Master Informatique spécialité TIIR

Projet de Fin d'Etudes

Déploiement d'une infrastructure pour enregistrer le trafic réseau de la salle TIIR

Auteurs :
M. Rémy Debue
M. Kévin Lanvin

Encadrants :
Julien Iguchi-Catigny

Version 0.1 du
5 février 2017

Remerciements

Nous souhaitons tout d'abord remercier Julien Iguchy-Cartigny, notre tuteur de projet qui nous a accompagné tout au long du PFE. Nos remerciements s'adressent également aux administrateurs du M5 qui nous ont acceuillis dans leur salle et dans la zone serveur et aidé à installer la nouvelle passerelle. Nous adressons nos remerciements également aux étudiants de la salle TIIR, pour leur patiente lors des coupures réseaux pendant que nous le configurions. Enfin, nous remercions les administrateurs du CRI qui se sont déplacés pour discuter de notre projet et de la nouvelle configuration du réseau.

Table des matières

Table des figures

Liste des sigles et acronymes

API	*Application Programming Interface*
CRI	*Centre de Ressources Informatique*
DHCP	*Dynamic Host Configuration Protocol*
DNS	*Domain Name Server*
ELK	*Elasticsearch Logstash Kibana*
IP	*Internet Protocol*
MOCAD	*MOdèles Complexes, Algorithmes et Données*
PFE	*Projet de Fin d'Etudes*
TIIR	*TIC, Infrastructures, Intégrité et Répartition*
VPN	*Virtual Private Network*
VPS	*Virtual Private Server*

Introduction

Le sujet "journalisation du trafic de la salle TIIR" trouve sa source dans le besoin d'autonomie des étudiants. En effet, Julien Iguchi-Cartigny, créateur de ce projet, souhaitait que l'on puisse s'affranchir du serveur proxy de l'université, afin de laisser plus de libertés pour la réalisation des travaux pratiques. Le proxy, serveur par lequel passait l'ensemble du trafic de la salle, permettait de filtrer les différentes connexions et ainsi empêcher les abus. Cependant, un tel filtrage ne peut pas être parfait, et certaines transmissions de données utiles dans le cadre des cours se trouvaient alors bloquées. Remplacer cette solution par une journalisation du trafic supprime ces contraintes pour les enseignants, mais permet de retrouver l'étudiant responsable en cas de comportement illicite (attaques sur le réseau d'entreprises, consultation de sites non-autorisés).

Ce projet de fin d'études se découpe alors en plusieurs parties. Tout d'abord, l'architecture du réseau de la salle TIIR a du être repensée. Il a fallu gérer l'installation des machines utiles pour notre projet, l'accès à internet de la salle et la répartition des adresses IP des postes de travail. Ensuite, un système a été mis en place pour enregistrer le trafic. La sauvegarde des données nécessitait un choix de technologie légère pour ne pas perturber le trafic. Les fichiers journaux sont envoyés vers une machine dédiée à leur indexation et leur analyse. Cette dernière proposera également une interface web permettant de visualiser les données plus rapidement.

Chapitre 1

Architecture

1.1 Analyse de l'existant

A notre arrivée, le réseau de la salle TIIR était organisé autour d'un proxy. Toutes les transmissions passaient à travers ce proxy, avant d'être chiffrées en utilisant IPSec. Ce système présentait plusieurs avantages. Le premier est la mise à disposition des administrateur d'un moyen de contrôler le trafic avec des règles de filtrage appliquées sur le proxy. Les activités illicites étaient ainsi rapidement détectées et bloquées automatiquement. Le second point positif est le fait que grâce à ce proxy, tout le trafic externe semblait provenir d'une seule et même machine, puisque l'adresse IP source de chaque paquet était l'adresse IP du proxy. L'inscription à de nouveaux services, comme l'accès à une bibliothèque digitale, en est ainsi facilité. Le chiffrement du trafic avec IPsec empêchait un quelconque vol des données sortant de l'université.

Néanmoins cette solution fût remise en question. Tout d'abord, l'absence de documentation sur cette architecture posait problème. Les technologies utilisées (le proxy et IPsec) furent laborieuses à mettre en place, et la maintenance s'annonçait ardue sans schéma ou fichiers de configuration à disposition. La notion de proxy posait également problème. En effet, le trafic était automatiquement filtré. Aucune confiance n'était accordée aux étudiants, et leurs marges de manoeuvre, lorsqu'ils souhaitaient sortir des sentiers battus était plus que limitée. Impossible de lancer la commande nmap qui permet entre autre de scanner les ports d'un poste, dans un tel réseau.

Une nouvelle solution a donc été envisagée : la journalisation du trafic. après la suppression du proxy, la liberté des étudiants augmenterait considérablement. Le trafic serait chiffré avant même d'arriver sur les postes des administrateurs de l'université, afin de les empêcher d'analyser les communications. Néanmoins la sécurité pénale de l'encadrant serait assurée grâce aux journaux que l'on pourrait consulter en cas de problème grave. De plus, cette structure pourrait servir de base pour des projets MOCAD. En effet le nombre

important de communications enregistrées leur permettrait de s'entraîner à l'extraction d'informations dans de grands volumes de données.

1.2 Conception de la nouvelle architecture

1.2.1 Accès à Internet

Pour avoir accès à Internet depuis la salle TIIR, il faut mettre en place une passerelle. Cette passerelle est reliée d'un côté à la salle TIIR via un switch. Elle possède une adresse IP privée qui lui permet de communiquer avec tout le sous-réseau. De l'autre côté, cette passerelle possède une adresse publique. Cela signifie qu'elle peut communiquer sur Internet. Néanmoins, la topologie du réseau nous oblige à relier cette passerelle aux machines du CRI avant d'accéder à Internet (voir figure 1.1).

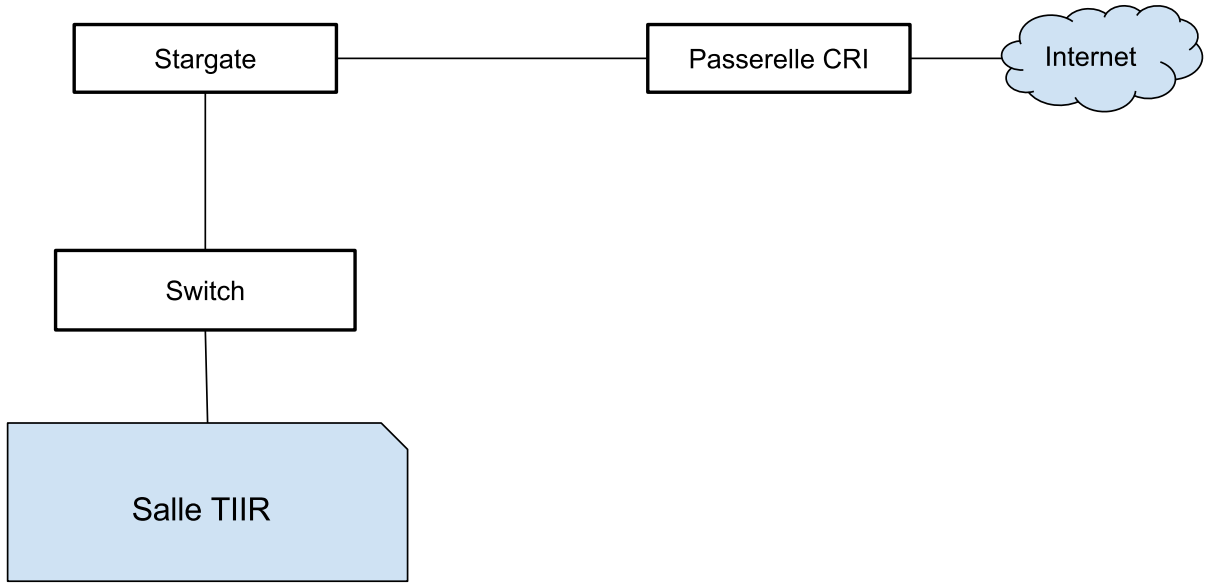

FIGURE 1.1 – Installation de la passerelle : Stargate

A ce stade, l'accès au réseau mondial est garanti, mais il reste deux problèmes. Le premier est la présence des machines du CRI sur la route des paquets sortants. Les administrateurs peuvent facilement lire, et bloquer les communications en provenance de la salle TIIR. Le second est que tout le trafic sortant a pour adresse IP la machine du CRI. Cela signifie que si un étudiant a un comportement illicite sur le réseau, l'université peut en être tenue pour responsable.

Pour résoudre ces problèmes, la solution retenue fût d'ajouter un VPS. Cela fixe une

adresse IP source de toutes les communication sortantes différentes de celle de l'université. Celle-ci n'est donc plus directement l'émettrice des paquets interdits. Cela permet également de créer un tunnel sécurisé entre notre passerelle et le VPS (voir figure 1.2). Ainsi, les communications que le CRI retransmet sont chiffrées. Les administrateurs ne peuvent donc plus surveiller le trafic sortant de la salle TIIR, et encore moins le filtrer.

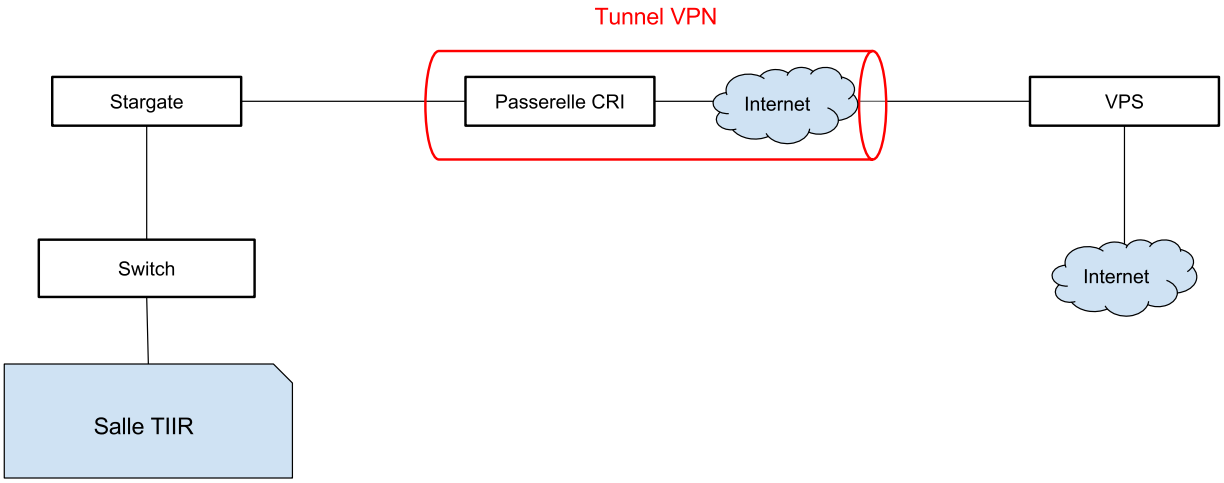

FIGURE 1.2 – Mise en place d'un tunnel sécurisé

L'accès à Internet est maintenant garanti. Le CRI ne peut plus surveiller les communication et l'adresse de sortie n'est plus une adresse appartenant à l'université. Cette solution remplit donc les objectifs fixés auparavant.

1.2.2 Architecture pour l'enregistrement des données

Comme nous l'avons vu plus haut, le trafic de la salle TIIR dans son intégralité passe par la passerelle. C'est donc sur cette machine que l'enregistrement des données doit s'effectuer. Cependant, l'indexation et l'enregistrement des données consomme beaucoup de ressources. C'est pourquoi nous avons décidé d'utiliser une seconde machine pour effectuer cette tâche. Cette dernière fait partie du même sous-réseau que la salle TIIR. Ainsi, une fois les données analysées, un dashboard disponible sur une interface web permettra à tous les étudiants de constater certaines métriques sur les données transmises (voir figure 1.3).

Cette architecture est la dernière itération de notre raisonnement. Elle permet de contourner tous les problèmes inhérents à l'ancienne solution. Le CRI ne peut plus voir ni filtrer les paquets des étudiants, l'adresse de sortie n'est pas une adresse de l'université, l'enregistrement des données permet de se couvrir juridiquement en cas de problème, et enfin la séparation entre la machine qui enregistre les données et celle qui les traite permet un gain de performances non négligeable.

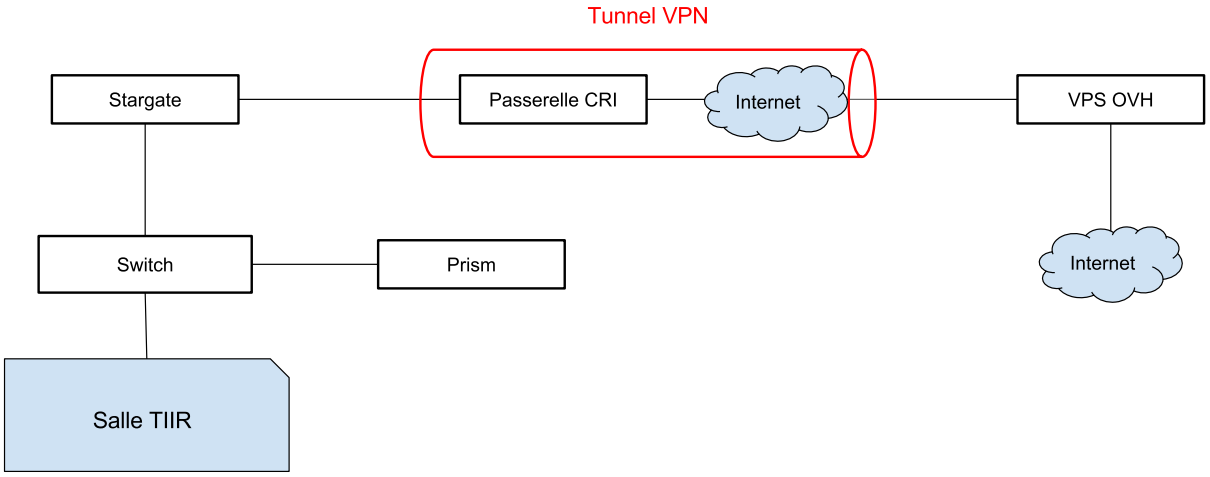

FIGURE 1.3 – Ajout d'une machine de traitement des journaux : Prism

1.3 Implémentation de la nouvelle architecture

Nous allons présenter ici l'implémentation de l'architecture, qui permettra à tous les étudiants d'accéder à Internet, mais également à la machine qui traite les données d'être connectée à la passerelle.

1.3.1 Accès à Internet

La première étape pour garantir l'accès à Internet fût d'installer la passerelle Stargate. Nous avons utilisé un serveur disponible dans le bâtiment, sur lequel nous avons installé un Debian, un système d'exploitation Linux libre. Nous l'avons ensuite relié à la salle TIIR pour le configurer à distance. Le défi ici était d'installer cette machine en garantissant une interruption du service minimale. Nous avons donc conçu le plan d'adressage, le DHCP, le DNS et le tunnel VPN avant d'effectuer ce changement. Le VPS que nous avons loué appartient à OVH. Nous avons choisi ce prestataire car il est l'un des partenaires de l'université.

Le plan d'adressage

La passerelle possède deux interfaces. La première est reliée à la salle TIIR, et la seconde est reliée indirectement au VPS. Pour la première, le choix de l'adressage a été assez simple. En effet, la salle TIIR appartenait déjà à un sous-réseau local : 10.1.1.0/24. La seconde est reliée à la passerelle du CRI. Une adresse IP lui a été distribuée : 134.206.225.12/32. Il était impératif dans notre configuration de faire passer le trafic via la passerelle du CRI

qui avait pour adresse : 134.206.225.1/32. Cette dernière retransmettait alors les paquets vers le VPS OVH à l'adresse 137.74.40.238/32. La figure 1.4 résume ces adresses. Les modifications d'adresses sont faites dans le fichier /etc/network/interfaces qui permet de donner une adresse à une interface réseau, ainsi que définir sa passerelle par défaut.

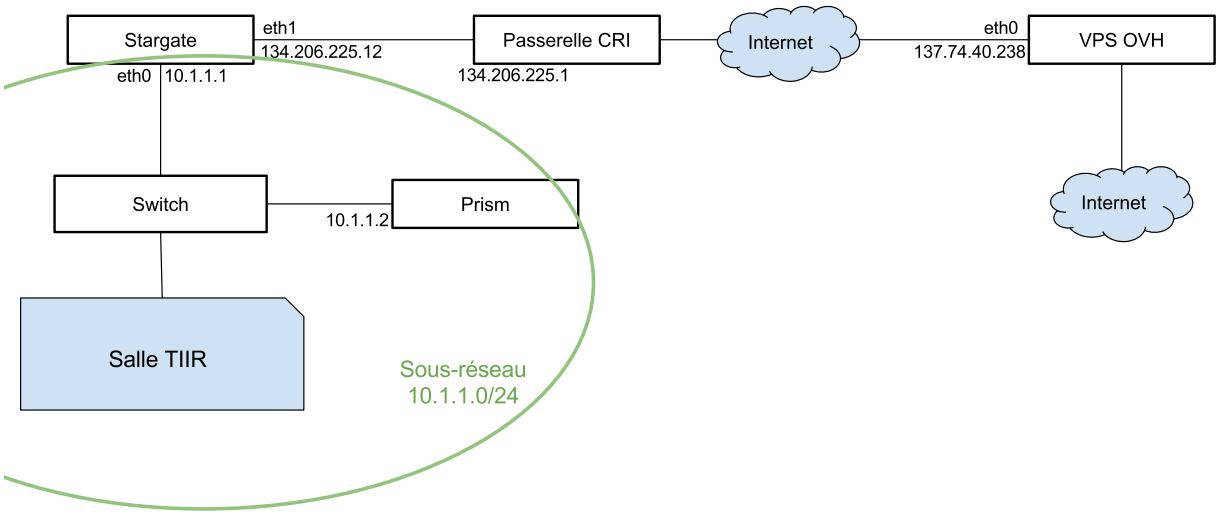

FIGURE 1.4 – Plan d'adressage du réseau

Le serveur DHCP et le serveur DNS

Avant de placer notre passerelle à la place de l'existante, certaines modifications restaient à effectuer pour garantir l'accès à Internet des utilisateurs de la salle TIIR. Tout d'abord, les adresses des postes de la salle ne sont pas fixes. Elles appartiennent toutes au sous-réseau 10.1.1.0/24 mais sont distribuées à la demande par un serveur DHCP qui se trouve sur la passerelle en service. Il faut également installer un serveur DNS qui permettra de résoudre les noms de domaines demandés par les étudiants, c'est à dire transformer les URL en adresses IP. S'il ne connait pas le domaine demandé, le serveur DNS devra rediriger la requête vers les serveurs DNS fournis par OVH.

Pour implémenter cette solution, nous avons choisi d'utiliser dnsmasq. Cet outil permet en effet de regrouper les fonctions de DNS et de DHCP. La configuration choisie distribue des adresses entre 10.1.1.100 et 10.1.1.199. Cela permet de s'assurer une plage d'adresses inutilisées en cas d'ajout de machines qui nécessiteraient une adresse IP statique.

Le tunnel VPN

Afin d'éviter que le CRI ne filtre le trafic de la salle TIIR, celui-ci est chiffré par Stargate et déchiffré ensuite par le VPS OVH. Pour réaliser ce chiffrement, nous avons utilisé un

tunnel VPN en point à point. Quand un étudiant émet un paquet depuis la salle TIIR, il arrive sur l'interface privée de la passerelle Stargate. Les règles de routage redirigent ensuite ce paquet vers une interface virtuelle (ici tun0). Cette interface est en fait reliée au programme OpenVPN qui chiffre le contenu du paquet. Puis, une seconde règle de routage redirige alors le paquet chiffré vers l'interface publique de la passerelle Stargate, qui expédie les données vers le CRI sur le port 1194, le port par défaut pour les tunnels VPN. Nous avons demandé au CRI de retransmettre ces paquets vers le VPS OVH qui va effectuer le processus inverse. Les données seront redirigées vers une interface virtuelle reliée à OpenVPN, qui va déchiffrer le contenu des paquets et les renvoyer vers Internet. Les mêmes manipulations sont réalisées dans l'autre sens lorsque le serveur contacté par l'utilisateur répond. Les données sont chiffrées avec une clef secrète générée au préalable et transmise au VPS via un canal sécurisé. La figure 1.5 illustre ce principe.

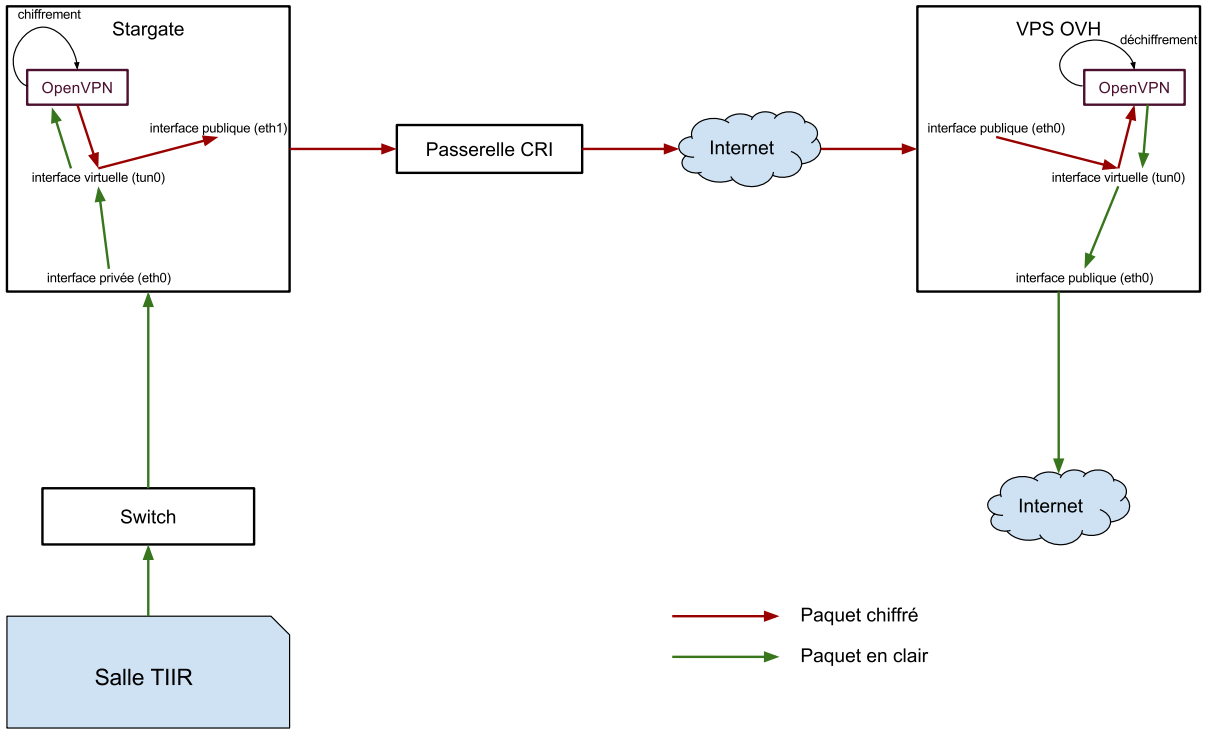

FIGURE 1.5 – Fonctionnement du tunnel VPN

1.3.2 Enregistrement des données

Afin d'enregistrer les données, nous avons ajouté une seconde machine au sous-réseau de la salle TIIR : Prism. Cette machine devra avoir une adresse IP statique pour être facilement accessible depuis le sous-réseau. Nous lui avons attribué l'adresse 10.1.1.2/32. Cette machine a été reformatée. Nous avons également installé un Debian. Cette machine a deux rôles. Elle doit indexer et enregistrer les données concernant le trafic de la salle TIIR mais également proposer une interface web pour faciliter la lecture de ces données. La figure 1.6 montre l'architecture finale du réseau.

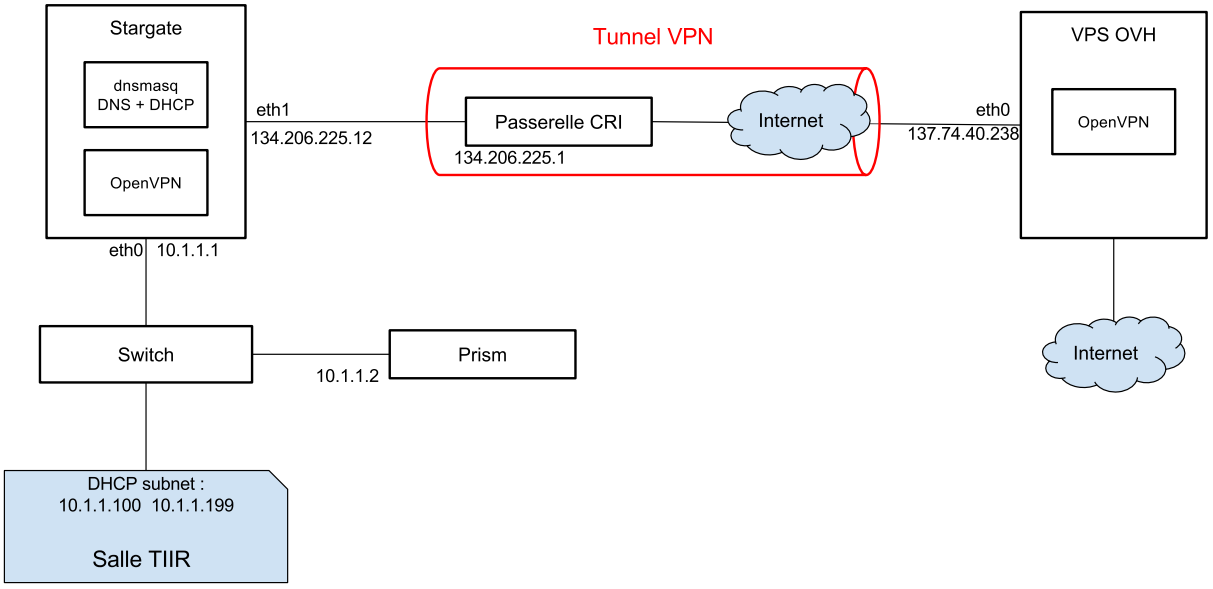

FIGURE 1.6 – Architecture du réseau de la salle TIIR

Chapitre 2

Enregistrement des données

2.1 Pourquoi journaliser ?

L'enregistrement des données circulant sur le réseau correspond à plusieurs besoins. En effet, avec la nouvelle architecture, les étudiants sont libres de faire ce que bon leur semble. Il devient alors indispensable d'être capable de connaître qui est l'auteur d'une action illicite sur le réseau si cela se produit. D'un point de vue légal, l'administrateur du réseau est responsable de ce qui en sort. Afin de se couvrir pénalement, M. Iguchi-Cartigny (alias Kartoche) n'a pas d'autre choix que de garder l'historique des connexions. Cela peut servir également un but statistique. En effet, avec de tels volumes de données, on peut voir les sites préférés des étudiants, ou encore analyser leur sérieux et leur productivité en fonction de l'heure, ou de l'enseignant présent dans la salle. Cela pourrait donc permettre d'éventuellement un jour peut-être améliorer même juste un peu la qualité des enseignements dispensés. Enfin, des volumes de données aussi importants peuvent servir d'entraînement pour le Master MOCAD ou encore pour les cours de calculs distribués en TIIR.

2.2 Les outils utilisés

2.2.1 Bro

Pour mettre en place la capture de données, plusieurs solutions ont été envisagées. Nous avions commencé à enregistrer les paquets en utilisant les iptables. Mais cette solution n'était pas satisfaisante car les fichiers journaux étaient trop gros et le filtrage pas assez fin. Nous nous sommes donc penchés vers des solutions plus abouties. Les détecteurs d'intrusion sont des logiciels parfaits pour notre usage. En effet ils analysent le trafic en temps réel et possèdent une fonctionnalité d'enregistrement des données analysées. Celui que nous

avons utilisé s'appelle Bro. Il est moins documenté que son concurrent Snort, mais ses performances ne sont pas affectées par le nombre d'intrusions détectées au préalable. Ils nous permet, via des scripts, de choisir le degré d'inspection des paquets. On peut par exemple grâce à lui récupérer le contenu d'un mail si ce dernier n'était pas chiffré.

2.2.2 La pile ELK

Logstash

FIGURE 2.1 – L'outil d'analyse de logs Logstash

Logstash est un outil de récupération et d'analyse de fichiers journaux. Il permet de récupérer des fichiers formatés de façon différente et de les organiser sous forme de paires clef/valeur. Il peut notamment formater les dates sous un format standard ou reconnaître des numéros de version grâce à des Regex. Une fois la transformation des fichiers terminée, il renvoie le fichier vers Elasticsearch

Elasticsearch

FIGURE 2.2 – La base de données distribuée Elasticsearch

Elasticsearch est une base de données NoSQL orientée vers le Big Data. Elle fonctionne comme un cluster, c'est à dire un regroupement de noeuds. Chaque noeud est une instance d'Elsaticsearch et la base de données fonctionne de façon distribuée. Cela lui permet de gérer de très grands volumes de données et assure une évolutivité importante. Elasticsearch est basé sur un framework Apache Lucene, ce qui lui permet de répondre à des requêtes sur du texte dans un temps minime.

Kibana

FIGURE 2.3 – L'interface web et moteur de recherche Kibana

Kibana est le dernier outil utilisé pour le traitement des données. C'est une interface web qui permet de requêter Elasticsearch pour en extraire les informations souhaitées. Ces informations peuvent alors facilement être affichées en divers graphiques directement grâce à Kibana. Cela permet une lecture très intuitive de l'enregistrement des données. Le tableau de bord personnalisable permet d'avoir en permanence un aperçu sur les données qui sont enregistrées comme sur la figure 2.4. Dans notre cas, il permettrait par exemple d'afficher les différents sites web les plus consultés pendant la dernière heure par les étudiants.

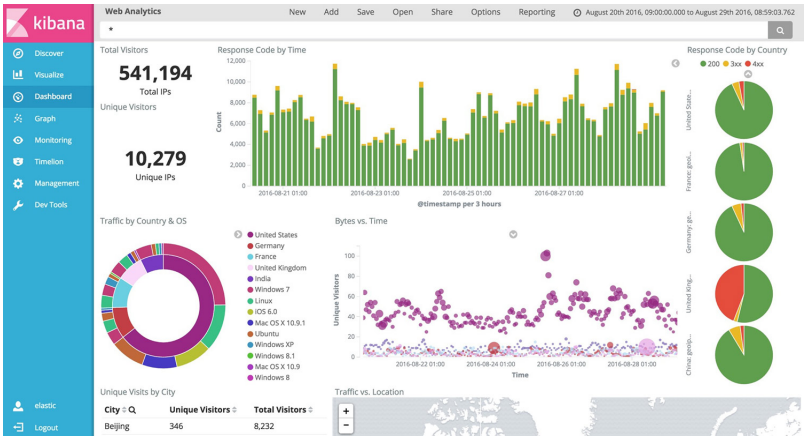

FIGURE 2.4 – Exemple de tableau de bord Kibana

2.2.3 Docker

Docker est un système de virtualisation qui fonctionne par conteneurs. Son principal avantage est d'isoler les applications lancées dans ses conteneurs. Son utilisation dans le projet se justifie pour augmenter la tolérance aux pannes. En effet, nous ne souhaitons pas qu'une panne de Bro sur la passerelle Stargate nous oblige à redémarrer la machine. Cela engendrerait une coupure Internet indésirable pour les utilisateurs de la salle TIIR. Docker est donc une solution qui permet de ne pas impacter le reste du système en cas

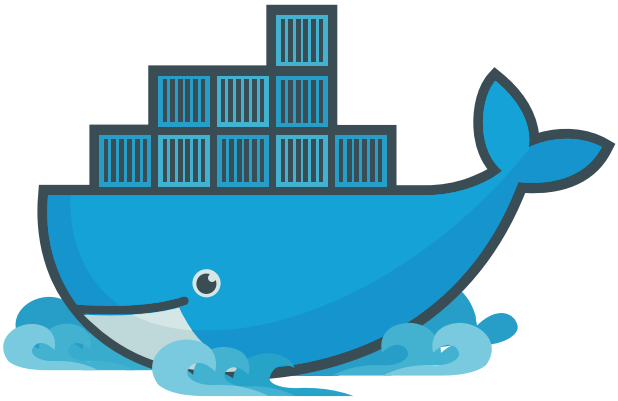

FIGURE 2.5 – Le système de virtualisation Docker

de dysfonctionnement d'un des conteneurs. Les applications lancées dans Docker sont Bro, Elasticsearch, Kibana et Logstash. Si l'un d'entre eux s'arrête il sera toujours possible d'utiliser les services des autres conteneurs. Si Logstash s'arrête, l'interface web Kibana fonctionnera toujours par exemple. La figure 2.6 montre le système et les différentes applications installées sur chacune des machines.

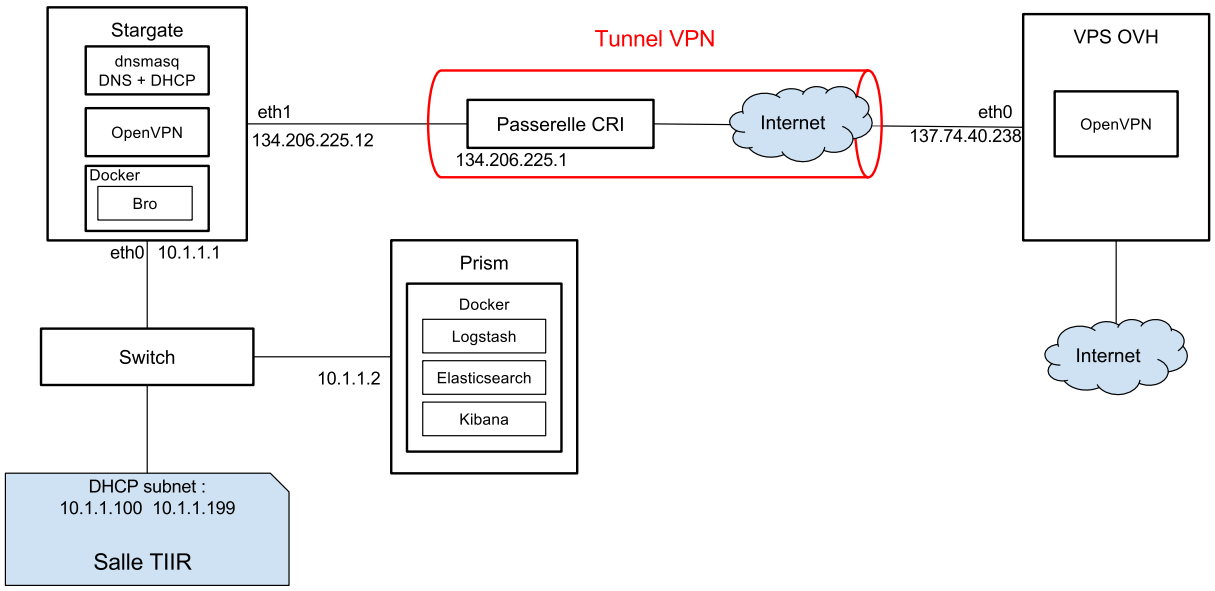

FIGURE 2.6 – Schéma du projet final

Conclusion et perspectives

Ce projet nous a fait travailler sur des technologies innovantes. Il nous a permis de concevoir une architecture fonctionnelle, puis de nous confronter aux difficultés techniques de l'implémentation de nos solutions. Le résultat final est fonctionnel mais certains axes d'amélioration peuvent être approfondis. La machine Prism notamment expose l'API Elasticsearch, ce qui permet à un étudiant malveillant de supprimer les fichiers journaux s'il le souhaite. Une attaque par déni de service sur la Passerelle peut également empêcher certaines données d'être enregistrées. Néanmoins, notre contribution n'est que la première étape d'un projet qui vise à augmenter l'autonomie des étudiants tout en garantissant une relative sécurité pénale pour l'encadrant. Notre travail pourra dans les années qui suivent être exploité pour continuer de sécuriser ce système.

Université de Lille 1
Cité Scientifique
59650 Villeneuve d'Ascq France