UNIVERSITÉ DE STRASBOURG

Projet mobilité

Philippe Pittoli

11 décembre 2014

TABLE DES MATIÈRES

1 Portail captif et connexion sécurisée 2
1.1 Matériel utilisé . . . 2
1.2 Solutions logicielles utilisées . . . 2
1.3 Fonctionnement . . . 3
1.3.1 Portail captif . . . 3
1.3.2 Connexion sécurisée 802.1X . . . 3
1.3.3 plan d'adressage . . . 3
1.3.4 Règles de pare-feu . . . 4
2 Mobilité 4
3 Temps passé sur les différentes parties 5
4 Difficultés rencontrées 5
4.1 Difficultés avec pfsense . . . 5
4.2 Difficultés avec pepperspot . . . 5
4.3 Difficultés avec SNMP et le point d'accès cisco WAP121 . . . 6
5 Conclusion 6

1 Portail captif et connexion sécurisée

1.1 Matériel utilisé

Concernant le matériel j'ai pris un AP cisco WPA121 comme il nous a été proposé. J'ai installé les différents logiciels (pepperspot, freeradius, etc) sur une clé USB persistante avec Debian stable, configurée pour être compatible avec le matériel d'un des ordinateurs de la salle C315. Cela fait que le nommage des cartes réseau est statique, et l'adresse IP configurée pour le tunnel broker est celle d'origine de la machine en C315.

1.2 Solutions logicielles utilisées

Premièrement, j'ai commencé par travailler sur pfsense avant de m'apercevoir au dernier moment qu'il ne supportait pas une fonctionnalité nécessaire au projet : la connexion au portail captif en IPv6. Il répondait cependant à toutes les autres fonctionnalités qui était demandées, tout en offrant une interface simple à configurer et un système de sauvegarde de la configuration simple et efficace.

Je me suis tourné par la suite vers pepperspot pour continuer le projet.

Les différents logiciels utilisés sont :

- Debian GNU/Linux version stable sur une clé USB persistante
- RadvdDesign [b], pour émettre des router advertisements
  - annonce des plages d'adresses IPv6 pour auto configuration
- Apache2Foundation pour le serveur http
- PepperspotVincent pour la gestion du portail captif
  - portail captif ipv(4|6), radius
- FreeradiusProject and Contributors, serveur RADIUS avec un backend MySQL
- DaloRADIUSTal pour la gestion des utilisateurs radius
  - interface simple
- Monitoring
  - Ntopntop pour visualiser la bande passante consommée par connexion
  - MonitDesign [a] pour faire des tests simples
    - test du statut d'un logiciel
    - permet l'arrêt et le démarrage d'un service
    - simple à utiliser et à mettre en place
  - MuninCommunauté [a] pour visualiser via des graphes un certain nombre d'informations sur le système
    - charge, uptime
    - taux d'utilisation du réseau
    - temps de latence des disques
  - Page de monitoring développée en plus
    - requêtes au serveur MySQL sur la bdd de freeradius
    - tient compte de la relation client-IP
    - suit la consommation par utilisateur (octet)
    - comporte des liens vers les autres outils de monitoring
  - Page de monitoring du point d'accès
    - basé sur des requêtes SNMP

— très limité, voir « difficultés rencontrées »

J'ai rajouté une page de monitoring listant les utilisateurs connectés. Les clients visibles sont ceux qui se sont connectés il y a moins de trois heures. Cependant seuls les utilisateurs connectés par le portail captif sont visibles. Voir la partie « difficultés rencontrées ».

1.3 Fonctionnement

L'infrastructure est composée d'un point d'accès cisco WAP121 et d'un ordinateur qui démarre sur la clé USB debian stable contenant toute la configuration nécessaire. Deux cartes réseau sont nécessaires : l'une allant vers Internet et l'autre connectée au point d'accès.

La connexion IPv6 est apportée par un tunnel Hurricane Electric qui route un /48. Cela est nécessaire pour avoir deux /64 (une plage /64 par VLAN) ce qui n'est pas disponible en salle C315 directement.

Le point d'accès est configuré pour annoncer deux SSID, à savoir « fakeosiris-captiveportal » et « fakeosiris-sec ».

— fakeosiris-captiveportal : VLAN 2, portail captif
— fakeosiris-sec : VLAN 3, connexion sécurisée 802.1X, EAP-TTLS-PAP

1.3.1 Portail captif

Lorsqu'une connexion se fait sur le premier SSID, la personne est redirigée vers le portail captif géré par pepperspot. Si la personne se connecte à une adresse IPv4 elle sera redirigée vers le portail captif en IPv4, et en IPv6 si la personne se connecte à une adresse en IPv6. Tant que la personne n'est pas authentifiée, aucune connexion n'est possible vers l'extérieur.

Le portail captif vérifie ensuite si la personne peut se connecter en envoyant une requête au serveur RADIUS (freeradius).

1.3.2 Connexion sécurisée 802.1X

Cette fois, l'utilisateur se connecte au SSID fakeosiris-sec. Le point d'accès va directement faire des requêtes au serveur radius pour authentifier les utilisateurs.

J'ai utilisé le protocole EAP-TTLSIETF car c'est un standard développé à l'IETF, il offre un bon niveau de sécurité, et il rend le certificat côté utilisateur optionnel. PAP est utilisé bien qu'il fasse passer les mots de passe en clair car il se base sur une connexion TLS déjà sécurisée.

1.3.3 plan d'adressage

— 198.18.255.0/30 : interconnexion point d'accès - ordinateur
— VLAN 2, clients du portail captif
  — 198.18.0.0/24
  — 2001 :470 :ca59 :2 : :/64
— VLAN 3, clients connexion sécurisée
  — 198.18.1.0/24
  — 2001 :470 :ca59 :3 : :/64

La connexion de l'ordinateur en salle C315 au réseau de la fac et la connexion HE sont statiques afin de prévenir des éventuels problèmes de DHCP.

1.3.4 Règles de pare-feu

Les règles du pare-feu *ip(6)tables* sont simples et nombreuses. Par exemple les connexions inter-VLAN ne sont pas autorisées.

```
# drop du trafic inter-VLAN
$IPTABLES -t filter -I FORWARD -i $INTIF -o $INTIFSEC -j DROP
$IPTABLES -t filter -I FORWARD -i tun0 -o $INTIFSEC -j DROP
$IPTABLES -t filter -I FORWARD -i $INTIFSEC -o $INTIF -j DROP
$IPTABLES -t filter -I FORWARD -i $INTIFSEC -o tun0 -j DROP
```

Il n'est pas possible pour une personne connectée par le portail captif (connexion non sécurisée) de se connecter en SSH à notre infrastructure (c'est à dire la machine hébergeant les services et le point d'accès).

```
# drop du trafic des clients à destination de l'infra
$IPTABLES -t filter -I FORWARD -i $INTIF -d 198.18.255.0/30 -j DROP
$IPTABLES -t filter -I FORWARD -i tun0 -d 198.18.255.0/30 -j DROP
$IPTABLES -t filter -I FORWARD -i $INTIFSEC -d 198.18.255.0/30 -j DROP
```

Les messages ICMP sont autorisés, tout comme les requêtes DNS. De même, on autorise les connexions au serveur web et aux divers services de monitoring.

2 Mobilité

Concernant la partie mobilité avec le logiciel UMIPCommunauté [b], l'installation s'est faite en recompilant le noyau Linux (version stable la plus récente 3.17.3). Les options du noyau à activer sont dans la documentation sur le site officiel.

Avant tout j'ai nettoyé le répertoire du noyau via **make distclean**. Puis j'ai généré une configuration de base compatible avec mon matériel via la commande **make localmodconfig**. Ensuite j'ai ajouté les options nécessaires à UMIP, j'ai compilé via **make**, et installé le nouveau noyau directement sur mon système : **make modules_install** ; **make install** ; **make headers_install**.

Le programme UMIP n'étant pas disponible pour mon système de manière empaquetée, je l'ai compilé directement depuis les sources. J'ai mis en commentaire toute la fin du fichier de configuration concernant IPSec comme demandé, et une option a été désactivée (UseMnHaIPsec). J'ai également modifié la partie **MnHomeLink** comme suit :

```
# info nécessaires pour contacter le HomeAgent et choisir notre adresse IP
MnHomeLink "br0" {
  HomeAgentAddress 2001:660:4701:f055:ffff::1;
  HomeAddress 2001:660:4701:f055:ffff::1001/64;
}
```

Le fichier de configuration est **/usr/local/etc/mip6d.conf**, fichier utilisé par défaut par le programme. La commande à taper pour activer la mobilité se résume à **mip6d**.

Pas de problème particulier avec la mobilité et UMIP, tout fonctionne et c'est assez rapide à mettre en place.

3 Temps passé sur les différentes parties

Les temps donnés ici sont approximatifs.

Concernant la partie sur pfsense, j'ai passé 20 heures approximativement pour comprendre le fonctionnement, faire des tests et configurer la plus grande partie de l'application.

Après m'être redirigé vers pepperspot, j'ai passé cette fois ci 15 heures uniquement pour manipuler ce logiciel, avoir des règles de pare-feu et freeradius avec simplement les utilisateurs dans un fichier texte pour avoir une base fonctionnelle.

La configuration d'une base de données (MySQL) d'utilisateurs radius pour freeradius, plus un logiciel pour gérer les utilisateurs simplement et tester les différentes fonctionnalités m'a pris quatre à cinq heures supplémentaires.

La configuration de la mobilité a duré quatre heures. Le temps de compiler un noyau qui réponde aux exigences de l'exercice était assez simple, mais je souhaitais en plus qu'il me soit utilisable quotidiennement, et avec mes contraintes.

La partie monitoring a pris six heures supplémentaires.

Enfin, l'écriture de ce rapport m'a pris quatre heures supplémentaires.

Le total revient à presque cinquante cinq heures.

4 Difficultés rencontrées

4.1 Difficultés avec pfsense

Tout d'abord, j'ai souhaité installer le système pfsense sur mon ordinateur portable. J'ai rencontré un certain nombre d'erreurs à l'installation assez particulières, par exemple la clé USB d'installation s'éteignait au début de l'installation. La documentation faisait mention de ce problème mais la solution donnée ne fonctionnait pas : il était dit qu'il fallait utiliser une option dans le menu de démarrage de l'installateur pour préciser que l'installation se faisait par clé USB, mais malgré cette option, l'erreur revenait. J'ai alors décidé de faire le projet directement sur les machines en salle C315.

Le système pfsense a une certaine logique d'un point de vue de l'interface, cela m'a pris un peu de temps pour comprendre le raisonnement à avoir pour configurer une connexion à un tunnel broker.

La documentation a été d'une grande aide, et les différentes pages me semble bien faite. Cependant, le manque d'organisation pour trier les pages est réellement dérangeant pour s'y retrouver.

4.2 Difficultés avec pepperspot

Tout d'abord, la documentation laisse à désirer. Il m'a fallu du temps pour comprendre comment faire fonctionner le logiciel correctement, avec des règles de pare-feu correctes et freeradius.

En suivant les différents messages d'erreurs j'ai su retrouver un script CGI manquant après installation. Une fois ce script replacé tout marchait, mais cela m'a fait perdre un peu de temps inutilement.

4.3 Difficultés avec SNMP et le point d'accès cisco WAP121

J'ai souhaité récupérer la liste des utilisateurs connectés avec l'AP cisco en mode sécurisé. Malheureusement, l'AP n'envoie pas d'informations au serveur radius, qui pourraient être directement obtenues en fouillant la base de données.

Comme le point d'accès a un serveur SNMP, j'ai voulu aller directement lui envoyer des requêtes. J'ai commencé à chercher la MIB qui me serait utile, et j'ai finalement téléchargé un ensemble de documents décrivant les MIB. J'ai parcouru ces MIB, normalement utiles pour le point d'accès WAP121, en les copiant dans le dossier **/usr/share/snmp/mibs** et je les ai parcourues avec le logiciel **tkmib**. Je n'ai cependant pas réussi à requêter l'AP pour qu'il me donne les informations voulues.

En définitive, les seules informations qui sont sont l'objet de requêtes sont l'uptime de l'appareil, et des informations générales sur celui-ci. Par exemple, quel est son noyau.

5 Conclusion

L'ensemble des fonctionnalités demandées sont présentes. Une fonctionnalité supplémentaire qui aurait été intéressant d'avoir aurait été la possibilité d'avoir le tableau listant tous les utilisateurs connectés de manière sécurisée qui fonctionne comme ceux connectés au portail captif.

Ce projet a été enrichissant d'un point de vue technique, puisqu'il m'a permis de revoir différents logiciels et protocoles, recompiler mon noyau aux petits oignons et découvrir en profondeur pfsense.

Références

Communauté. *Munin*, a. URL `http://munin-monitoring.org/`.

Communauté. *UMIP*, b. URL `http://umip.org/`.

Litech Systems Design. *Monit*, a. URL `http://mmonit.com/monit/`.

Litech Systems Design. *Linux IPv6 Router Advertisement Daemon*, b. URL `http://www.litech.org/radvd/`.

Apache Software Foundation. *Apache2*. URL `https://httpd.apache.org`.

IETF. *EAP-TTLS draft*. URL `http://tools.ietf.org/html/draft-funk-eap-ttls-v1-01`.

ntop. *NTOP*. URL `http://www.ntop.org/`.

The FreeRADIUS Server Project and Contributors. *FreeRadius*. URL `http://freeradius.org/`.

Liran Tal. *DaloRADIUS*. URL `http://www.daloradius.com/`.

Sebastien Vincent. *PepperSpot, portail captif*. URL `http://pepperspot.sourceforge.net/`.