# Principes généraux de l'authentification 802.1x

## Les acteurs

Trois acteurs interviennent dans la mise en oeuvre des protocoles 802.1x et Radius :

### Le supplicant

C'est l'équipement de l'utilisateur, son PC, son téléphone IP, sa tablette ou son smartphone qui souhaite se connecter au réseau de l'organisation. C'est le **client final** de la demande de connexion. Pour cela :

- l'utilisateur s'identife
- auprès d'un poste de sécurité central,
- qui décide d'autoriser l'accès et des prérogatives à lui accorder aprsè son admission.

> Le **supplicant** est aussi appelé **client demandeur** ou **client final**.

L'équipement de réseau sur lequel le client final se connecte (commutateur ; borne Wifi compatible 802.1x) relaye, en tant que client RADIUS, cette demande de connexion à un serveur d'authentification RADIUS, qui va, par exemple, identifier la personne en rapprochant le nom de connexion et le mot de passe de ceux stockés dans un annuaire LDAP ou encore une base de données SQL.

Si l'identification réussit, l'accord est transmis au client RADIUS qui **ouvrira** alors le port de connexion.

### Le serveur d'authentification

> Le **serveur d'authentification** est aussi appelé **serveur d'identification**.

### Le client Radius

C'est un équipement réseau (commutateur, borne wifi, ...) qui est central dans la connexion 802.1x et qui doit gérer le protocole 802.1x et le protocole d'authentification EAP.

> Le **client RADIUS** est aussi appelé **Authenticator** ou **NAS** (Network Access Server) ou encore **contrôleur d'accès**.

## Le protocole 802.1x

Le protocole 802.1x est une solution standard de sécurisation de réseaux mise au point par l'IEEE en 2001. 802.1x permet d'authentifier un utilisateur souhaitant accéder à un réseau (câblé ou Wifi) grâce à un serveur central d'authentification.

L'autre nom de 802.1x est **Port-based Network Access Control** ou **User Based Access Control**.

802.1x permet de **sécuriser** l'accès à la **couche 2** (liaison de donnée) du réseau. Ainsi, tout utilisateur, qu'il soit interne ou non à l'entreprise, est dans l'obligation de s’authentifier avant de pouvoir faire quoi que soit sur le réseau. Certains équipements de réseau compatibles 802.1x peuvent réserver un traitement particulier aux utilisateurs non authentifiés, comme le placement dans un VLAN **guest**, une sorte de quarantaine sans danger pour le reste du réseau.

802.1x a recours au **protocole EAP** (Extensible Authentification Protocol) qui constitue un support universel permettant le transport de différentes méthodes d'authentification qu'on retrouve dans les réseaux câblés ou sans-fil.

802.1x nécessite donc la présence d'un **serveur d'authentification** qui peut être un **serveur RADIUS** (serveur Microsoft NPS - Network Policy Server ; Cisco ; un produit libre comme FreeRADIUS) ou encore un serveur **TACACS** dans le monde fermé des équipements Cisco.

Un port d'un commutateur réglé en mode 802.1x peut se trouver dans deux états distincts :

- État **contrôlé** si l'authentification auprès du serveur RADIUS a **réussi**.
- État **non contrôlé** si l'authentification a échoué.

La réussite ou l'échec de l'authentification va donc ouvrir ou fermer le port à toute communication.

Un port ouvert va, par exemple, permettre au client final d'obtenir une adresse IP auprès d'un serveur DHCP.

Dans des implémentations plus cloisonnées, le serveur RADIUS indiquera par exemple au client RADIUS dans quel VLAN placer le client final.

# RADIUS

RADIUS (acronyme de Remote Authentication Dial-In User Service) est un **protocole client-serveur** permettant de centraliser des demandes d'authentification relayées par des équipements de réseau, comme des commutateurs ou bornes Wifi, considérés alors comme ses clients.

Par extension, un serveur qui centralise des demandes d'authentification et les soumet à un service d'annuaire LDAP ou à un service de base de données SQL est appelé serveur RADIUS.

RADIUS interroge une base de données d'authentification et d'autorisation qui peut être un domaine Active Directory, une base LDAP ou une base de données SQL. Ces bases ou annuaires peuvent se trouver sur le serveur lui-même ou sur un serveur tiers. Certaines implémentations de RADIUS disposent d'une base de données en propre.

A l'origine, RADIUS était surtout utilisé pour l'identification des clients des FAI, ses capacités de comptabilisation des accès (accounting) permettant notamment la journalisation des accès et leur facturation. RADIUS a été utilisé par la suite en entreprise pour l'identification des clients finals WIFI et pour l'identification des clients finals câblés.

## Rôles du serveur RADIUS

#### 1. Authentification

Le serveur RADIUS

- doit authentifier les requêtes qui sont issues des clients finals,
- via les clients RADIUS.

Base de l'authentification selon le protocole d'authentification négocié avec le client final :

- soit un couple identifiant/mot de passe,
- soit un certificat. Cela dépendra du protocole d'authentification négocié avec le client final.

#### 2. Autorisation

Le serveur RADIUS décide quoi faire du client authentifié en délivrant une autorisation.

Pour cela, le serveur RADIUS envoie des informations (**attributs**) aux clients RADIUS comme par exemple le numéro du VLAN dans lequel placer le client authentifié.

#### 3. Comptabilisation

Le serveur RADIUS comptabilise/journalise plusieurs données liées à la connexion : date et l'heure ; adresse MAC de l'adaptateur réseau du client final ; le numéro de VLAN, etc.

C'est son rôle comptable ou **d'accounting**.

Le serveur RADIUS se range dans le modèle AAA (Authentication, Authorization, Accounting).

RADIUS peut aussi servir à centraliser les accès sécurisés aux pages ou aux terminaux de paramétrage de tous les équipements réseau : commutateurs, routeurs, bornes wifi, contrôleurs wifi, etc.

# Les protocoles d'authentification

**EAP** est la couche protocolaire de base de l'authentification. Elle va servir à faire passer un dialogue d'authentification entre le client final et le serveur RADIUS alors que le port de connexion est fermé à toute autre forme de communication.

C'est un **protocole extensible**, au sens où il va permettre l'évolution de méthodes d'authentification transportées, de plus en plus sûres au cours du temps.

## Les méthodes d'authentification

### PAP

Le premier protocole a été **PAP** (Password Authentification Protocol) avec lequel les mots de passe circulaient en clair. La sécurité proposée par ce protocole est faible.

### CHAP et MS-CHAP

Le second protocole qu'ont utilisé les serveurs RADIUS a été **CHAP* (Challenge Handshake Authentication Protocol). Il est défini dans la RFC 1994. Avec CHAP, il n'y a pas d'échange de mots de passe sur le réseau. Les deux interlocuteurs, qui disposent donc de la même chaîne de caractère secrète, s'authentifient sans échange du mot de passe par une technique de challenge (ou défi) basée sur une fonction de hachage à sens unique du secret partagé, telle que MD5. Cette méthode était disponible avec le couple XP/Windows-2003-Server, mais ne l'est plus en génération Seven/2008. Au début de la connexion, le serveur réclame la preuve de l'identité du client, en lui demandant de chiffrer une information. Le client ne peut relever le défi que s'il possède effectivement la clé unique et secrète partagée. Dialogue client-serveur avec CHAP * A. Après l'établissement de la connexion, l'authentificateur envoie une valeur aléatoire xxxxxx au client. * B. Le client concatène cette valeur xxxxxx au secret partagé, applique une fonction de hachage (telle que MD5) sur la chaîne obtenue et retourne le résultat. * C. Le serveur effectue la même opération et compare avec le résultat reçu. La connexion n'est acceptée que si le résultat est identique. * D. A intervalle régulier, il y a un nouveau défi à relever pour pérenniser la connexion. Microsoft a développé une variante de CHAP appelée MS-CHAP qui ajoute une authentification mutuelle, MSCHAP-V1, puis MSCHAP-V2. Dialogue client-serveur avec MSCHAP-V2 : * A. Le serveur envoie au client une chaîne composée d'un identifiant de session et une chaîne aléatoire xxxxx. * B. Le client renvoie son nom d'utilisateur et le résultat d'un hachage de la chaîne aléatoire xxxxx + l'identifiant de session + le mot-de-passe, et une seconde chaîne aléatoire yyyyy. * C. Le serveur vérifie le résultat (succès/échec) et retourne celui-ci, avec un hachage de la chaîne yyyyy et du mot de passe utilisateur. * D. Le client vérifie enfin la correspondance entre les chaînes. E. La connexion est établie. === PEAP === PEAP est un protocole de transfert sécurisé (P comme Protected) d'informations d'authentification. Il a été mis au point par Microsoft, Cisco et RSA. Il ne nécessite pas de certificat sur les postes clients, contrairement à EAP/TLS. MS-CHAP s'appuie sur PEAP. === Articulation EAP / PEAP / MSCHAP-V2 === * EAP est le mécanisme permettant à un client final de pouvoir communiquer sur un port 802.1x fermé à toute autre forme de communication. * PEAP ajoute la notion de protection des échanges par tunnel à ce mécanisme * MSCHAP est la méthode de reconnaissance mutuelle du client serveur et du serveur RADIUS qui passe par ce tunnel. ==== Les différentes phases (simplifiées) d'une connexion 802.1x ====**

Au démarrage de la communication, le client final est prié d'envoyer ses identifiants au serveur RADIUS. Or, à ce moment là, le client final ne connaît pas l'adresse du - ou des - serveurs RADIUS du réseau. Il ne dispose peut-être même pas d'adresse IP. De même, le port du commutateur sur lequel il est connecté est censé être fermé (état non contrôlé). En réalité, le port contrôlé du commutateur n'est pas totalement fermé. Il va laisser passer le protocole EAP (Phase 1 sur le schéma suivant). Cette communication ne peut donc se faire que par des trames Ethernet de base et non par des paquets IP. Le client final peut donc envoyer son identité dans un paquet EAP au commutateur. Celui-ci le retransmet, encapsulé dans un paquet au format RADIUS, au premier serveur RADIUS de sa liste (s'il en connaît plusieurs) (Phase 2). Le serveur RADIUS reçoit le paquet et interroge sa base de données (Phase 3). Il renvoie le résultat de cette interrogation au commutateur (Phase 4), sous forme d'un commandement d'ouverture du port, éventuellement assorti d'un numéro de VLAN dans lequel placer le client final. A partir de ce moment seulement, il peut y avoir d'autres trames échangées entre le client final et le reste du réseau, comme une trame de requête DHCP par exemple. ==== Retour Authentification 802.1x ==== * Authentification réseau avec le protocole 802.1x

From:
/ - **Les cours du BTS SIO**

Permanent link:
**/doku.php/reseau/802.1x/principes?rev=1700423558**

Last update: **2023/11/19 20:52**