

# PV Réunion Réseau PDCLP 5

Nero, Rico, Gick, Krunch, FiFi, et BoB, plus tard

May 10, 2005

## 1 Matériel

CISCO Supervisor 3

- 2948 Switch de table
- 5500 Intermédiaire
- 6500 Backbone Fibre (peut être 8500)

Mettre en place deux 6500 avec redondance (pour pas avoir la blague de l'année dernière)

## 2 Reprise de l'année précédente

### 2.1 Internet

Même chose que l'année dernière, deux lignes vers la zone Internet, TVCableNet

- CARP : Load Balancing et redondance sous OpenBSD, plusieurs machines partagent la même IP
- VLAN avec Cisco (demander explications à Nero :D )

## 3 Nouveau et idées

### 3.1 Trafic non désiré

Au niveau Extended ACL sur le Cisco 6500

- Deny All, on autorise ce qu'on connaît, mais attention, il faudra mettre à jour le trafic non connu mais autorisé,  
autre problème, le Cisco 6500 doit éventuellement être redémarré pour prendre la config en compte
- On autorise tout, on bloque ce qu'on connaît, et après la première journée on prend en compte ce qu'on a pas encore répertorié et explicitement bloqué

Que fait-on du broadcast ?

### **3.2 DHCP**

Fait en France depuis bien longtemps, attention, il faut penser au niveau des DHCP des joueurs (partage de connexion windows)

Donc mettre le DHCP directement au niveau Cisco, la réponse ira plus vite et on peut "doubler" les DHCP user

Pour accès aux logs on utilisera SNMP Trap

### **3.3 Identification**

On isole les nouveaux connectés dans un VLAN

Une fois enregistré ils sont autorisés sur le réseau joueur

Une fois sur le VLAN, le joueur a accès uniquement à un serveur WEB, où il doit entrer son ID UNIQUE (concernant l'inscription)

On se retrouve avec une correspondance Port Réseau - Adresse MAC - Adresse IP

On peut les identifier au niveau des serveurs Web

Penser aussi à mettre tout "à plat"

Dans la même DB, ou en correspondance, avoir le Nick, Nom, Adresse IP, Scores etc ...

### **3.4 IRC Quakenet**

Ne pas oublier de demander l'autorisation à Quakenet

### **3.5 Jabber - Transport MSN/ICQ/...**

Installation d'un serveur Jabber sur lequel les joueurs peuvent se connecter et discuter au travers de la LAN

Autorisation de connexion à MSN et autre via les passerelles, on pourra facilement shaper le trafic et pas de transfert de fichier

### **3.6 Serveur**

Développement par FAI (Fully Automated Installation)

Distribution Ubuntu Server vs. Woody ?

Paquets perso, pour Kernel and others -> Par FAI

## 4 Post Scriptum (by Krunch)

Cette réunion a été bien fructueuse je pense. Les principaux points sont résumés dans le pdf de Fifi mais c'est pas forcément très compréhensible pour ceux qui n'étaient pas là. Je sais pas si ce que je raconte dans ce post est plus compréhensible mais au moins c'est plus détaillé.

### 4.1 Matériel

Au niveau des switches on sait qu'on aura: - un Catalyst 2948 par table - un (ou deux) 6500 comme "hub" central Si on a deux 6500 on peut les configurer pour qu'ils soient redondants de manière à ce que si un des deux décide de prendre des vacances (remember last summer), l'autre prend le relais et le réseau continue de fonctionner comme si de rien n'était. Devrait aussi y avoir moyen d'éviter de centraliser tout en faisant du spanning tree mais ça a l'air d'être un peu la foire à configurer.

### 4.2 Traffic Filter/Shaping

On peut utiliser les ACL sur le(s) 6500 pour bloquer le trafic "suspect". Les ACL ne permettent de bloquer que par paquet et selon l'adresse/port source/destination si j'ai bien compris. Le but n'est donc pas de sécuriser mais plutôt de diminuer le trafic dû aux vers/virus/... Soit on laisse tout passer sauf une liste de ports suspects connus. Soit on laisse passer uniquement les ports "connus" mais alors dès que quelqu'un va vouloir jouer à un jeu qu'on a pas listé ça va pas marcher (si les joueurs sont sur des tables différentes). À noter que d'après Néro, il est nécessaire de redémarrer le 6500 pour pouvoir effectuer des changements dans les ACL. Si on en a deux avec de la redondance qui marche correctement ça devrait pas trop poser de problème mais ça veut quand même dire qu'on peut pas se permettre de mettre les filtres à jour toutes les 5 minutes.

On limite le broadcast au minimum (5%) sans complètement le stopper.

On fait tourner un Nessus en permanence qui log les failles trouvées chez les joueurs quelque part et les gens qui s'embêtent peuvent aller patcher/déverroller les joueurs concernés. Un Snort ça serait bien aussi mais il reste encore des trucs à régler (où le placer pour qu'il intercepte le plus de trucs possibles sans ralentir le trafic et en droppant le moins possible ?). Faudrait rendre des antivirus et les patches disponibles quelque part. Le plus simple est probablement de monter un proxy qui laisse tout le monde accéder à windowsupdate.com, download.microsoft.com,...

### 4.3 Accès Internet

Pour la double connexion internet j'ai proposé d'utiliser CARP+pfsync plutôt que le système de l'année passée. L'intérêt c'est que les deux routeurs sont alors complètement indépendants alors que l'année passée il suffisait qu'il y en ait un des deux qui tombe pour ne plus avoir de connexion internet du tout. Néro a aussi parlé de VLAN Cisco mais dans ce cas ci j'ai pas bien compris l'intérêt. Pour donner des accès internet individuels, le système de l'année passée (authpf) semble convenir. Mais comme je

suis paranoïaque je vais quand même regarder s'il n'y a pas moyen d'utiliser un VPN en gardant la même simplicité d'utilisation.

#### 4.4 Projet DHCP

D'après Néro, les 2948 peuvent faire serveurs DHCP (mais j'ai un affreux doute: c'est pas dans la doc). Ils sont censés répondre systématiquement plus vite aux requêtes que les machines sous Windows (ça semble logique: ils reçoivent la requête avant et ils sont plus "proches" de la machine qui effectue la requête). C'est une proposition de Néro et d'après lui "en France ils font ça depuis des lustres et ça marche nickel".

#### 4.5 MAC - IP - Port Switch

Les 2948 implémentent un truc que Cisco appelle "port security". L'idée c'est que lorsque le switch "voit" une adresse MAC sur un port physique, il la note (comme tout bon switch) mais en plus il peut décider qu'il refuse automatiquement le trafic ne concernant pas cette adresse sur ce port. De plus lorsqu'il fait ça il peut envoyer le couple port physique/adresse MAC à un serveur SNMP ou syslog. Il suffit ensuite d'avoir un petit programme qui analyse les logs en live pour surveiller les conneries du genre une même adresse MAC qui apparaît sur des ports différents, un changement d'adresse MAC sur un port,... et qui prend les mesures nécessaires (bloquer le port, actionner un gyrophare+sirène dans la zone admin,...). Avec ça on peut lier assez fiablement un port physique à une adresse MAC.

Lorsque le dhcpd d'un 2948 attribue une adresse IP, il peut aussi envoyer le couple adresse MAC/adresse IP au même serveur SNMP/syslog et là le même programme que tout à l'heure met à jour une grosse table ARP qui est ensuite redistribuée à tous les serveurs et switches (et admins et joueurs paranoïaques). Avec ça on lie assez fiablement une adresse IP à une adresse MAC à un port physique pour la durée de la LAN. Aucun joueur ne peut "faire croire" à une des machines à qui la table ARP est redistribuée qu'il à l'adresse d'une autre machine. Par contre (à moins que les 2948 soient plus "intelligents" que ce que je pense, à vérifier) ça n'empêche pas un joueur de faire croire à un autre (qui n'utilise pas la table ARP) qu'il est le serveur (man in the middle, toussa) mais ça sera toujours moins pire que l'année passée.

C'est ici que ça devient intéressant. Vu qu'on a de quoi relier assez fiablement une adresse IP à une adresse MAC à un port physique, il suffit d'utiliser uniquement l'adresse IP pour identifier celui qui se connecte pour tous les services qui requièrent une authentification. Plus besoin de login/mdp du tout. Le principe serait qu'à son inscription le joueur reçoive un identifiant unique généré aléatoirement. On stock cet identifiant dans la base de données à côté du nick/nom/prénom/... Lorsque le joueur se connecte et que les étapes décrites dans les deux paragraphes précédentes (association du port physique à l'adresse MAC à l'adresse IP) ont été réalisées, il se retrouve dans une VLAN où il n'a accès qu'à un serveur web dont l'adresse est clairement indiquée sur le papier qu'il a reçu à l'entrée. Sur le serveur web il remplit un formulaire qui lui demande uniquement l'identifiant qui lui a été remis et de là on rajoute son adresse MAC/port physique/IP à côté de son nom dans la DB (+éventuellement une entrée DNS qui va bien) et il sort de la VLAN. À partir de là, toutes les applications

qui nécessitent une authentification peuvent se servir de l'adresse IP source sans rien demander de plus. Evidemment ça nécessite d'adapter les applications existantes mais même si on ne le fait pas et qu'on continue d'utiliser un login/passwd, ça permettra d'identifier directement chaque personne via son adresse IP en cas de problème.

#### **4.6 Jabber ( Gateway MSN/ICQ/...)**

Pour Jabber, il est à noter que si le serveur fait passerelle MSN/ICQ/AIM/Y!, le login/passwd est stocké directement sur le serveur Jabber. Ca veut dire que techniquement tout admin ayant accès au serveur à les moyens de récupérer le login/mdp MSN/ICQ/AIM/Y! d'une personne qui utiliserait le service. Il y a comme qui dirait un petit soucis éthique :/ (bon d'un autre côté même si c'était pas stocké on saurait le récupérer)

#### **4.7 Gestion Parc Serveur**

Comme je l'ai dit précédemment, FAI peut être très intéressant pour l'installation des serveurs mais ça va prendre un peu de temps à configurer. D'un autre côté il semblerait qu'on ait un parc de machines encore plus grand et hétérogène que l'année dernière donc je pense que ça peut vraiment être intéressant. Pour la distribution à utiliser, je crois qu'on a pas vraiment de raison de changer de Debian Woody. Il faudra juste préparer des packages "maison" pour les jeux et les noyaux qui vont bien.

#### **4.8 BoB, ça va ?**

Je sais pas si Bob était dans son état normal et s'il a tout compris mais il a plus ou moins approuvé tout ça :)