



# Méthodes d'authentification avec un serveur Radius

Serge Bordères  
(Centre d'Etudes Nucléaires de Bordeaux-Gradignan)

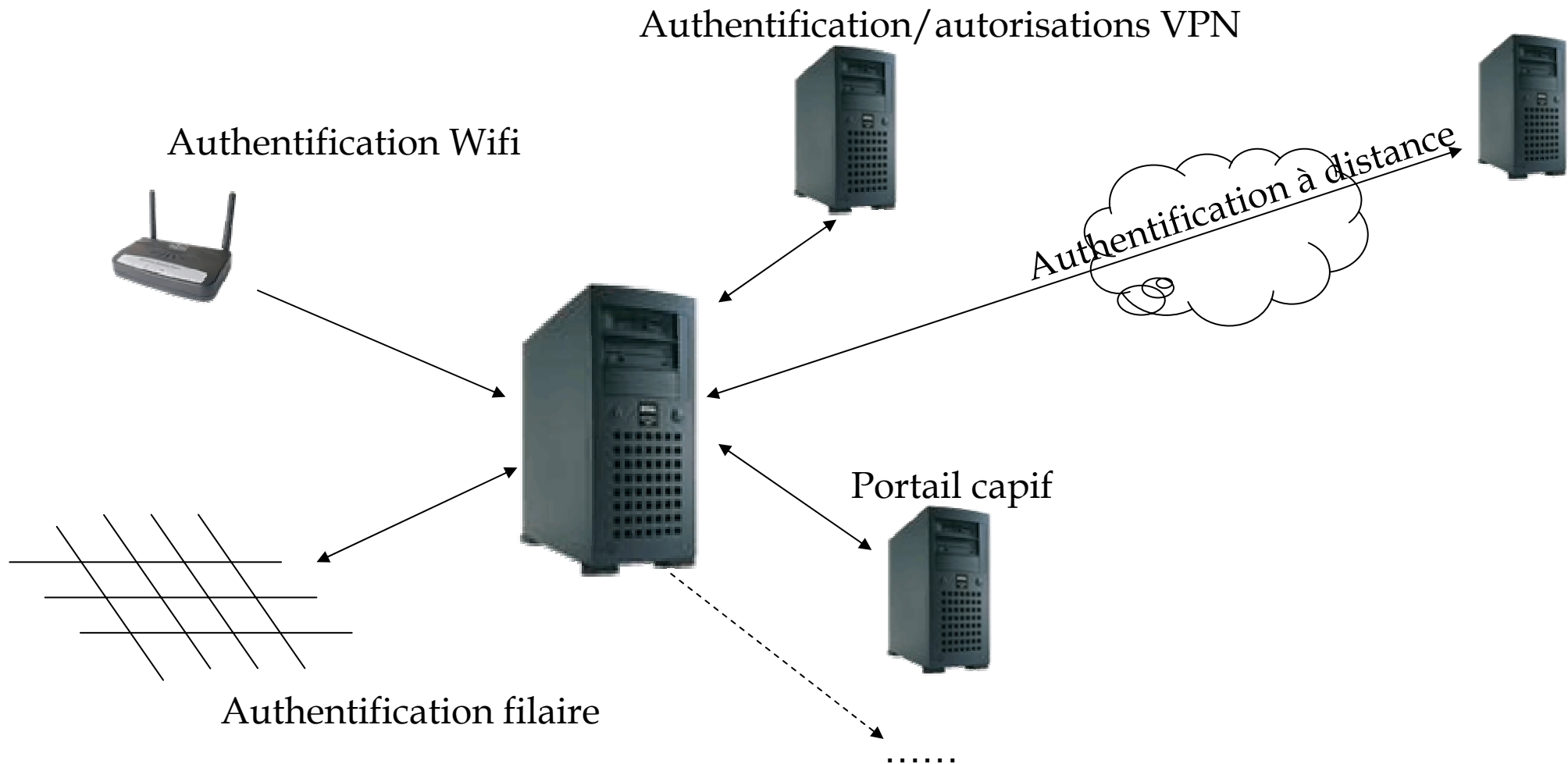
20 mars 2007

Institut d'Astrophysique de Paris

# Sommaire

- Radius, principes
- Radius et 802.1X
- Usages de Radius
  - ✓ Protocoles d'authentification
  - ✓ Réseaux virtuels
  - ✓ Programmation
  - ✓ Portails captifs
  - ✓ Transformer un serveur en client Radius

# Radius - pour faire quoi ?



# Qu'est-ce-que Radius ?

Protocole d' **A**uthentication

=> Qui parle ?

d' **A**utorisation

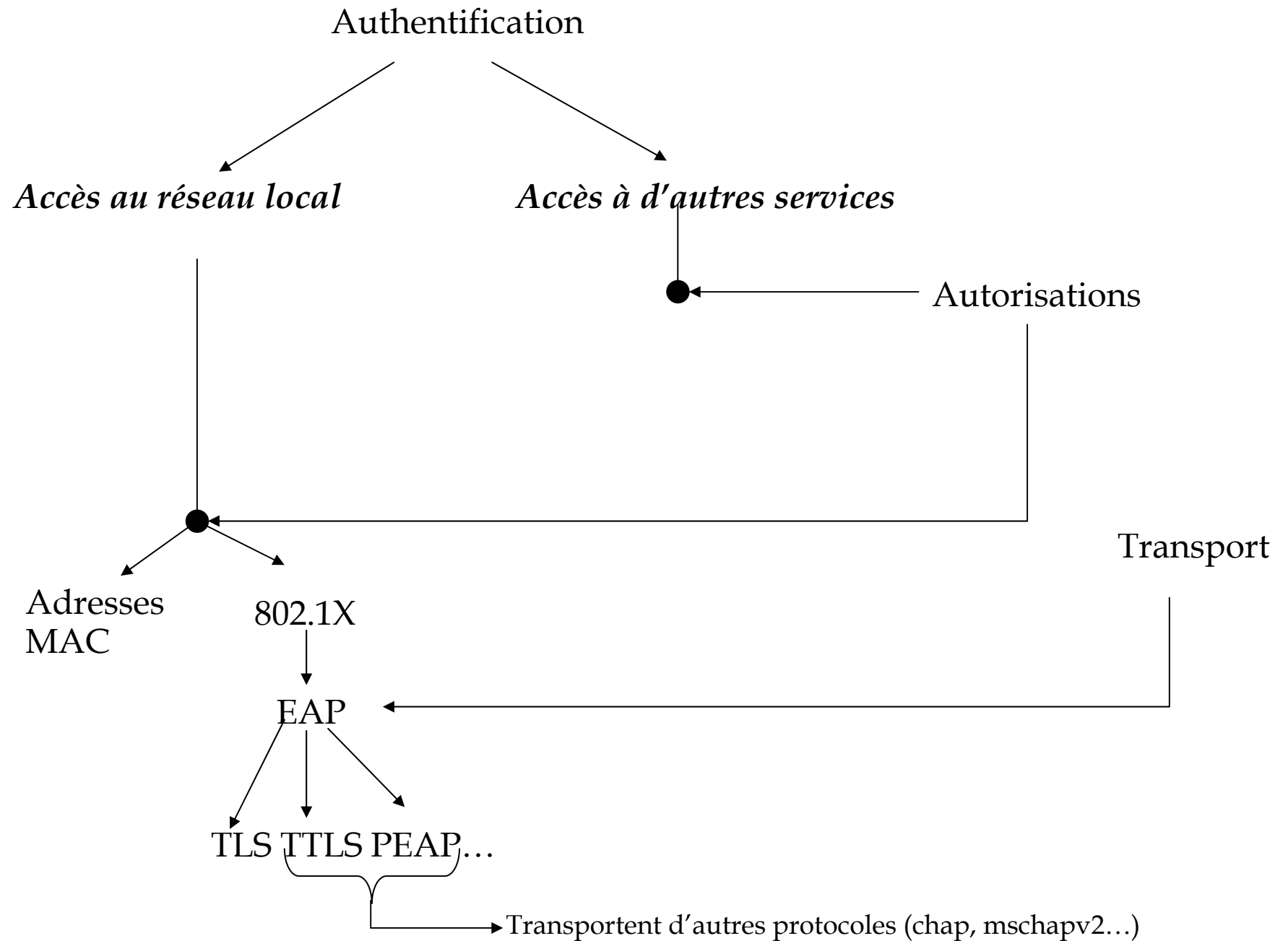
=> Quels sont ses « droits » ?

d' **A**ccounting

=> Que fait-il ?

**Principe client/serveur**

# Le monde de Radius



# Qu'est qu'une authentification ?

C'est le processus qui prouve qu'une identité appartient bien à celui qui l'a présente

- Un identifiant est proposé au serveur radius.
- Il doit vérifier qu'il est bien présent dans sa base
- Il doit vérifier que celui qui présente cet identifiant peut prouver qu'il en est bien le propriétaire

Méthodes les plus courantes :

- ✓ Adresse MAC (faible, pas de preuve)
- ✓ Login/mot de passe
- ✓ Certificat

# Qu'est-ce qu'une autorisation ?

Le terme « autorisation » a un sens très large

- Accès ou refus de la connexion au réseau
- Affecter un N° de VLAN
- Donner une adresse IP

Ou encore :

- Positionner des ACLs
- Exécuter une commande (filtrage, routage...)

Il faut voir ces autorisations comme des attributs de connexion (Reply-item)

# Les clients Radius

Serveur Radius

*Accès au réseau local*

*Accès à d'autres services*

Equipements réseau

Serveurs

Clients Radius

Commutateurs

Routeurs

Bornes sans-fil

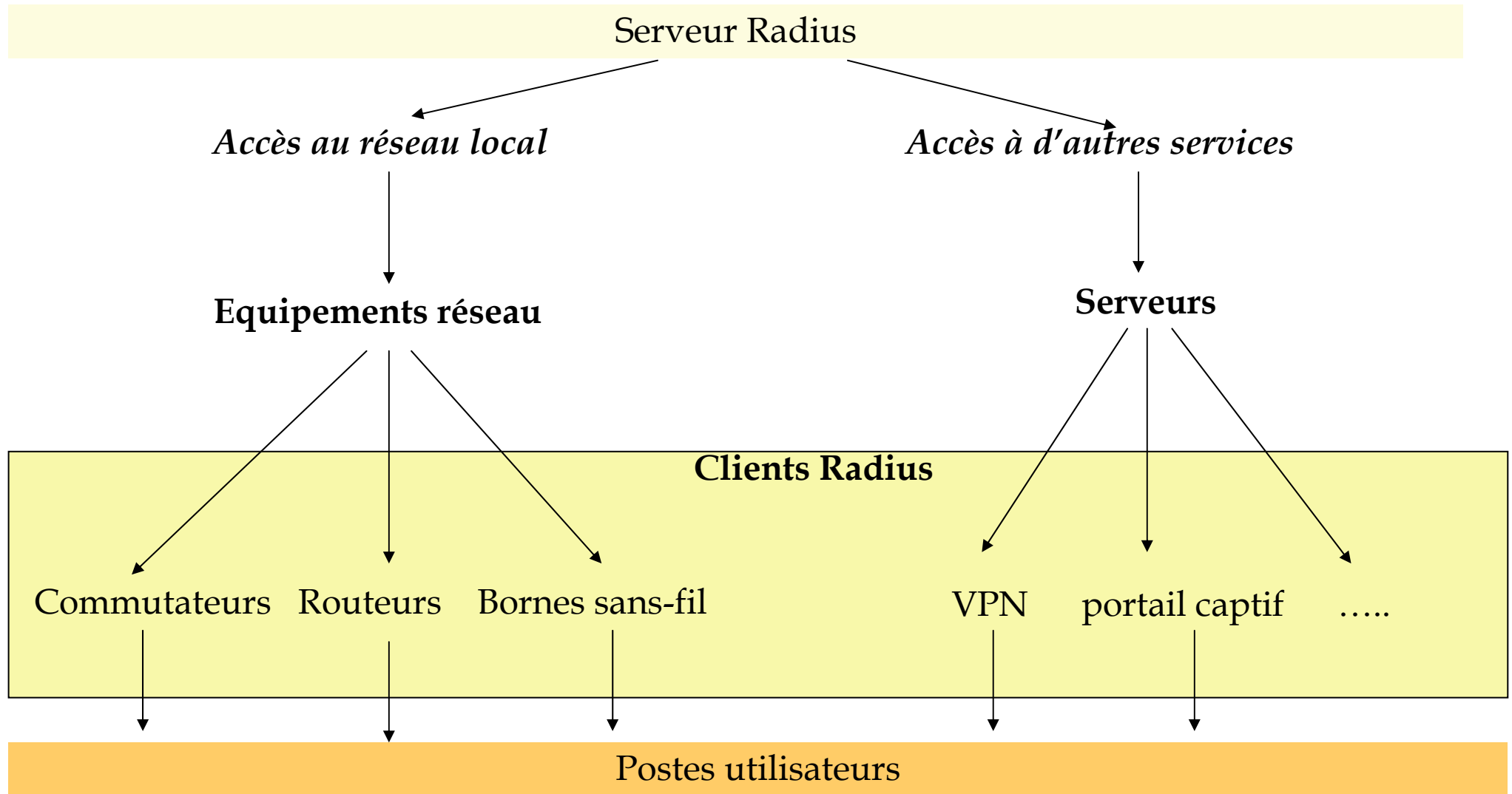
VPN

portail captif

.....

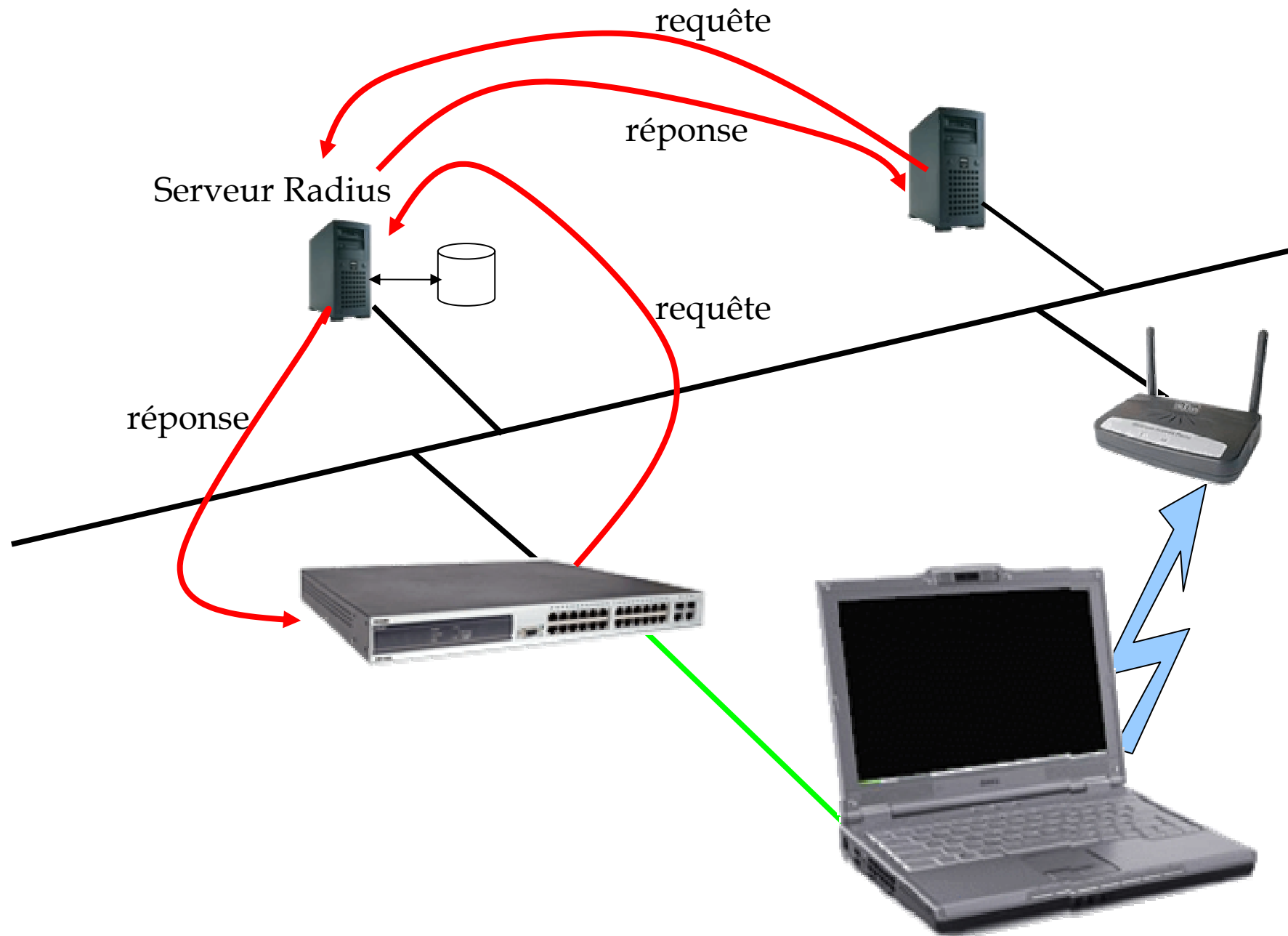
Postes utilisateurs

Pour être client Radius il faut partager un secret partagé

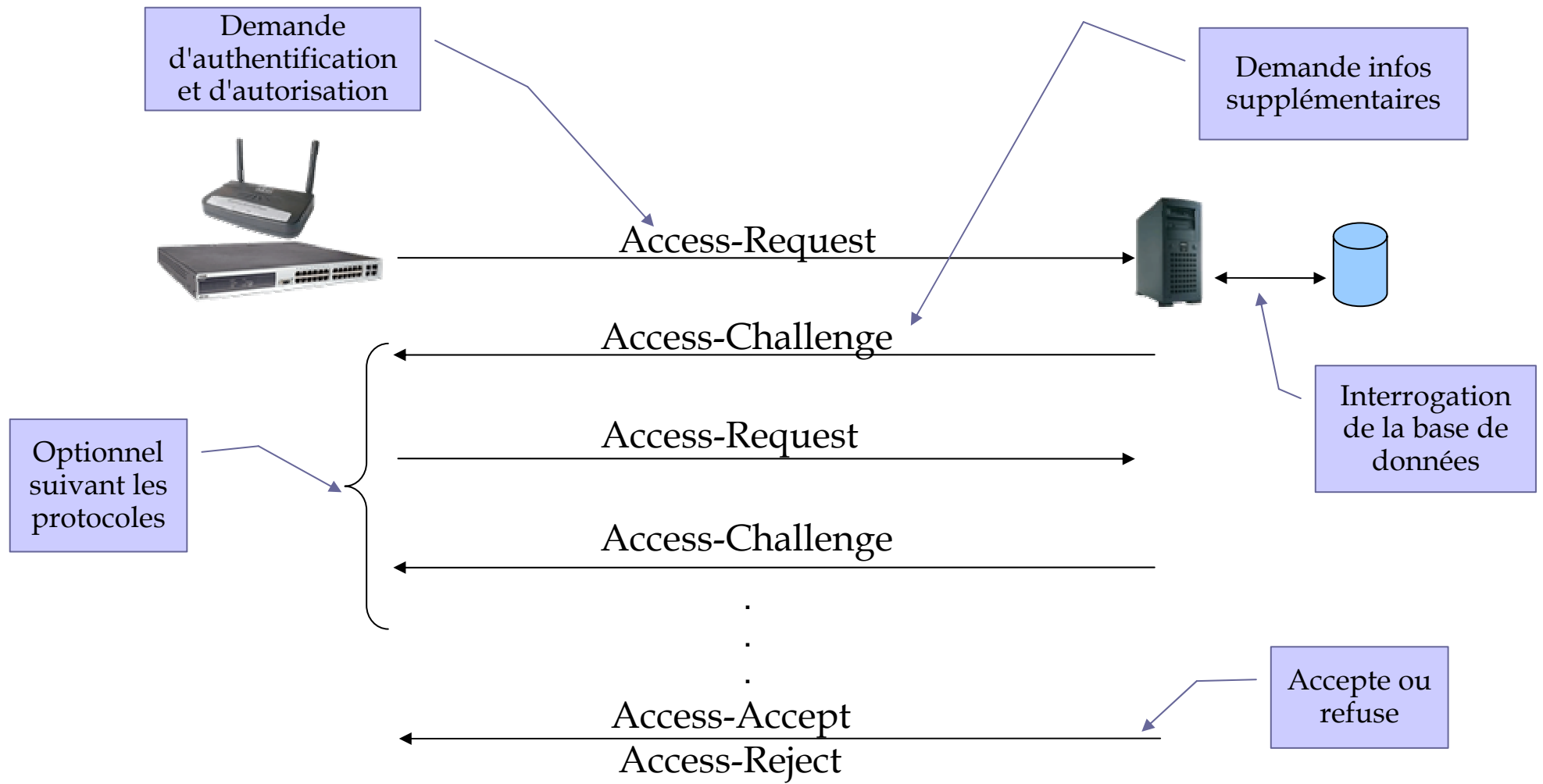




# Principes du protocole Radius



# Principes du protocole Radius



# Principes du protocole Radius : Les attributs

- Toutes les informations échangées entre le serveur Radius et le client Radius passent par des attributs.

Un attribut = Un nom et une valeur

- Certains attributs sont utilisables, d'autres non ou bien dans certains cas

- Exemples d'attributs

- ✓ User-Name
- ✓ Calling-Station-Id
- ✓ Called-Station-Id



Envoyés par les clients au serveur Radius  
(Request-Items)

- ✓ Tunnel-Private-Group-Id
- ✓ Framed-IP-Adress

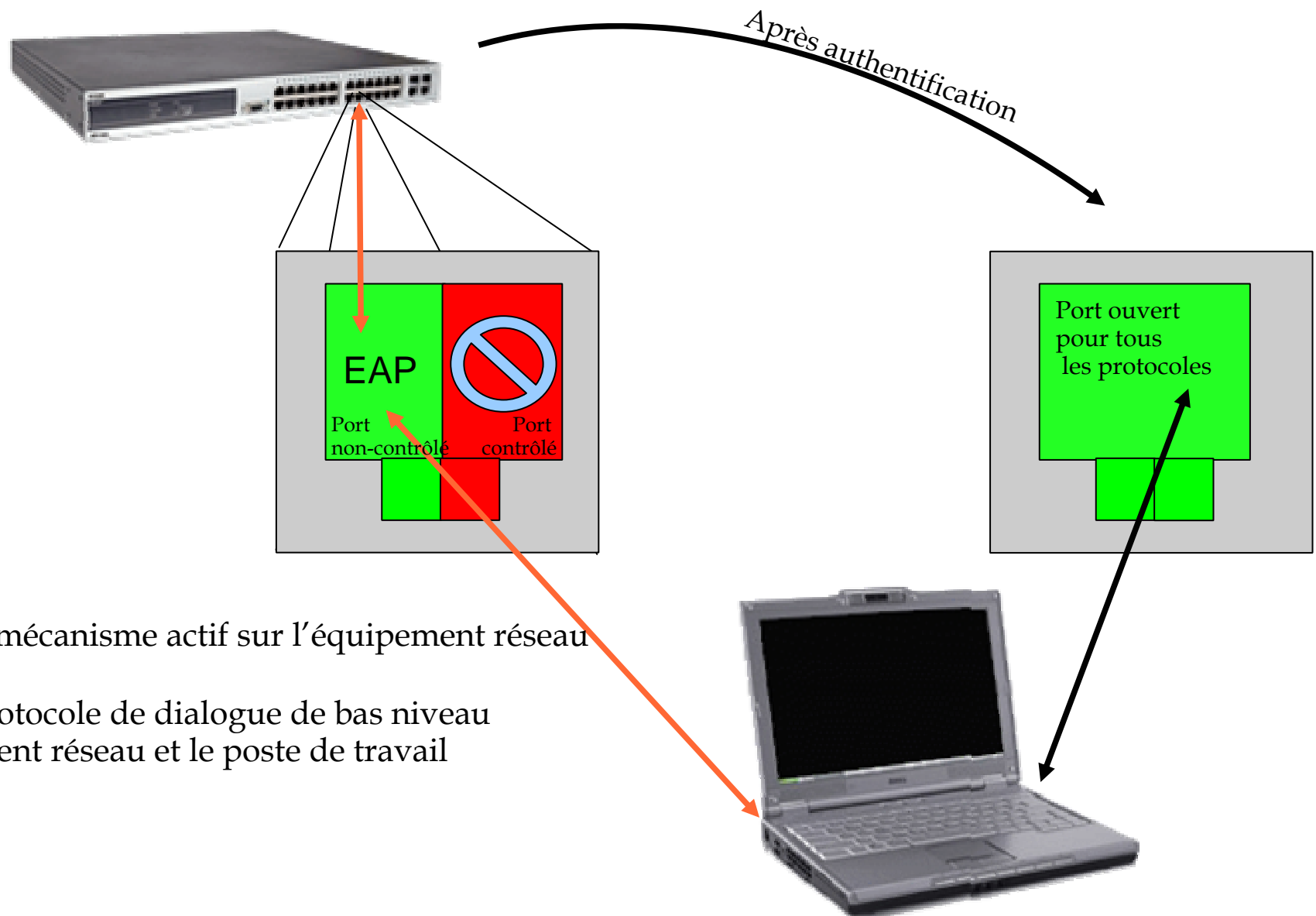


Envoyés par le serveur Radius aux clients  
(Reply-Items)

↘  
Pas compatible avec EAP

- Attributs vendor-specific

# Principes du protocole 802.1X



- 802.1X est un mécanisme actif sur l'équipement réseau
- EAP est un protocole de dialogue de bas niveau entre l'équipement réseau et le poste de travail

# Le protocole EAP

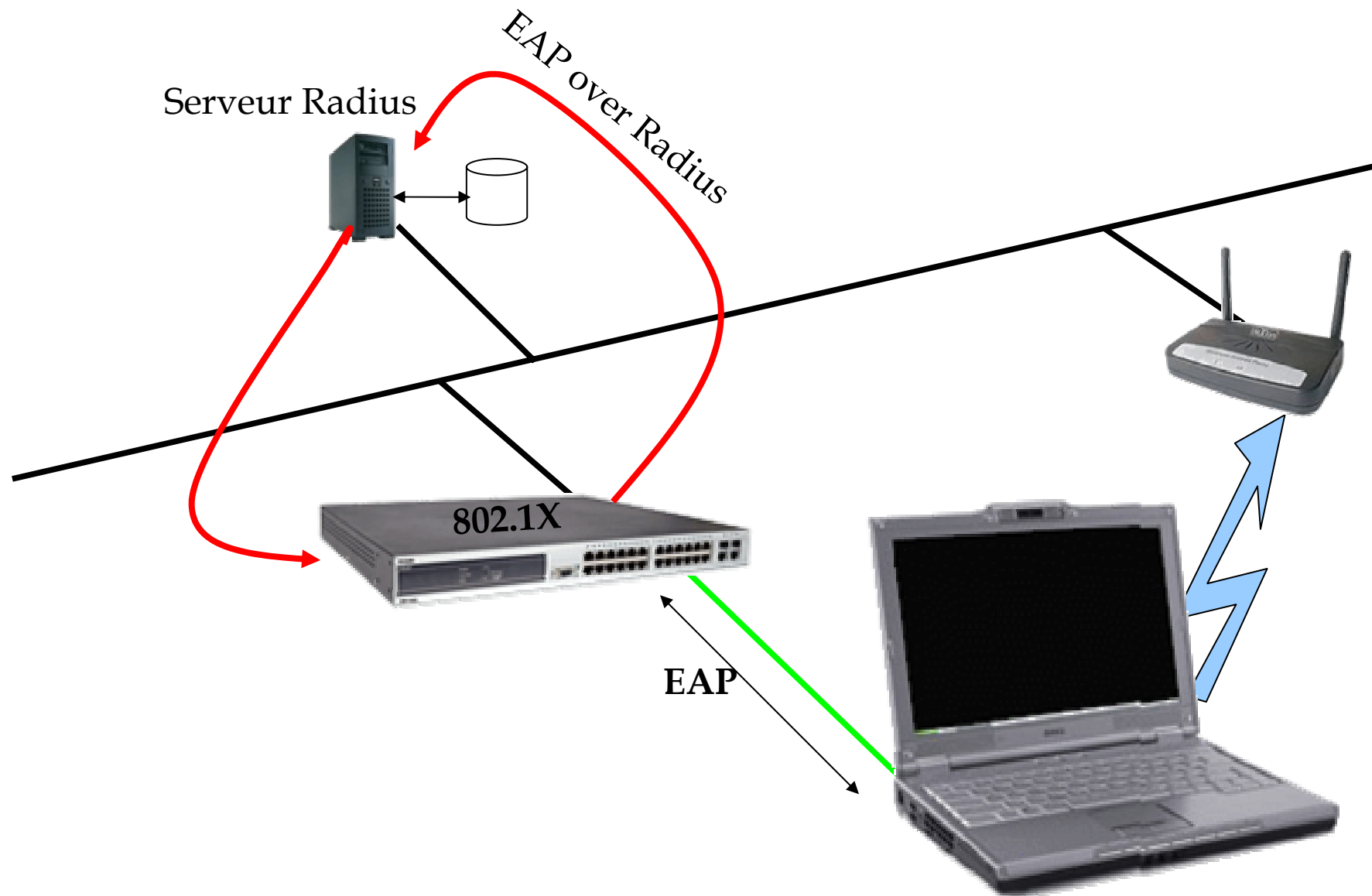
- EAP (Extensible Authentication Protocol) n'est pas un protocole d'authentification
- C'est un protocole de transport de protocole d'authentification (TLS, PEAP, TTLS...)
- Les paquets du protocole d'authentification sont encapsulés dans des paquets EAP
- EAP dispose de quatre types de paquets
  - ✓Request
  - ✓Response
  - ✓Success
  - ✓Failure

L'équipement réseau :

- Connaît le protocole EAP et c'est tout (il ne sait pas ce que transporte EAP)
- Redirige les paquets EAP vers un serveur d'authentification grâce au protocole Radius

Les équipements réseau sont indépendants du protocole d'authentification utilisé

# 802.1X, EAP et Radius



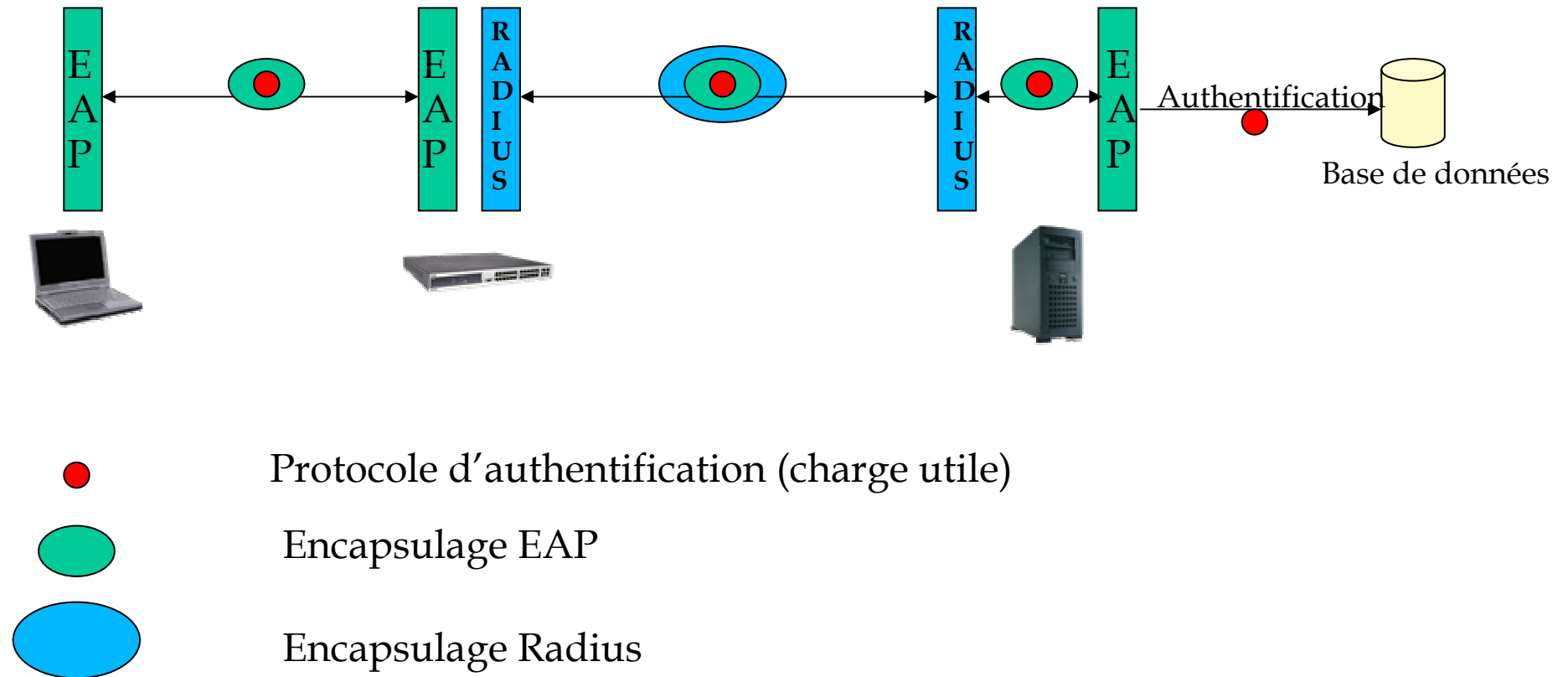
# Compatibilité EAP dans Radius

La compatibilité EAP dans Radius est réalisée au moyen d'un attribut supplémentaire:

## EAP-Message

- Lorsque l'équipement réseau reçoit un paquet EAP du poste utilisateur, il le copie dans un attribut EAP-Message, lui même copié dans un paquet Access-Request.
- Lorsque le serveur Radius reçoit ce paquet il extrait le contenu de EAP-Message et le passe à un module EAP pour dérouler le protocole qu'il contient.
- Ce qui suppose que le serveur Radius dispose de ce module EAP (qui ne fait pas partie du protocole Radius)

# Compatibilité EAP dans Radius





# Bases de données


- La base de données associée au serveur Radius contient des informations d'authentification et/ou d'autorisations.
- Cette base n'est pas spécifiée par le protocole Radius

Avec FreeRadius par exemple :

- |                                     |  |
|-------------------------------------|--|
| ■ Fichier plat (users)              | ➔ Autorisations et/ou authentification |
| ■ Base LDAP (avec le schéma Radius) | ➔ Autorisations et/ou authentification |
| ■ Domaine Windows                   | ➔ Authentification                     |
| ■ Base SQL                          | ➔ Autorisation                         |

L'identifiant (attribut User-Name) envoyé au serveur Radius est utilisé comme clé de recherche dans la base

# L'identifiant

Type d'authentification	Emetteur de l'identifiant	Protocole
adresse MAC	l'équipement réseau	RADIUS
login/password certificat	 le poste utilisateur (suppliquant)	RADIUS + 802.1X + EAP

# L'intérêt d'un serveur Radius

- Authentifier les machines/utilisateurs pour l'accès au réseau local
- Utilisable en filaire et sans-fil
- Placer les machines dans des sous-réseaux virtuels
- Plusieurs moyens d'authentification
- Initialiser les algorithmes de chiffrement des communications (WPA)  
Les communications WiFi peuvent être sécurisées
- Radius est un élément actif du réseau, pas seulement une base de donnée.  
Grâce aux modules ou attributs programmables.
- Interfaçage avec des logiciels de portails captifs
- Authentification distante par redirection de requêtes (proxy)
- Utilisable par d'autres types de serveurs (VPN)

# Implémentations Radius

## Open sources

- Freeradius
- Openradius
- Gnuradius

## Commerciales

- ACS (Cisco)
- IAS (Microsoft)

# Quelques protocoles d'authentification

Radius-MAC => **Authentification par adresses MAC**

- Equivalent à VMPS pour le filaire
- Pas recommandé en sans-fil

TLS => **Authentification mutuelle par certificats**

- Deux phases :
  - ✓ TLS Handshake - Authentification des certificats
  - ✓ TLS Record - Création d'un tunnel chiffré
- Seule la première phase est utilisée
- L'identifiant dans la base est le CN du certificat

PEAP et TTLS => **Authentification du client par login/password  
et authentification du serveur par son certificat**

- Met en œuvre TLS Handshake puis TLS Record
- Le protocole d'authentification du mot de passe circule dans le tunnel chiffré

802.1X

# Usage avec les réseaux virtuels

**Intérêt** : Une fois l'authentification faite, l'équipement réseau ouvre le port sur un VLAN (VLAN dynamique)

Utilisation des attributs :

✓Tunnel-Type	(VLAN)
✓Tunnel-Medium-Type	(Ethernet 802)
✓Tunnel-Private-Group-Id	<= Le numéro de VLAN

**Processus** :

- Le serveur trouve l'identifiant dans sa base
- Récupère les attributs Tunnel
- Envoie ces attributs à l'équipement réseau avec l'Access-Accept
- L'équipement réseau ouvre le port dans le VLAN contenu dans Tunnel-Private-Group-Id

Les bornes WiFi doivent être capables de gérer plusieurs VLAN par SSID

# Usage avec les réseaux virtuels

## ■ Exemples de configuration avec FreeRadius

0123456789ab Auth-Type := Local, User-Password == 0123456789ab  
Tunnel-Type = VLAN,  
Tunnel-Medium-Type = IEEE-802,  
Tunnel-Private-Group-Id = **3** Le VLAN

Adresse MAC

Authentication  
Par adresse MAC  
(Pas de 802.1x)

Dupont Auth-Type := EAP  
Tunnel-Type = VLAN,  
Tunnel-Medium-Type = IEEE-802,  
Tunnel-Private-Group-Id = **3**

Authentication 802.1X/EAP.  
Dupont peut se connecter depuis  
importe quel poste

Dupont Auth-Type := EAP, Calling-Station-Id == 0123456789ab  
Tunnel-Type = VLAN,  
Tunnel-Medium-Type = IEEE-802,  
Tunnel-Private-Group-Id = **3**

Authentication 802.1X/EAP.  
Dupont peut se connecter  
uniquement depuis le poste  
d'adresse MAC 0123456789ab

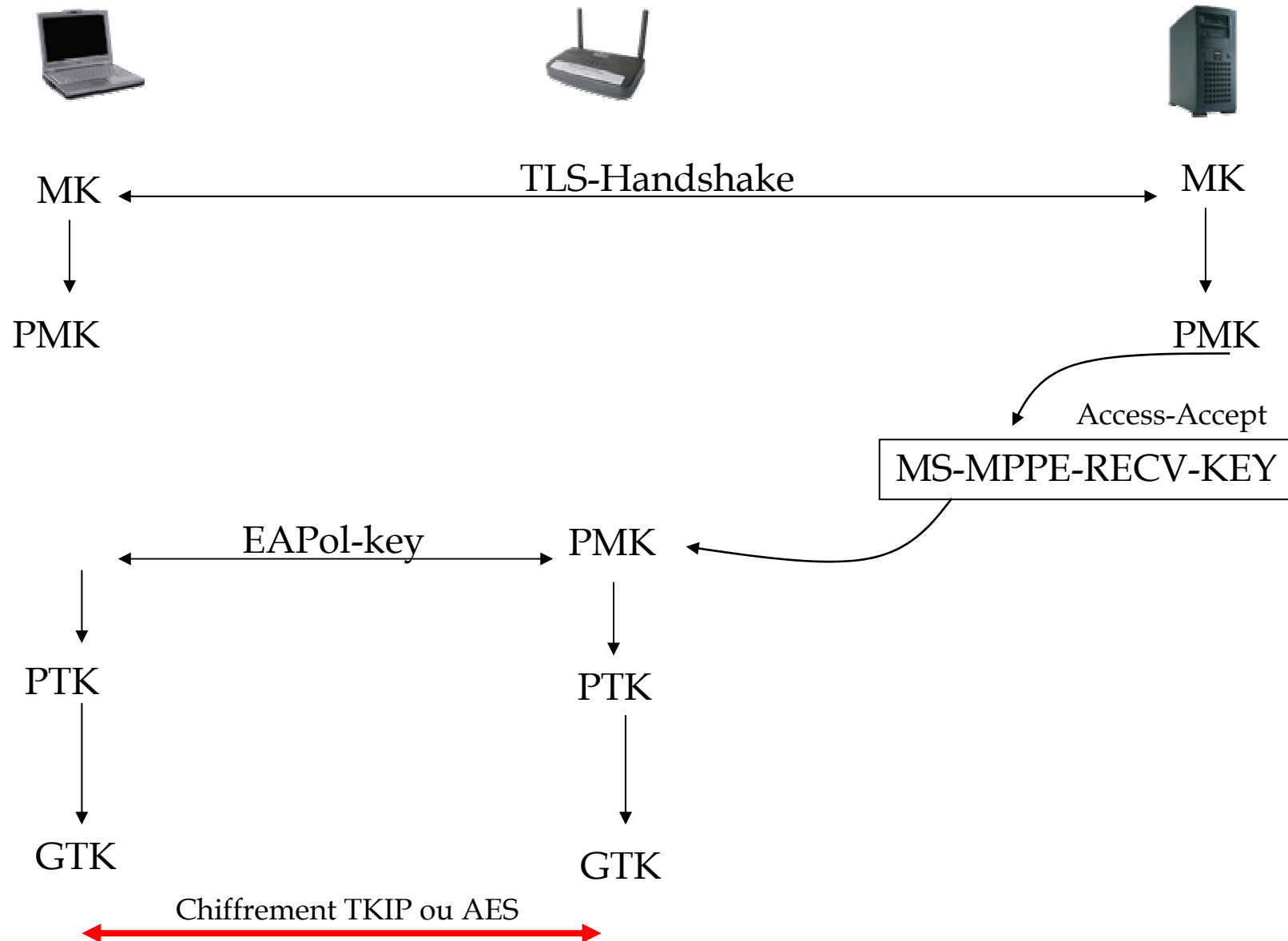
# Wifi Protected Access - WPA

**WPA = 802.1X/EAP + méthode de chiffrement**

- WPA permet :
  - ✓Authentification
  - ✓Chiffrement des communications (TKIP ou AES)
- L'authentification est réalisée par le serveur Radius avec EAP
- Le serveur Radius envoie à la borne une clé de chiffrement calculée à partir de la clé de session calculée dans la phase TLS Handshake.
- Cette clé est envoyée dans l'Access-Accept au moyen de l'attribut MS-MPPE-RECV-KEY
- Le rôle du serveur Radius s'arrête là
- La borne et le poste utilisateur dérivent des clés de chiffrement
- Un chiffrement symétrique est ensuite établi entre la borne et le poste utilisateur



# WPA : Initialisation des clés de chiffrement



# Critères d'authentification supplémentaires

Exemples avec Freeradius

- Authentification suivant l'équipement réseau  
Called-Station-Id ou Nas-IP-Address

*Identifiant* Auth-Type := EAP , Called-Station-Id== "*adresse mac*"

- Suivant le jour et l'heure

*Identifiant* Auth-Type := EAP, Login-Time = "*any0700-2000*"

- Expiration

*Identifiant* Auth-Type := EAP, Expiration = "*30 Mar 2007 00:00:00*"

- Imposer la méthode d'authentification

*Identifiant* Auth-Type := EAP,EAP-Type:=PEAP

# Modules et attributs programmables

Il est possible d'écrire des programmes qui s'exécuteront pendant le processus d'authentification/ autorisation.

Deux méthodes :

- Au moyen de modules déclarés dans la configuration Radius
- Avec l'attribut Exec-Program-Wait dans la base de données.

## Exemples :

- ✓ Générer des lignes de logs spécifiques
- ✓ Envoyer un mail avant expiration
- ✓ Faire des vérifications supplémentaires
- ✓ Modifier dynamiquement la base de données

...

# Modules et attributs programmables

- Dans le fichier de configuration du daemon FreeRadius (radiusd.conf) on définit un module spécifique.

```
exec logrecord {  
    wait = yes  
    program = "/usr/local/bin/logrecord %{Calling-station-Id} %{NAS-IP-ADDRESS} %{NAS-PORT} "  
    input_pairs = reply  
    output_pairs=none  
    packet_type = Access-Accept  
}
```

- Ce module est appelé dans une des sections de radiusd.conf

```
post-auth {  
    ....  
    logrecord  
    ....  
}
```

- Ecrire le programme appelé

/usr/local/bin/logrecord

```
#!/bin/sh  
  
CALLING_STATION_ID=$1  
NAS_IP_ADDRESS=$2  
NAS_PORT=$3  
logger "radiusd:ALLOW: ${CALLING_STATION_ID} -> VLAN${TUNNEL_PRIVATE_GROUP_ID//"/"/}, nas  
${NAS_IP_ADDRESS} port ${NAS_PORT} USER ${USER_NAME}"
```

# Modules et attributs programmables

- Appel d'un programme dans les attributs stockés dans la base de données (utilisation de l'attribut Exec-Program-Wait.)

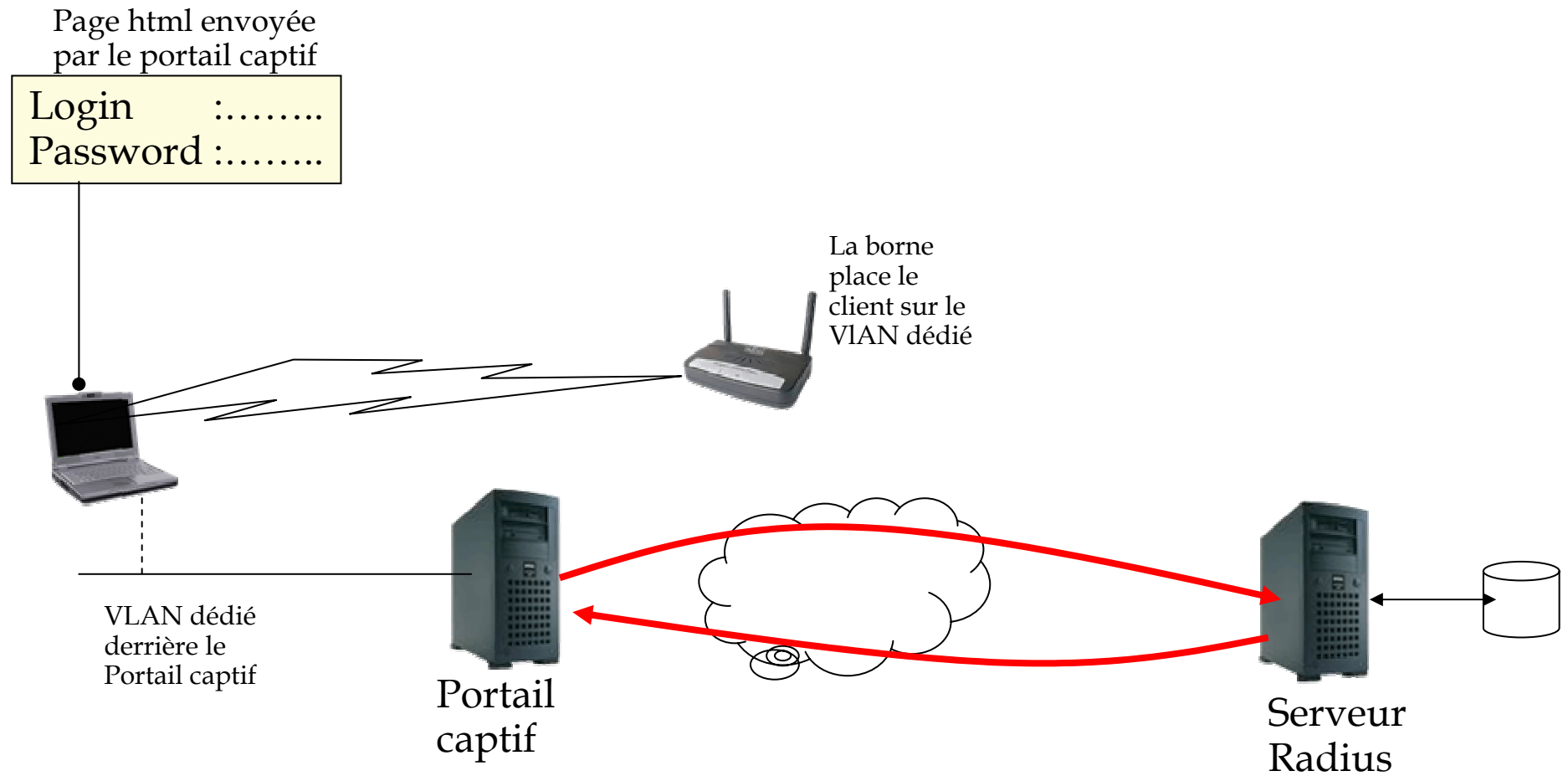
Exemple : Envoi d'un mail avant date d'expiration

```
Identifiant  Auth-Type := EAP, Calling-Station-ID == 0123456789ab, Expiration = "30 Mar 2007 00:00:00"  
              Tunnel-Type = VLAN  
              Tunnel-Medium-Type = IEE-802  
              Exec-Program-Wait = /usr/local/bin/un-programme 3 dupont "30 Mar 2007 00:00:00"
```

- Ecriture du programme correspondant

```
#!/bin/sh  
# Les attributs sont passés dans des variables d'environnement  
vlan=$1  
mail=$2  
expire=$3  
.... Calcul délai avant expiration  
.... Envoi d'un mail  
....  
  
echo « Tunnel-Private-Group-Id = $vlan »  
exit
```

# Radius et les portails captifs



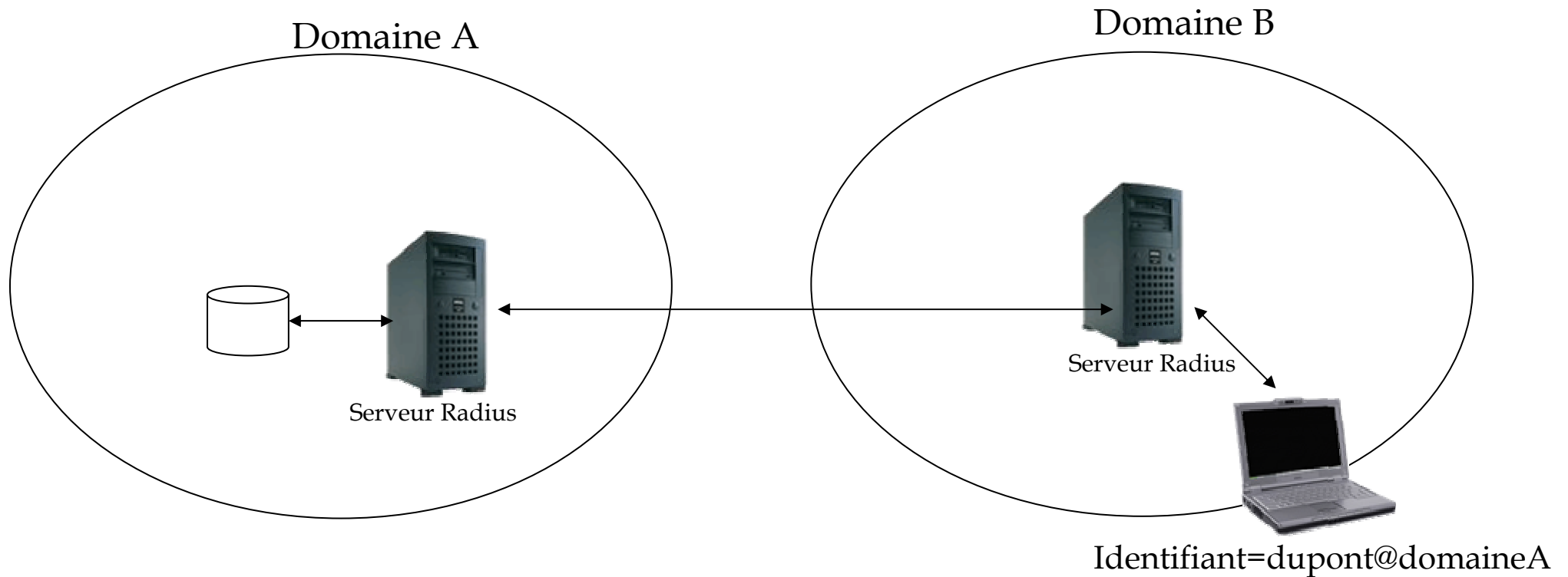
# Radius et les portails captifs

- Le serveur de portail captif est déclaré comme client dans la configuration du serveur Radius
- Le serveur Radius accepte les requêtes si elles viennent du portail captif (Called-Station-Id)
- Le portail captif envoie comme identifiant le login/password.
- Le portail captif agit comme routeur / firewall / dhcp

Plusieurs stratégies possibles :

- ✓ Inconnus toujours acceptés sur le portail captif (pas d'authentification)
- ✓ Enregistrement dans Radius des utilisateurs autorisés
- ✓ Auto-enregistrement (par exemple avec la programmation Radius)

# Proxy-Radius, principes



- C'est le serveur Radius du domaine auquel appartient l'utilisateur nomade qui l'authentifie.

- Il envoie un Access-Accept au serveur Radius du domaine d'accueil

- Le serveur d'accueil choisit le VLAN

- Principe de Eduroam



# Proxy-Radius, principes

## Domaine-A

- clients.conf  
Déclaration du serveur du domaine B comme client
- proxy.conf  
Déclaration du domaine-B
- Filtrage d'attributs

## Domaine-B

- clients.conf  
Déclaration du serveur du domaine A comme client
- proxy.conf  
Déclaration du domaine-A
- Filtrage d'attributs

# Transformer un serveur en client Radius

## Utilité :

- Se servir de Radius comme serveur central pour réaliser des authentifications
- Se servir de Radius pour obtenir des informations (adresse IP, règles de filtrage...)

# Transformer un serveur en client Radius avec pam\_radius

- Compilation/Installation du module pam\_radius sur le serveur-client
- Configuration de PAM
- Déclaration du serveur-client sur le serveur Radius (secret partagé dans clients.conf)
- Création du fichier pam\_radius\_auth sur le serveur-client pour inscrire le secret partagé.

Limitations :

- On ne peut obtenir qu'une authentification login/password
- Pas de retour d'attributs

# Transformer un serveur en client Radius avec radclient

- Radclient est un utilitaire fourni avec FreeRadius
- Il faut installer Freeradius sur le serveur client (on lance pas de daemon, pas de config)
- Permet d'interroger la base Radius et d'obtenir les attributs dans une chaîne de caractères

# Transformer un serveur en client Radius avec radclient

## Le fichier users du serveur Radius

```
Dupont  Auth-Type := Local, User-Password == test, Nas-IP-Address == @IPdu-serveur-client
        Framed-IP-Address = 172.16.0.3,
        Tunnel-Type = VLAN,
        Tunnel-Medium-Type = IEEE-802,
        Tunnel-Private-Group-Id = 3,
```

## Sur le serveur-client

```
echo User-Name=Dupont,User-Password=test | radclient -x serveur-radius auth secret > tempfile
```

## Réponse reçue par le serveur-client

```
cat tempfile
Sending Access-Request of id 229 to 172.16.0.10 port 1812
  User-Name = " Dupont"
  User-Password = " test"
rad_recv: Access-Accept packet from host 172.16.0.10:1812, id=229, length=42
  Framed-IP-Address = 172.16.0.3
  Tunnel-Type:0 = VLAN
  Tunnel-Medium-Type:0 = IEEE-802
  Tunnel-Private-Group-Id:0 = "3"
```

# Openvpn et Radius et radclient

## Premier exemple

- But : le serveur Openvpn demande à Radius si un utilisateur est connu et quelle adresse IP lui donner
- Openvpn authentifie les utilisateurs avec leur certificat  
Si le certificat n'est pas authentifié, l'utilisateur est rejeté
- Appel d'un script pour interroger le serveur Radius avec l'ordre **client-connect**
- Ce script exécute une commande radclient (utilise Radius comme base d'autorisation)
- Si Access-Reject cela signifie que cet utilisateur n'est pas autorisé à utiliser OpenVpn. (exit 1)
- Si Access-Accept, analyse de la chaîne de caractères renvoyées pour en extraire l'adresse IP.
- Ensuite on écrit dans le fichier de config temporaire géré par Openvpn

**echo ifconfig *adresse-envoyée-par radius adresse-server-openvpn* > \$1**

### Intérêts :

- ✓ Même avec un certificat valide, un utilisateur sera autorisé à se connecter uniquement s'il est enregistré dans la base de Radius.
- ✓ Un utilisateur reçoit toujours la même adresse IP.
- ✓ Possibilité de mettre en œuvre un serveur VPN multi-vlan

# Openvpn et Radius et radclient

## Deuxième exemple

- But : Exécuter, sur le serveur Openvpn, une commande iptables pour un utilisateur particulier
- Méthode : Créer un attribut qui contient une commande Iptables qui sera envoyée par le serveur Radius au serveur Openvpn au moment de la connexion de l'utilisateur.
- Sur le serveur Radius :

```
Jean Dupont  Auth-Type := Local, User-Password == test, Nas-IP-Address == @IP-du-serveur-client  
             Ipables = « -I FORWARD -s 172.16.0.3 -j ACCEPT »,  
                 Framed-IP-Address = 172.16.0.3,  
                 Tunnel-Type = VLAN,  
                 Tunnel-Medium-Type = IEEE-802,  
                 Tunnel-Private-Group-Id = 3
```

- Sur le serveur Openvpn, le script client-connect est du type :

```
echo User-Name=Dupont,User-Password=test | radclient -x serveur-radius auth secret > tempfile  
✓ Recherche de l'adresse IP et affectation  
✓ Recherche du mot clé « Iptables » dans le fichier tempfile  
✓ Extraction de la commande Iptables  
✓ Exécution de la commande Iptables
```

# Openvpn et Radius et radiusplugin

- Radiusplugin permet une authentification sur le serveur Radius avec retour de certains attributs (Framed-IP-Address, Framed-Routes, Acct-Interim-Interval)
- S'utilise avec l'option plugin de Openvpn
- Projet intéressant mais pas encore au point (http://www.nongnu.org/radiusplugin)  
Bien moins souple que radclient



# Bilan

Le serveur Radius est le « moteur » de la mobilité  
C'est un élément structurant du réseau

- Apporte des solutions pour authentifier sans-fil/filaire (unification)
- Apporte des solutions pour exploiter au mieux un réseau fortement structuré
- Apporte des solutions pour l'accueil des visiteurs
- Extension à d'autres services réseau

# Références

- RFC 2865, RFC 2868, RFC 2869 ..... (<http://www.ietf.org>)
- <http://www.freeradius.org>
- <http://www.freeradius.org/list/users.html>
- <http://www.levkowetz.com/pub/ietf/drafts/eap/rfc2284bis/draft-ietf-eap-rfc2284bis-07.html>  
(EAP)
- <http://www.wi-fi.org> (WiFi Alliance/WPA)
- Livre: Authentification réseau avec Radius par Serge Bordères - Eyrolles