustes aux ergonomies accessibles à tout utilisateur
- L'environnement du réseau hôte peut empêcher une solution de fonctionner :
 - Blocage du trafic UDP 500, ESP, AH : IPsec
 - Adresses privées + NAT : IPsec
 - Blocage de UDP 1194 : OpenVPN/UDP
 - Adresses privées + proxy HTTP : SSH, OpenVPN/TCP
- En général SSH et OpenVPN en mode tunnel sur TCP passent toujours

Bibliographie

- « Tutoriel VPN - Protocoles et fonctionnement des réseaux privés virtuels », B. Dexheimer, R. Dirlewanger, F. Morris, JRES, Novembre 2003, Lille.
<http://2003.jres.org/TUTORIELS/paper.A.pdf>
- « La mobilité : quelles solutions ? », R. Dirlewanger, Séminaire RAISIN, mars 2005, Bordeaux.
http://raisin.u-bordeaux.fr/article.php3?id_article=32
- « Why TCP Over TCP Is A Bad Idea », O. Titz, avril 2001.
<http://sites.inka.de/~W1011/devel/tcp-tcp.html>