ERM - UCL - ULB - UNamur - HE2B - HELB

Master en cybersecurité à finalité analyse et conception de systèmes

Année académique 2018 - 2019

Analyse et mise en place d'un simulateur et d'un hyperviseur incluant des scénarios d'entrainement destinés aux étudiants du master en cybersécurité

Pierre Michaux

Promoteur : Prof. Jean-Michel Dricot

Mémoire

"*Passwords are like underwear : don't let people see it, change it very often, and you shouldn't share it with strangers.*"

Chris Pirillo

"*It takes 20 years to build a reputation and few minutes of cyber-incident to ruin it.*"

Stephane Nappo

Avant toute chose, je tiens à remercier mon promoteur de mémoire, le PhD. Jean Michel Dricot, pour son suivi et son aide dans la réalisation de ce mémoire de fin d'étude.

Le professeur Thibault Debatty pour ses conseils relatifs à la phase de recherche et à la présentation.

Xavier et Chloé Guérit pour la relecture orthographique.

Ma famille pour le soutien et les encouragements.

Résumé

A l'heure actuelle, on peut affirmer que la technologie évolue très rapidement. Une course entre les Géants du monde numérique est en cours, visant à trouver les solutions les plus performantes. Il suffit de jeter un rapide coup d'oeil sur l'évolution massive des technologies ces vingt dernières années pour comprendre les enjeux auxquels notre société doit faire face. Les solutions et applications créées ne sont pas toujours pensées en privilégiant l'aspect sécurité. La sécurité dont il faut tenir compte lors de l'élaboration de nouvelles technologies (aussi bien au niveau applicatif que physique) a un rôle essentiel : protéger la vie privée de l'utilisateur.

Ce mémoire vise à mettre en place une plateforme d'émulation hybride sur laquelle les étudiants pourront s'entrainer en vue d'acquérir les compétences nécessaires au titre de master en cybersécurité. La plateforme d'émulation se veut hybride pour permettre l'intégration d'éléments qui ne sont pas virtualisables. Cette technologie se veut moderne, flexible et innovante. Différentes techniques de virtualisation sont étudiées pour permettre un comparatif entre ces différents systèmes.

La deuxième partie de ce mémoire porte sur l'élaboration de scénarios qui seront remis aux étudiants lors des travaux pratiques. Ces scénarios sont exécutés sur la solution de virtualisation choisie précédemment.

La majorité des annexes de ce mémoire contient les scénarios créés, de la documentation destinée à l'administrateur système du laboratoire et des vues des différents systèmes.

Table des matières

Table des figures

Liste des tableaux

Acronymes

CML Cisco Modeling Labs. 6

ERM Ecole Royale Militaire. v, 6, 7

GNS3 Graphical Network Simulator. 22, 40

IOS Internetwork Operating System. 6, 21

KVM Kernel-based Virtual Machine. 18, 40

OS Operating System. 6–8

PC Personal Computer. 22
PLC Programmable Logic Controller. 11

RAM Random Access Memory. 7, 26
RDP Remote Desktop Protocol. 7

SCADA Supervisory Control And Data Acquisition. 11

VIRL Virtual Internet Routing Lab. 5, 6
VM Virtual Machine. 26, 41
VNC Virtual Network Computing. 7

Glossaire

hardware Hardware : matériel informatique, pièce détachée d'un appareil informatique. 10, 22

LUN LUN, de l'anglais Logical Unit Number : désigne, dans le domaine du stockage informatique, le numéro d'unité logique d'un équipement SCSI. 38

MTU MTU : taille maximale d'un paquet pouvant être transmis en une seule fois sans être fragmenté.. 38

NAS Network Attached Storage : serveur de fichiers autonomes où les fichiers permettent d'être stockés et partagés sur le réseau. 36, 38, 39

pentesting Pentesting (test d'intrusion) : test dont le but est d'évaluer la sécurité d'un système informatique. 4

Chapitre 1

Introduction

1.1 Objectifs du mémoire

Ce mémoire a pour objectif la mise en place d'une plateforme d'émulation hybride et la création de scénarios relatifs à la cybersécurité. Les scénarios ont pour but d'apporter à l'étudiant un entrainement qualitatif en utilisant des vulnérabilités connues dans le monde professionnel.

La solution choisie se veut hybride pour pouvoir intégrer par la suite des composants qui ne sont pas virtualisables. On peut imaginer des scénarios d'attaque sur des systèmes industriels (SCADA). Les équipements virtualisés doivent pouvoir interagir avec les équipements physiques, d'où la notion hybride.

Des contraintes ont été identifiées dès le début de la phase de recherche. Elles sont détaillées au point 1.4.

Ce mémoire a donc pour but de mettre en place un simulateur choisi sur base de critères et de contraintes, tout en élaborant des scénarios destinés à l'entrainement des étudiants.

1.2 Structure du mémoire

Plusieurs sections composent ce mémoire. La phase d'introduction a pour but de présenter la problématique. La phase de recherche, ou *Etat de l'Art*, répertorie les solutions existantes et les études qui ont déjà été faites sur le sujet.

La phase d'implémentation est relative à l'étude des différentes solutions qui ont été retenues. Cette étude est basée sur le résultat de critères établis préalablement. Les résultats sont interprétés pour en déduire la solution qui correspond au mieux aux besoins. Elle intègre en réalité deux composants qui ont été étudiés : l'hyperviseur et le simulateur. Ces deux composants sont analysés dans la section implémentation.

Après la mise en place de l'hyperviseur et du simulateur, les scénarios ont été imaginés en fonction des besoins professionnels réels. Des instances ont été créées et sont prêtes au déploiement.

Les scénarios créés et la documentation destinés aux étudiants, assistants, administrateur système et professeur se retrouve dans la partie annexe du mémoire. Cette documentation a pour but de faciliter le management de la solution déployée.

Certaines annexes sont volontairement rédigées en anglais de manière à pouvoir être consultées par tout public.

1.3 Contribution

A ce jour, il n'existe pas de comparatif entre Hynesim et GNS3. Aucune étude liée aux performances de simulateurs sur des hyperviseurs de couche différente n'existe.

En ce sens, la contribution de ce travail est basée d'une part sur le comparatif entre les simulateurs, et d'autre part, sur l'observation des résultats liés aux performances de l'hyperviseur.

Une seconde contribution est la création des scénarios pour les étudiants du master en cybersécurité. Aucune manipulation, basée sur les techniques mises en place dans ce mémoire, n'existe pour le cours de *Network Security*. D'un point de vue personnel, il est intéressant que les manipulations soient pensées par un étudiant en cybersécurité en fin de cursus qui a eu la chance de goûter aux difficultés et aux enjeux du monde professionnel, de par l'expérience acquise lors de ses stages ou grâce aux compétitions auxquels il a pu participer dans son master (Cybersecurity Challenge Belgium, CTF Tournament Ph. Jérôme Dossogne, Proximus Arena...).

1.4 Contraintes

Des contraintes ont été identifiées au début de la phase de recherche. La recherche d'une solution optimale s'est faite en prenant en compte les critères suivants :

- Innovation : la solution doit se montrer innovante. Il faut qu'elle améliore la solution existante.
- Hybride : la solution doit pouvoir accueilir des équipements qui ne peuvent être virtualisés. Elle doit permettre une interconnexion parfaite entre des périphériques physiques et des périphériques virtualisés.
- Durable : la durabilité est un critère important. La solution doit pouvoir tenir dans le temps et ne pas devenir obsolète. Elle doit pouvoir bénéficier de mises à jour et d'améliorations. L'idéal étant qu'elle ait un support et des mises à jour

régulières.
- Performance et flexibilité : la solution doit être performante et flexible. Elle doit, par exemple, répondre aaux besoin d'un étudiant qui ne disposerait pas d'une puissance de calcul suffisante lors des travaux pratiques.
- Portabilité : la solution doit être portable. L'étudiant doit pouvoir exporter le projet sur lequel il travaille afin de le réimporter sur une autre machine de manière facile.
- Professionnelle : la solution doit être professionnelle.

Les contraintes listées ci-dessus constituent une première approche de la solution qu'il faut déterminer.

Chapitre 2

Etat de l'art

Le choix des technologies utilisées pour la mise en place d'un simulateur hybride, repose sur différentes étapes.

La première étape est de regarder les simulateurs existant sur le marché. Les différentes solutions disponibles sont comparées sur base de leurs performances.

Un autre point (non négligeable) consiste à répertorier les ressources disponibles qui peuvent être utilisées pour faire du pentesting.

Pour terminer, l'aspect didactique utilisé lors de la création des manipulations doit être étudié afin de proposer des scénarios qui soient les plus pertinents possibles. Un scénario sera pertinent si les objectifs et les compétences que l'étudiant doit acquérir sont clairement définis au début de sa conception. Il est évident qu'un scénario mal conçu ou qu'une matière mal comprise conduiront inévitablement l'étudiant a négliger de potentiels vecteurs d'attaque.

2.1 Simulateurs

Les simulateurs retenus pour créer un environnement d'entrainement sont *Hynesim, GNS3, VIRL et la solution proposée par l'Ecole Royale Militaire.*

2.1.1 Hynesim

Hynesim est la solution qui est actuellement déployée pour les manipulations en laboratoire. Hynesim, pour Hybrid Network Simulation, est une plateforme qui permet de simuler des topologies réseaux. La solution est proposée par Diateam, société française spécialisée dans la cybersécurité et l'ingénieurie numérique [1].

1. Informations disponibles sur site Diateam : `https://www.diateam.net/`

Hynesim utilise un contrôleur développé également par Diateam, connu sous le nom de Hyneview. Hyneview est mis à disposition des utilisateurs d'Hynesim sous forme d'une machine virtuelle classique Linux. Il est impossible d'utiliser Hynesim sans le contrôleur Hyneview.

Hynesim est proposée sous différentes versions : une version gratuite et deux versions payantes. La version gratuite peut être installée en suivant le guide d'installation disponible sur leur site web alors que la solution payante a l'avantage d'être livrée avec une entité physique prête à l'emploi. Le support n'est évidemment pas compris dans la version gratuite [2].

2.1.2 GNS3 [3]

GNS3 est une autre solution d'émulation comparable à Hynesim. GNS3 permet la simulation de réseaux informatiques virtuels ou en interaction avec un réseau physique existant. La solution est donc également hybride.

GNS3 a été créé il y a un peu plus d'une dizaine d'années. Solution gratuite et utilisée par de nombreuses personnes (tant dans le domaine public que dans le domaine privé), son but premier était l'émulation d'équipement Cisco en utilisant de vraies images. La plateforme a cependant bien évolué puisqu'elle permet à l'heure actuelle l'intégration de divers équipements en plus des ressources Cisco.

Son caractère open source et entièrement gratuit en fait une solution très intéressante pour un usage académique. Elle est continuellement soutenue et améliorée par une communauté de plus de 800.000 membres à travers le monde.

GNS3 est disponible sous deux options. Soit on utilise la solution sur une (seule) machine locale, soit on télécharge une machine virtuelle qui jouera le rôle de serveur sur lequel viendra se connecter un client GNS3.

Le support de cette solution, à la différence d'Hynesim, est basé sur la communauté qui soutient le projet puisque la solution est open source.

2.1.3 VIRL

Virtual Internet Routing Lab (VIRL) est une solution directement proposée par la firme Cisco pour la simulation de réseaux.

La solution possède beaucoup de points communs à Hynesim et GNS3 mais un comparatif fiable ne peut être établi. VIRL, à la différence des deux solutions précédentes, est une solution qui est orientée vers le cloud. Un comparatif entre une solution cloud

2. Informations disponibles sur le site Hynesim : `https://www.hynesim.org/`
3. Informations disponibles sur le site de GNS3 : `https://gns3.com/`

et une solution locale est extrêmement difficile à faire puisqu'il n'y a pas d'accès direct aux systèmes. On peut néanmoins comparer quelques fonctionnalités proposées par cette solution.

L'avantage d'utiliser VIRL réside dans le fait d'utiliser des images légales dans la création de topologies. Toutes les ressources qui sont proposées dans VIRL sont des ressources officielles sous licence. Les images sont attribuées à l'utilisateur de manière légale.

Malheureusement, cette solution n'est pas disponible gratuitement. L'utilisateur doit souscrire à un abonnement pour utiliser la plateforme. Aucune version d'essai n'est proposée.

D'après la documentation disponible sur le site de VIRL, une interconnexion avec un réseau hybride n'est pas possible. Il n'est malheureusement pas possible de déterminer avec précision comment la plateforme fonctionne puisqu'aucune version d'essai ne peut être octroyée.

D'un point de vue marketing, Cisco déconseille de s'entraîner sur d'autres simulateurs qui utilisent des Internetwork Operating System (IOS) obtenus de manière illégale. Pour cause, ceux-ci ne seraient pas tenus à jour et ne seraient donc pas munis des dernières fonctionnalités disponibles.

Il existe une variante de VIRL disponible pour les entreprises, plus connue sous le nom de Cisco Modeling Labs (CML).

Un autre avantage d'utiliser VIRL est l'accès à un service de support compris dans l'abonnement. La solution peut être installée sur n'importe quel Operating System (OS) puisque la simulation des ressources s'effectue dans le cloud.

En raison des différents points énoncés ci-dessus, cette solution ne sera pas retenue dans le comparatif des simulateurs. Pour cause, la solution n'est pas hybride et un abonnement est requis.

2.1.4 Ecole Royale Militaire[9]

La solution proposée par l'ERM repose sur le concept de *Cyber Range*. Le *Cyber Range* est utilisé pour l'entrainement individuel et collectif. Il s'agit d'un outil qui permet de simuler un réseau. Ce réseau est utilisé à des fins d'entrainement. Cette solution rend possible la virtualisation de gros réseaux (souvent complexes) pour améliorer le réalisme et la qualité de la simulation.

L'implémentation propre de l'ERM d'un *Cyber Range* à des fins d'entrainement regroupe de nombreux avantages :

— Le scénario de l'exercice est décrit dans un format texte (comme yaml ou json).

Le fait d'utiliser un format texte permet de faciliter le contrôle de version, la détection de changement et l'échange.
— La définition d'un scénario autorise un nombre flexible de participants. Ainsi, le même scénario utilisé pour dix personnes peut être réutilisé pour cent personnes.
— Il est possible d'utiliser directement des images Vagrant comme machines virtuelles, ce qui donne accès à des miliers d'images prêtes à l'emploi.
— Une configuration complète des machines virtuelles (nombre de vCPU, interfaces réseaux, Random Access Memory (RAM), OS, *hostname*...).

Les principaux composants du *Cyber Range* de l'ERM regroupent un hyperviseur, une passerelle de bureau d'accès à distance et un orchestrateur.

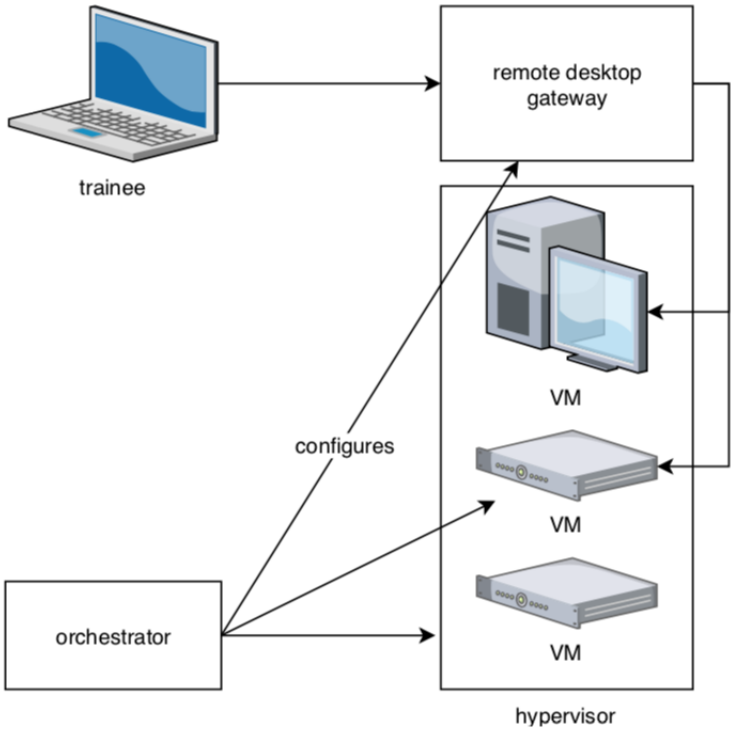


FIGURE 2.1 – Elements du *Cyber Range* de l'ERM.
Source : Thibault Debatty et Wim Mees [9]

L'hyperviseur exécute les machines virtuelles et gère les réseaux. Actuellement, la solution propose d'utiliser VirtualBox. L'intégration d'autres technologies de virtualisation en utilisant la même technique d'organisation est possible mais est encore à l'étude. L'hyperviseur doit absolument autoriser la connexion en bureau d'accès à distance pour accéder aux machines virtuelles.

La passerelle de bureau d'accès à distance autorise l'utilisateur à se connecter au *Cyber Range* et à utiliser les machines virtuelles. L'accès se fait en utilisant un navigateur. Actuellement, le serveur web utilise *Apache Guacamole* qui est basé sur HTML5. Aucun autre plugin n'est requis. Virtual Network Computing (VNC) et le Remote Desktop Protocol (RDP) sont supportés.

La pièce maitresse de cette implémentation est l'orchestrateur. L'orchestrateur est chargé de provisionner les machines virtuelles. Il déploie les images requises, configure le hardware de chaque machine (CPU, mémoire, interfaces réseaux...), configure l'OS qui a été installé (configuration adresse IP, comptes utilisateurs, mots de passe...). Il configure et installe également les logiciels additionnels souhaités.

En plus de la partie liée aux machines virtuelles, l'orchestrateur est chargé de configurer les différents réseaux virtuels et de créer les comptes utilisateurs qui doivent accéder à l'interface web de la solution en utilisant la passerelle de bureau d'accès à distance.

Cette autre solution de simulation, malgré les avantages et la facilité de déploiement qu'elle offre, est écartée. Pour cause, elle ne permet pas de satisfaire deux des contraintes essentielles soulevées. L'aspect hybride et la portabilité (export) ne sont pas pris en charge et ne sont pas implémentés sur la solution de *Cyber Range*.

2.2 Types d'hyperviseur

Actuellement, grâce aux progrès accomplis en terme de virtualisation, il est possible de virtualiser n'importe quel type d'élément. Les techniques de virtualisation sont devenues tellement complètes qu'il est possible de virtualiser une entité entière, une application, une ressource, un réseau et même de l'espace de stockage [26].

Deux catégories d'hyperviseur ont vu le jour. L'hyperviseur de type 1, ou hyperviseur natif (*bare metal*), et l'hyperviseur de type 2, ou hyperviseur *hosted*. La classification de ces hyperviseurs est liée à leur méthode de fonctionnement. L'hyperviseur de type 1 est installé à même le *hardware* de la machine. L'hyperviseur de type 2 est quant à lui installé sur le système d'exploitation. Dans ce cas de figure, l'hyperviseur de type 2 partage les ressources physiques avec le système d'exploitation.

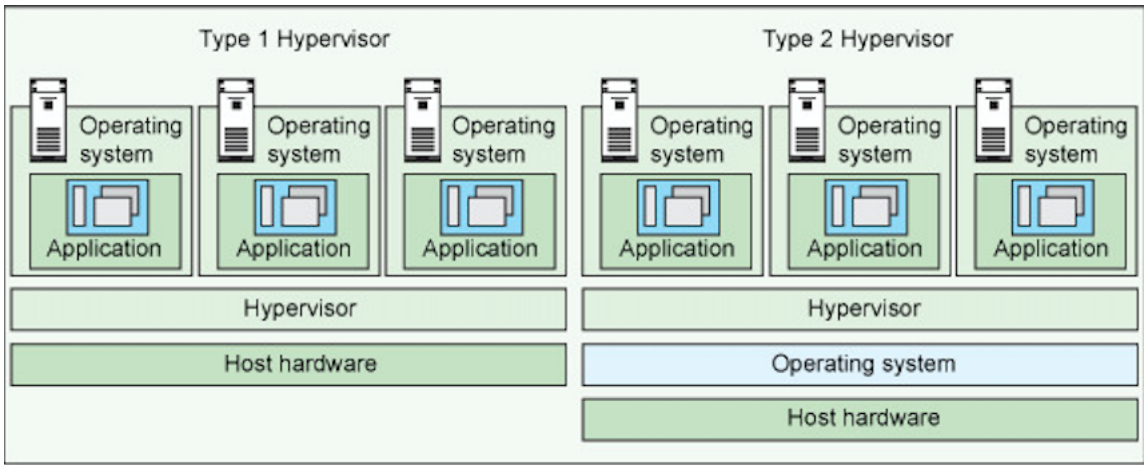


FIGURE 2.2 – Différences entre hyperviseur de type 1 et 2.
Source : ibm.com

On retrouve dans la catégorie des hyperviseurs de type 1, les solutions telles que VMWare ESX et ESXi, Microsoft Hyper-V ou les solutions basées sur Citrix XenServer.

Les hyperviseurs de type 2 regroupent quant-à eux des solutions comme Oracle VirtualBox, VMWare Workstation (Fusion pour Mac), QEMU sur Linux...

La virtualisation propose beaucoup d'avantages. Puisque les systèmes sont virtualisés, le coût du hardware diminue, on optimise les charges de travail et le système informatique devient flexible (1.4) et propose une meilleure réactivité. Le choix du bon type d'hyperviseur est donc crucial puisqu'il s'agit du pilier de la solution. Afin de savoir quel type d'hyperviseur correspond au mieux à ce projet, il faut effectuer un comparatif de performances.

Le choix de l'hyperviseur doit être fait entre un hyperviseur de niveau 1 et un hyperviseur de niveau 2. L'hyperviseur de niveau 2 est Oracle VirtualBox, qui est actuellement utilisé dans le laboratoire. L'hyperviseur de niveau 2 est comparé à l'hyperviseur de niveau 1, VMware ESXi.

L'hyperviseur de niveau 2 est installé sur un système d'exploitation Linux, Debian. Afin d'alléger au maximum les charges de la machine, le système d'exploitation a été installé sans interface graphique.

2.2.1 Performances

On retrouve dans la bibliothèque scientifique des articles qui comparent les différents types d'hyperviseur : *Performances analysis of selected hypervisors* (de Waldemar Graniszewski et Adam Arciszewski) ou encore *Hypervisors vs. Lightweight Virtualization : a Performance Comparison* (de Roberto Morabito and Jimmy Kjällman and Miika Komu).

Les principaux composants qui reviennent et qui sont évalués sont le CPU, la mémoire, les disques, les cartes réseaux et d'autres composants (de manière sporadique).

Les solutions retenues dans ces tests de performances sont le plus souvent Microsoft Hyper-V, VMware ESXi, OVM, VirtualBox et XenServer. La liste énoncée comporte des hyperviseurs de type différent.

En ce qui concerne les résultats, on remarque que le CPU est plus fortement impacté ou sollicité lorsqu'on utilise un hyperviseur de type 2. La cause logique est l'utilisation des ressources par l'hyperviseur mais également par le système d'exploitation sur lequel il est installé. D'après ce comparatif (qui date de 2016), VMware ESXi remporte la première place [16].

2.2.2 *Lightweight virtualization*

La *lightweight virtualization* est une solution de virtualisation plus légère que celle qui utilise un hyperviseur. Le principe d'une virtualisation plus légère réside dans le fait

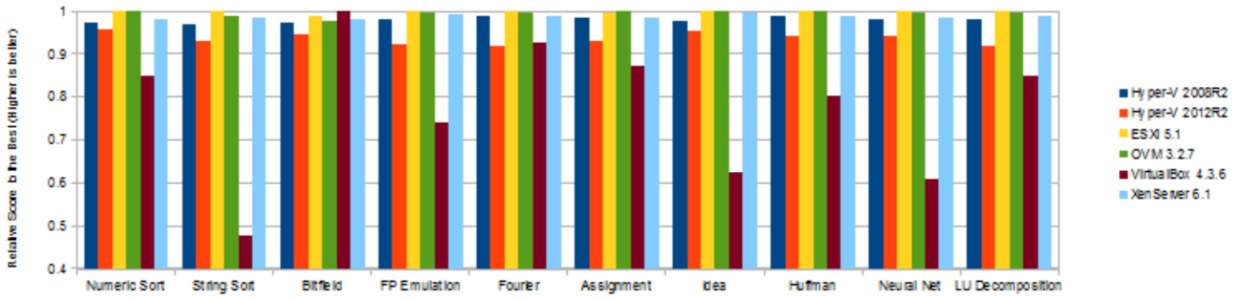


FIGURE 2.3 – Benchmark CPU hyperviseurs.
Source : [16]

d'utiliser des conteneurs pour isoler différents processus. Comme chaque solution, la conteneurisation a des avantages et des inconvénients. On peut virtualiser plus d'instances et l'image des disques est plus petite que sur un hyperviseur. En revanche, on ne peut pas exécuter des conteneurs Windows par dessus un hôte Linux et les ressources sont moins bien isolées que sur un hyperviseur. A l'inverse de la conteneurisation, l'hyperviseur opère directement sur la couche hardware de la machine. Les systèmes sont donc complètement isolés, ce qui permet d'instancier des machines avec un système d'exploitation différent. Le désavantage est qu'un système d'exploitation complet est installé là où un processus aurait pu suffire. Il y a donc un gaspillage de ressources. La conteneurisation permet de virtualiser uniquement le service dont on a besoin sans devoir instancier une machine complète. Il y a donc un gain de ressources[23].

Des tests complémentaires de performances sont réalisés dans ce mémoire sur les deux hyperviseurs choisis (pour rappel, VMware ESXi et VirtualBox). Le but de ces tests est, d'une part, de venir confirmer l'idée transmise par les différentes publications scientifiques, et d'autre part, de voir l'évolution, la gestion et le comportement des ressources de l'hyperviseur lorsqu'il exécute les scénarios sur le simulateur.

2.3 Cyber ranges

Les cyber ranges sont des solutions conçues pour que leurs utilisateurs puissent pratiquer, entrainer et améliorer leurs compétences de défense en cybersécurité. Le cyber range a pour but de simuler un environnement virtuel qui se rapproche le plus possible de l'environnement réel. Dans cet environnement, l'utilisateur pourra, par exemple, tester ses compétences en tant que *pentester*, en tentant de protéger le réseau d'une entreprise ou encore de faire face à une attaque liée à l'infrastructure d'un client. L'imagination est la seule limite du cyber range. Les cyber ranges sont catégorisés selon deux critères : le type et le secteur d'activité. Les types identifiés sont les cyber ranges de simulation, d'émulation et les cyber ranges *overlay*. Le secteur d'activité est défini sur base de l'environnement : académique, militaire ou commercial[8].

Les cyber ranges sont catégorisés cyber ranges *overlay* s'il existe une interconnexion entre un réseau de production et le réseau du cyber range. Le cyber range est dans ce

cas une solution hardware connectée à un réseau physique de production ou une solution software où le cyber range est virtualisé mais lié à un environnement de production. Un cyber range de simulation utilise des machines et des objects physiques réels. Un cyber range d'émulation utilise uniquement des ressources virtualisées[8].

Le concept de cyber range n'est pas un nouveau concept. Dès le début des années 2000, différentes solutions de cyber ranges voient le jour. Certains de ces cyber ranges étaient orientés compétition, tels que les compétitions *capture the flag* que l'on connait aujourd'hui.

Voici un récapitulatif des premiers cyber ranges basés sur les critères énoncés ci-dessus[8].

Simulation			Overlay	Emulation		
Académique	**Militaire**	**Commercial**		**Académique**	**Militaire**	**Commercial**
SECUSIM	SIMTEX	Breaking Point	PlanetLab	Emulab	NCR	NorthropGrumman
RINSE	CAAJED	EXata	X-Bone	Deter	JIOR	CounterHackChallenges
NetEngine	SAST	Building blocks		Virtualised CR	INL	Detica
Arena	Stealthnet			Reassure	Military Academy CRs	ATC
OPnet-based	High level wargaming					
Lariat						
VCSTC						

FIGURE 2.4 – Résumé des cyber ranges

2.3.1 CYRAN

CYRAN est un cyber range hybride construit par l'université de Montfort. Cette solution fournit l'infrastructure nécessaire en terme de ressources et de scénarios ainsi que des jeux de données sur lesquels l'utilisateur peut s'entrainer. Les données fournies dans les cyber ranges sont souvent des données recueillies lors de compétitions où les personnes sont chargées d'attaquer un système. Ces données visent toujours le même type d'équipement. CYRAN est le premier cyber range à proposer des données basées sur des systèmes industriels (Supervisory Control And Data Acquisition (SCADA) et Programmable Logic Controller (PLC)). Ces systèmes ne pouvant être virtualisés, leur intégration est faite de manière hybride avec le cyber range[18].

2.3.2 Alpaca

Alpaca est un autre type de solution qui propose la création de cyber range en prenant en compte les contraintes spécifiques des utilisateurs. Cette solution est basée sur trois composants. Une intelligence artificielle, une base de données qui regroupe différentes

vulnérabilités et les configurations des machines. Sur base de ces trois facteurs, la solution est capable de créer un cyber range unique qui regroupe les vulnérabilités sélectionnées dans la base de données. Les vulnérabilités sont mises en place de manière à ce que l'utilisateur puisse entrainer une compétence en particulier. Les machines virtuelles destinées au cyber range sont également créées de manière automatique[14].

Avec cette solution, chaque utilisateur peut bénéficier d'un scénario simultanément. Alpaca fonctionne de la manière suivante : un objectif de départ et un objectif de fin sont définis. La solution génère ensuite un chemin jusqu'aux vulnérabilités (dans la base de données) qu'elle estime correspondre au mieux aux besoins décrits précédemment. Chaque vulnérabilité dans la base de données peut avoir une configuration et chaque étape du chemin qui a été généré peut avoir plusieurs configurations possibles. Toutes ces configurations sont ensuite utilisées pour générer les machines virtuelles qui contiennent les vulnérabilités spécifiques aux choix de l'utilisateur[14].

2.3.3 Cyber Warfare Testbed

Cette solution de cyber range est mise en place sur un hyperviseur de type 1, XenServer. Le cyber range utilise des outils open source. Le *Tesbed* se compose de quatre parties. Le cluster de noeuds et les réseaux, le système d'attaque, le système de défense et le système de génération de rapport. Le cluster *Testbed* est composé de plusieurs noeuds qui sont connectés à un switch programmable. Ce switch programmable permet de construire n'importe quelle topologie. L'hyperviseur Citrix est suffisant pour la configuration de l'infrastructure de base. L'outil open source *Cloudstack* permet de configurer les noeuds plus en détail. Des noeuds sont créés pour les systèmes d'attaque et de défense[6].

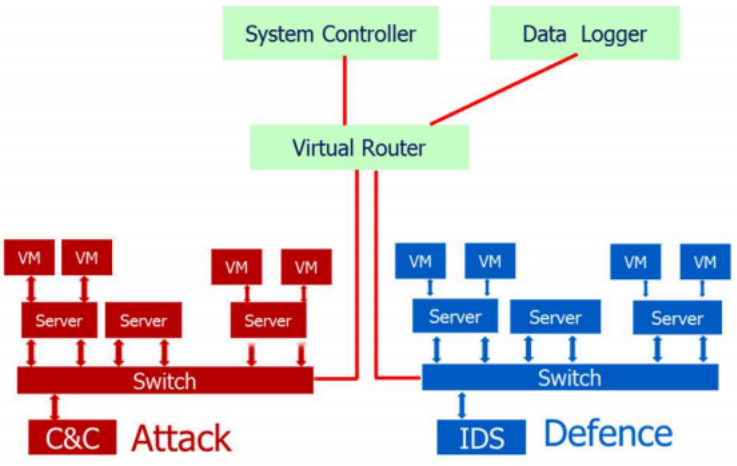


FIGURE 2.5 – Cyber Warfare Testbed.
Source : [6]

2.4 Ressources

Les scénarios sont imaginés sur des concepts de sécurité vus pendant les séances théoriques. Il existe des ressources qui peuvent être intégrées sur la plateforme de simulation et qui présentent des failles de sécurité volontaires pour l'entrainement.

2.4.1 Rapid7

Rapid7 est une société spécialisée dans le domaine de la cybersécurité. Son but est de fournir à ses clients différents services pour assurer la sûreté de l'information. On retrouve parmi les services proposés, du pentesting, des sessions d'*awareness*, du management de vulnérabilités...

La société Rapid7 met à disposition une machine virtuelle appelée Metasploitable. Cette machine est remplie de vulnérabilités à des fins éducatives. Le but premier est d'utiliser le *framework* Metasploitable pour tirer parti des vulnérabilités (d'où le nom de la machine). En plus des exploits qu'il est possible de réaliser, la machine regroupe de nombreuses vulnérabilités : injection SQL, failles XSS, failles CSRF, escalade de privilèges...

2.4.2 Damn Vulnerable Linux

Damn Vulnerable Linux est une distribution Linux basée sur les mêmes concepts que la machine Metasploitable. La différence entre la machine de Rapid7 est que Damn Vulnerable Linux n'est pas fournie sous la forme d'une machine virtuelle mais en fichier .iso. Etant donné les failles de sécurité que la distribution intègre, il n'est pas conseillé de l'installer sur une machine physique mais plutôt de l'utiliser en tant que LiveCD ou via une machine virtuelle.

2.4.3 VulnHub

VulnHub est une mine d'or en ce qui concerne le domaine de la cybersécurité. VulnHub est un site web qui répertorie une multitude de ressources destinées à acquérir des connaissances en cybersécurité. Des explications sur les différents types de failles et leur mode de fonctionnement sont disponibles. VulnHub est composé d'une large communauté. Cette communauté soumet à intervalle régulier des machines vulnérables basées sur la découverte de vulnérabilités.

VulnHub constitue une sérieuse base de documentation et de ressources qui peuvent être exploitées par les étudiants lors de leurs entrainements.

2.5 Didactique et pertinence des scénarios pour les étudiants en cybersécurité

La création des manipulations repose sur deux concepts : la pertinence de la matière inclue et l'aspect didactique, nécessaire pour que les manipulations proposées soient créées de manière à ce que l'étudiant puisse en tirer un maximum de connaissances et de bénéfices.

La didactique nécessaire à un bon enseignement n'est pas toujours évidente à mettre en place. Il est nécessaire de s'interroger sur la stratégie pédagogique qu'on va adopter. La question à se poser est de savoir comment les savoirs qui sont enseignés lors de la phase d'apprentissage théorique, sont bien compris et appris par l'étudiant.[24]

Afin d'apprendre, l'étudiant doit faire preuve de motivation. Cette motivation est soutenue par la confiance en soi et par le sentiment de compétence qu'il doit ressentir. Si l'étudiant se sent capable d'effectuer le travail et que celui-ci est assez clair pour que l'étudiant voit directement ce qu'il doit faire, il part de l'idée qu'il est capable de mener à bien la tâche qu'il lui a été confiée. Il est donc important que l'étudiant ait toutes les compétences nécessaires avant la séance de travaux pratiques. On parlera d'un *background* suffisant pour réaliser les manipulations. Pour avoir ces prérequis, une liste des matières à connaître peut être remise avant le début des scéances de laboratoire ou lors du cours théorique.[24]

Les savoirs ne doivent pas être donnés tels quels aux étudiants. Ils doivent être transformés pour être rendus accessibles. Le but n'est pas de leurs donner des exercices qui ne nécessitent aucune compréhension mais de leur apprendre à construire les scénarios de manière à ce qu'ils puissent en dégager quelque chose. Ils doivent savoir **pourquoi** ils font ce choix-là plutôt que de le faire car ils l'ont vu écrit quelque part. Ce type de situation oblige l'étudiant à réfléchir de lui même. Il sera amené, dans le monde professionnel, à prendre des décisions au cas par cas. Chaque situation est différente et répond donc à une solution qui doit être pensée en fonction de la situation. Le savoir doit être contextualisé et problématisé. Cela permet de définir le problème et ensuite de chercher une manière de le résoudre. Opérer de cette manière est souvent mieux perçu par les élèves puisqu'ils ont une situation concrète d'un cas donné.[24]

Des méthodes existent afin de développer des compétences :

- Un élève actif apprendra plus qu'un élève passif. Rien de tel que la pratique pour mettre en exergue les concepts théoriques.
- L'élève doit être confronté à de vraies simulations.
- Pratiquer un retour réflexif sur ce qu'il vient d'accomplir. L'élève doit se poser certaines questions : **pourquoi** a-t-il fait ça ? **Comment** l'a-t-il fait ? Et enfin **qu'a-t-il appris** ?

Une fois que les enjeux didactiques ont été cernés, on peut travailler sur le deuxième concept, tout aussi important : la pertinence de la matière.

Il s'agit ici d'identifier ce qui est important pour des élèves qui étudient la cybersécurité. Cette discipline est une matière dont les idées reçues sont assimilées aux notions de *hackers, hacking ou pentesting*. Elle reste néanmoins un domaine très vaste.

La cybersécurité est la sécurité des systèmes d'information. Elle désigne tout système informatique, qu'il soit embarqué ou non, émulé ou pas... C'est ici qu'avoir une plateforme de virtualisation hybride est un avantage considérable. Les étudiants peuvent maintenant interagir avec des composants utilisés pour l'informatique embarquée.

Deux idées sont exploitées dans la création des scénarios. La première fait appel à l'esprit de *hacker* qui consiste à identifier les failles d'un système. Dans ce cas de figure, l'étudiant sera amené à trouver toutes les vulnérabilités qui pourraient mener à une fuite de données. Le fait de mettre à disposition de l'étudiant une machine volontairement truffée de failles de sécurité permettra d'utiliser sa motivation pour tenter de corrompre un système ou de subtiliser des données sensibles. La maitrise et la compréhension du fonctionnement d'outils se fera lors de la manipulation.

L'autre scénario sera basé sur l'esprit de compétitivité de l'étudiant. Le but est qu'il mette en place une topologie réseau du début à la fin. La sécurité doit être pensée dès la phase de conception ! Une fois la topologie opérationnelle et fonctionnelle, l'étudiant passera son scénario à un autre étudiant qui devra tenter de trouver quelles sont les faiblesses dans son implémentation. Plusieurs points de didactique entrent en jeu ici. L'élève sera confronté à une situation réelle. Sa motivation sera stimulée par le fait qu'un membre de sa classe soit chargé de trouver des traces de négligeances de sa part lors de la configuration des équipements. La fierté personnelle devrait conduire l'étudiant à se donner au maximum pour ne pas qu'un condisciple trouve une vulnérabilité dans son implémentation.

D'autres scénarios sont créés, tout en conservant la méthode active d'apprentissage. Les autres scénarios sont basés sur des concepts vus lors du cours théorique.

Chapitre 3

Implémentation

L'implémentation de la solution se divise en deux grandes parties. La première partie concerne le simulateur et la deuxième partie concerne l'hyperviseur.

3.1 Simulateurs

GNS3 et Hynesim sont les simulateurs qui répondent aux besoins identifiés.

3.1.1 Critères de comparaison

Les critères de comparaison suivants sont utilisés pour comparer GNS3 à Hynesim. Le choix du simulateur est le résultat de ce comparatif. **Actuellement, aucune étude qui compare Hynesim à GNS3 n'a été recensée.**

On retient les éléments de comparaison suivants :

- Le système d'exploitation qui héberge le simulateur.
- Le type de licence.
- La documentation.
- L'installation.
- Le support et la communauté.
- La possibilité de travailler en mode client ou en mode serveur.
- Les *appliances* disponibles.
- Le mode de fonctionnement hybride qui comprend l'interconnexion à un réseau physique.
- La collaboration (définie comme l'aptitude d'une personne a pouvoir travailler sur le même projet de manière simultanée).
- L'aspect multi-projets qui regroupe la manière dont les ressources sont partagées par le serveur.
- La méthode de démarrage du logiciel.
- La configuration de la plateforme.
- La facilité d'utilisation. La solution est-elle *"user-friendly"* ?

— La présence native de Dynamips [1].
— L'exportation des projets.
— La comptabilité avec des versions différentes.
— L'éventuel *loadbalancing*, ou répartition des charges.

3.1.2 Hynesim [2]

Hynesim est la solution non open source utilisée actuellement. Hynesim doit être installé sur une machine Linux lorsqu'on l'utilise en version gratuite. L'installation est laissée à l'utilisateur. Il est spécifié dans la documentation que la version de Linux qui doit être utilisée est une Debian 8.0 Jessie. Diateam ne garantit pas une totale compatibilité si Hynesim est installé sur une autre version que celle spécifiée.

Hynesim est disponible sous trois types de licences, mais il n'y a que la licence gratuite qui a été testée. Lorsqu'on utilise la version gratuite, c'est l'utilisateur qui doit installer le système.

Installation

Hynesim est un produit assez capricieux en terme d'installation quand l'utilisateur doit déployer la solution. Ce système doit être installé sur une machine Linux Debian 8.0 Jessie. Une appliance d'Hynesim est disponible au format OVA pour les personnes qui souhaitent tester la solution gratuitement. Attention, VirtualBox ne supporte pas la virtualisation imbriquée, il est donc impossible de profiter des performances offertes par le module KVM lorsqu'on utilise cette appliance [3]. Il est préférable de désactiver le gestionnaire de réseau Debian lorsqu'on utilise Hynesim afin d'éviter tout conflit. Un nettoyage de certains paquets présents nativement sous Debian doit être fait afin d'éviter tout problème lors de l'installation.

Le dépôt d'Hynesim doit être ajouté dans les sources du gestionnaire de paquets. Il faut aussi ajouter la clé de signature du dépôt Hynesim. Ensuite, on met à jour le noyau Linux en utilisant les dépôts qui ont été précédemment ajoutés dans le fichier de source du gestionnaire de paquets. La mise à jour du noyau nécessite le redémarrage de la machine pour éviter tout problème de compatibilité[3].

On utilise le gestionnaire de paquets pour installer Hynesim. Le méta-paquet Hynesim contient la suite logicielle complète qui comprend Hyneview. Hyneview est nécessaire pour le management d'Hynesim. Hynesim fonctionne selon une arborescence bien précise que l'utilisateur doit créer[3] :

1. Dynamips est un émulateur utilisé pour virtualiser la partie *hardware* des routeurs Cisco sur base de leur IOS.
2. La majorité des informations provient de la documentation officielle d'Hynesim disponible sur `https://www.hynesim.org/`
3. Source : https ://www.hynesim.org/

```
/data
        |---hynesim
                |---catalog
                        |---entities
                        |---guestfoundry
                        |---topologies
                |---export
                |---import
                |---resources
                |---shared
```

Il faut modifier une variable dans les fichiers de configuration d'Hynesim pour activer Hynesim-Glacier. Aucune information n'est disponible sur la configuration de ce service mise à part le fait qu'il s'agisse d'un *daemon linux* qui s'exécute en arrière plan et qui permet à Hynesim de fonctionner. Il est d'ailleurs étonnant que ce paramètre ne soit pas directement activé dans les fichiers de configuration d'Hynesim puisqu'il est indispensable à son fonctionnement. D'autres *daemons* doivent être activés : Hynesim-Glacier, Hynesim-Node et Hynesim-Master[3].

Pour installer Kernel-based Virtual Machine (KVM)/QEMU qui permet de virtualiser des équipements dans Hynesim, il faut une nouvelle fois modifier les fichiers de configuration[3].

Pour terminer, il faut activer l'accès à distance au serveur. Ce paramètre est modifié dans les fichiers de configuration suivants pour permettre au client Hyneview distant de se connecter au serveur Hynesim :

```
/etc/hynesim/hynesim-glacier.conf
/etc/hynesim/hynesim-master.ini
/etc/hynesim/hynesim-node.ini
```

Un champ commun se trouve dans ces trois fichiers de configuration :

```
Ice.Default.Host = 127.0.0.1
```

Si ce champ reste sur l'adresse de *loopback*, il ne sera pas possible d'accéder à Hynesim en utilisant Hyneview depuis une machine distante. Hynesim ne sera accessible que via la machine qui l'exécute. Pour accéder à Hynesim depuis une machine différente qu'elle-même, il faut remplacer l'adresse de *loopback* par l'adresse du serveur.

Prise en main

L'utilisateur utilise Hynesim à travers Hyneview. L'utilisateur doit être authentifié pour accéder à Hynesim. Les *credentials* par défaut sont disponibles dans la documentation.

Il est possible de voir si les technologies de virtualisation telles que Dynamips ou QEMU sont bien installées sur le serveur en se rendant dans l'onglet monitoring de plate-

forme. La création d'une topologie est triviale. En revanche, Hynesim n'intègre pas d'équipements par défaut. Un catalogue d'entités est disponible gratuitement mais il ne comprend que quelques équipements : un switch, une carte hybride, un point d'accès Wi-Fi et un pont intertopologie.

Entités

Les entités sont décrites au format XML. La configuration des équipements se fait via ce fichier. Si on désire supprimer un port du switch, il faut retirer ce port dans le fichier XML.

L'import des ressources se fait en récupérant l'archive. Cette archive doit être extraite dans un dossier bien précis : /data/hynesim/import. Dès que les entités sont placées dans ce dossier, elles apparaissent dans Hynesim et sont marquées comme prêtes pour l'importation. Lorsqu'elles sont importées dans Hynesim, les entités disparaissent du dossier d'importation.

Cette procédure peut paraitre triviale lorsqu'on dispose d'entités correctement formatées pour Hynesim. Aucune autre image n'est disponible à l'import, ce qui rend la difficulté d'intégrer d'autres entités dans Hynesim considérable [4].

Une ressource importée dans une feuille de travail est verrouillée. Si la ressource est verrouillée, il est impossible de l'utiliser autre part que dans la feuille de travail où elle se trouve. Pour la déverrouiller, l'utilisateur doit supprimer la ressource de la feuille de travail. Si l'utilisateur veut disposer de la même ressource plusieurs fois, il doit la cloner autant de fois que nécessaire.

Le clonage d'une ressource non fournie par Hynesim (du moins pour le routeur fourni par le Ph. Dricot) provoque un crash du simulateur. A ce jour, aucune solution n'a été trouvée, mis à part le fait de la réimporter autant de fois que nécessaire.

4. Merci au Ph. Dricot d'avoir fourni l'image d'un routeur prêt à l'import.

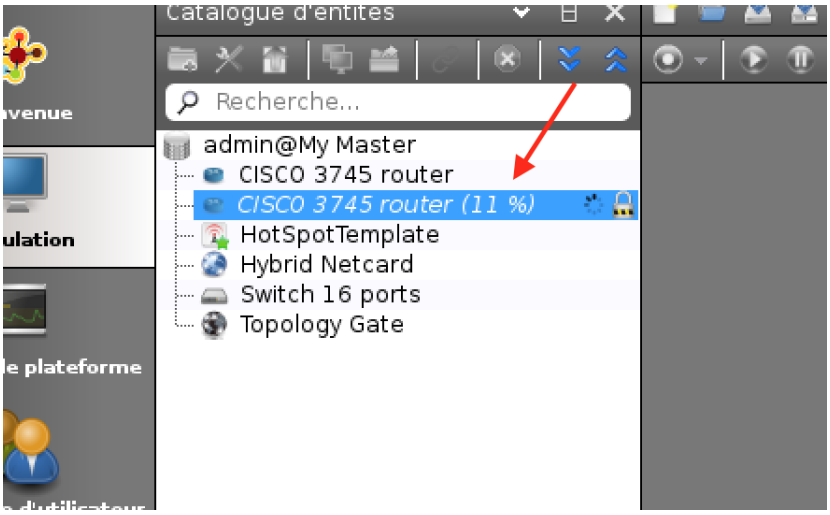


FIGURE 3.1 – Problème lors du clonage d'une entité.

Mode hybride

Hynesim supporte le mode Hybride. Une carte hybride est fournie dans le catalogue d'équipements mis à disposition gratuitement. Cependant cette carte n'a jamais fonctionné durant les tests et malgré les nombreuses recherches. Ce problème peut venir du fait que la ressource soit limitée dans la version gratuite.

Dashboard

La vue globale d'Hynesim reprend les différentes entités disponibles, la topologie, le noeud où sont assignés les équipements et les différents onglets et bouttons de configuration. Un aperçu de la vue principale est disponible au point C.1.

Langues

Hynesim est disponible en version française et en version anglaise. La documentation est uniquement disponible en version française.

Longévité

La solution, du moins dans sa version gratuite, n'est pas faite pour durer puiqu'elle ne fonctionne que sur une version de Debian qui est maintenant obsolète. La documentation (limitée) relative à la version gratuite n'est également pas à jour.

3.1.3 GNS3

GNS3 est un simulateur qui peut fonctionner en mode client ou en mode serveur. Les élèves qui ne disposent pas d'une puissance de calcul suffisante peuvent profiter des

ressources du labo. Ceux, en revanche, qui possèdent une puissance de calcul suffisante peuvent travailler sur leur propre poste de travail. Ce procédé permet de réduire la charge des serveurs tout en permettant à tout les élèves de travailler sur la manipulation.

Lorsqu'on utilise GNS3 en mode serveur, le client qui est utilisé pour faire la liaison au serveur doit posséder la même version que le serveur. GNS3 ne gère pas le *loadbalancing* de manière native. Il est par contre possible de rajouter plusieurs serveurs dans le même scénario et de répartir les ressources déployées. C'est l'utilisateur qui choisit sur quel serveur la ressource doit être déployée.

Les tests de la solution ont permis de déterminer que lorsque le client GNS3 est lié à plusieurs serveurs, c'est le serveur principal qui gère la topologie. Cette déduction a été faite en consultant la charge et la mémoire des serveurs. Une gestion efficace des ressources, basée sur cette observation, consiste à provisionner le serveur principal avec moins de ressources que les autres puisqu'il gère uniquement la topologie et non les équipements qui la composent. On peut assigner plus de ressources aux serveurs qui doivent gérer les *nested VM*. Les *remote servers* qui ne sont pas des serveurs principaux sont liés à un serveur principal. Lorsqu'on indique le serveur principal dans les réglages de GNS3, les *remote serveurs* qui lui sont attachés sont également assignés au client.

Installation

Le client GNS3 est fourni pour Windows, Linux et Mac. Pour la version serveur, une machine au format OVA est disponible depuis le site GNS3. Cette machine doit simplement être importée sur l'hyperviseur. Aucune configuration supplémentaire n'est nécessaire.

Prise en main

GNS3 ne nécessite pas d'authentification lorsqu'il est utilisé en mode serveur. Si besoin est, une authentificaion peut quand même être définie. Comme Hynesim, GNS3 est fourni sans entités. Des ressources comme un noeud cloud, les switch ou encore des PC virtuels sont installés nativement.

En revanche, il est possible d'installer des appliances de routeurs Cisco uniquement en se basant sur le fichier IOS. D'autres équipements peuvent être ajoutés de la même manière (ou via une image .iso).

Entités

Dynamips est utilisé pour simuler les équipements Cisco. Pour les systèmes qui sont basés sur Solaris, on peut également utiliser *IOS on Unix*. Pour la virtualisation embarquée, on peut utiliser QEMU (intégré nativement), VirtualBox, VMware ou Docker. Docker permet d'utiliser des conteneurs différents pour les instances virtualisées. Il utilise moins de puissance de calcul que QEMU et il peut être utilisé quand on désire virtualiser des

services sans utiliser une grande quantité de mémoire. A l'inverse de Docker, QEMU virtualise la partie hardware du Personal Computer (PC), ce qui demande un système d'exploitation. Beaucoup de fabriquants proposent des images QEMU qui peuvent être utilisées avec Graphical Network Simulator (GNS3) (marketplace).

Les solutions de base qui sont utilisées pour nos scénarios sont Dynamips et QEMU. Les équipements sont accessibles via le terminal ou en utilisant un client VNC.

Les serveurs GNS3 fonctionnent de manière intelligente. Les entités sont assignées à un serveur. Si pour une quelconque raison, l'utilisateur décide de déployer une entité qui est présente sur le serveur A, sur le serveur B, celle-ci est directement copiée vers le serveur B (de manière transparente).

Mode hybride

Le cloud fourni avec GNS3 est l'entité qui sert à interconnecter un réseau physique à un réseau logique. Ce noeud ne nécessite aucune configuration pour fonctionner. L'utilisateur doit simplement spécifier l'interface physique qui sera assignée au noeud. Un *device* logique connecté au noeud peut alors communiquer avec un équipement physique.

Un test de connexion a été établi entre une ressource virtualisée (vPC) et une ressource physique présente dans le LAN. Ce test s'est montré positif.

Dashboard

GNS3 intègre la même sorte de vue qu'Hynesim. On retrouve une feuille travail et la liste des équipements disponibles à gauche. Sur la droite, l'utilisateur a un aperçu de la consommation du CPU et de la mémoire de la machine virtuelle. La partie du bas est réservée aux logs.

Langues

GNS3 est disponible en français et en anglais. La documentation GNS3 est disponible en anglais.

Longévité

La solution GNS3 existe depuis de nombreuses années. Elle s'est étoffée au fil du temps, intégrant de plus en plus de fonctionnalités. L'arrivée du marketplace GNS3 où les constructeurs peuvent rendre disponibles des images de leur appliance en fait un plus. GNS3 est open source et est supporté par une large communauté. L'ensemble de ces caractéristiques suggèrent que GNS3 est une solution à long terme.

3.1.4 Choix du simulateur

Sur base de l'analyse effectuée ci-dessus et au vu du manque de fonctionnalités et de documentation dans la version gratuite d'Hynesim, le simulateur choisi pour être intégré à l'hyperviseur est GNS3.

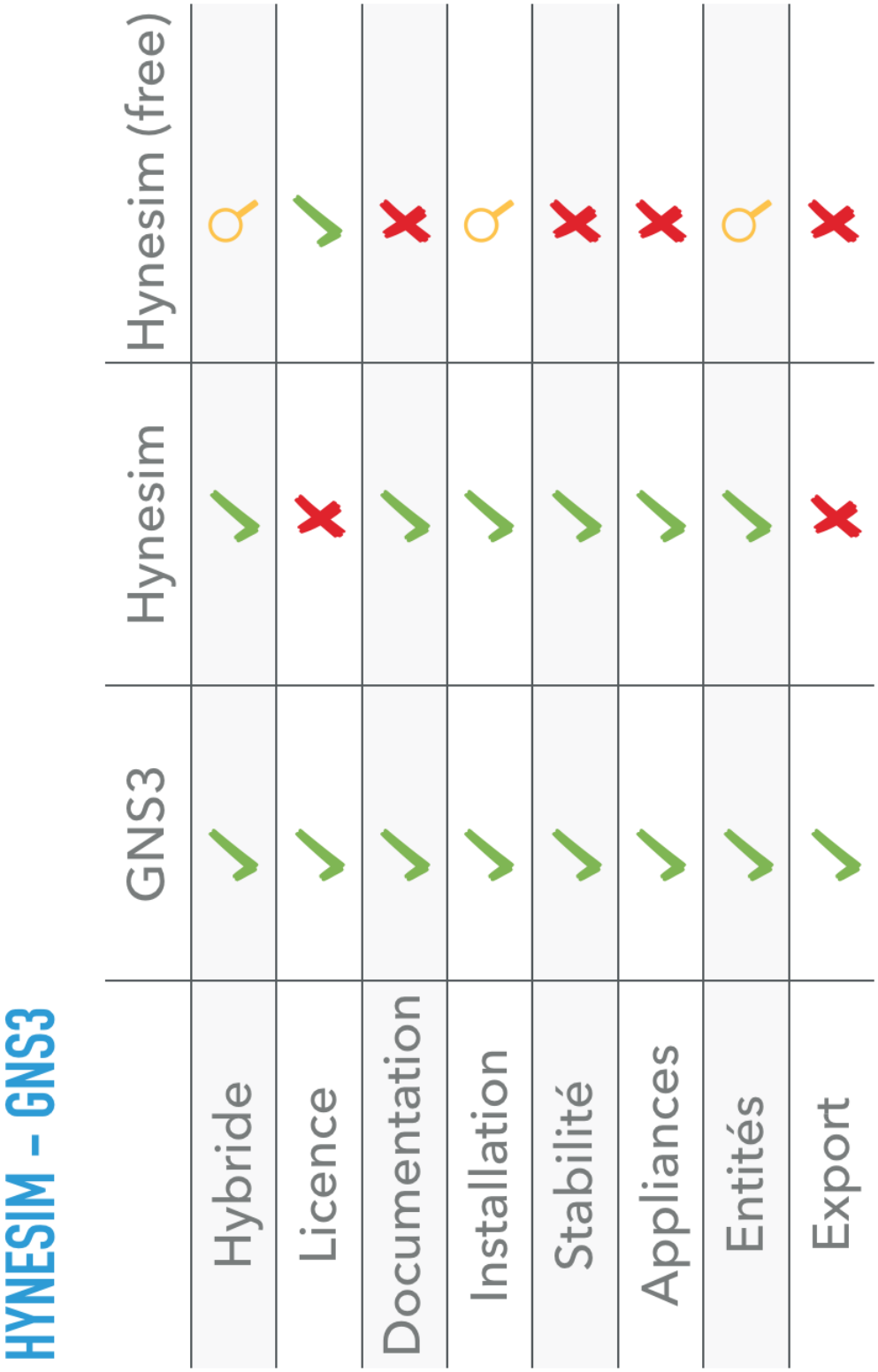


FIGURE 3.2 – Tableau récapitulafif des simulateurs

3.2 Plateforme de virtualisation

3.2.1 Oracle VirtualBox

VirtualBox est la solution actuellement utilisée. Pour maximiser les gains de performances, l'hyperviseur est installé sur une machine sans interface graphique (Debian). Le management de la machine se fait en ligne de commande via le protocole SSH.

Coût

VirtualBox est proposé gratuitement.

Installation

VirtualBox n'est pas disponible à l'installation via le gestionnaire de paquets. Les sources et les différents répertoires doivent être ajoutés à la main. La version de Debian actuelle est Debian 9, ce qui pose problème puisque certaines librairies utilisées sous Debian 8 sont devenues obsolètes sous Debian 9. VirtualBox, dans sa version actuelle en ligne de commande, nécessite l'installation de paquets qui ne sont plus maintenus à jour par Debian. L'archivage de certaines librairies pose problème car le serveur a été installé d'après une image *netinstall*. La particularité de ce type d'image est d'aller rechercher les paquets les plus récents lors de l'installation du système.

Différentes étapes ont été nécessaires pour installer VirtualBox en ligne de commande [5]. L'ajout des sources VirtualBox dans le gestionnaire de paquets.

```
root@isengard5:~# cat /etc/apt/sources.list.d/virtualbox.list
deb [arch=amd64] http://download.virtualbox.org/virtualbox/debian bionic contrib
```

FIGURE 3.3 – VirtualBox sources.

L'ajout des clés publiques du dépôt d'Oracle.

```
wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc
 -O- | sudo apt-key add -
```

Si d'autres versions de VirtualBox sont installées sur la machine, elles doivent être désinstallées pour éviter tout problème de compatibilité. Des dépendances sont manquantes lors de l'installation. Elles peuvent être installées en utilisant le gestionnaire de paquets et ses attributs de recherche pour les paquets manquants. Il est possible que certaines dépendances diffèrent du nom de paquet. Dans ce cas, une simple recherche sur le site Debian permet de trouver le paquet correspondant.

VirtualBox, son environnement et ses machines virtuelles fonctionnent exclusivement en ligne de commande (ce qui est voulu pour une optimisation des ressources de la

5. Guide d'installation disponible sur https ://www.virtualbox.org/wiki/Linux$_{D}ownloads$

machine). La commande *vboxmanage* est la commande qui permet d'utiliser VirtualBox sans interface graphique. L'image du serveur GNS3 est téléchargée depuis le site de GNS3, en choisissant le bon format (VirtualBox). La machine est ensuite transférée puis importée sur l'hyperviseur.

```
vboxmanage import GNS3\ VM.ova
```

Cette procédure d'importation est répétée à quatre reprises afin de disposer de quatre serveurs GNS3.

Les machines importées sont ensuite configurées pour avoir les mêmes spécifications. Le serveur est composé de 4vCPU et de 8GB de mémoire. L'adaptateur réseau est mis en *bridge* afin que les flux réseaux ne soient pas *"natés"*.

```
vboxmanage showvminfo "GNS3 VM"
vboxmanage modifyvm "GNS3 VM" --memory 8192 --cpus 4
vboxmanage modifyvm "GNS3 VM" --nic1 bridged
```

3.2.2 VMware ESXi

ESXi fait parti d'un groupe de produits proposés par VMware appelé VMware vSphere. vSphere regroupe les solutions ESX, ESXi et vCenter Server. ESXi est la nouvelle version d'hyperviseurs que VMware propose. La différence entre ESX et ESXi est l'architecture sur laquelle ces solutions sont construites. ESXi est basé sur la dernière architecture développée par VMware. VMware indique d'ailleurs que leurs futurs développements seront basés exclusivement sur la version ESXi, laissant tomber ESX[25].

Le management des hôtes ESXi est plus facile que pour les hôtes ESX. ESXi peut être totalement géré depuis un navigateur web. Son interface de management est basée sur de l'HTML5. ESX quant à lui doit être managé en installant un client vSphere.

3.2.3 Coût

ESXi est disponible en version gratuite ou en version payante. La version gratuite est limitée et ne propose donc pas les options suivantes[3] :

- Le management de l'hôte en utilisant vCenter Server.
- On ne peut utiliser qu'un seul serveur physique (les clusters font partis de la version payante).
- Un maximum de 2 CPUs est autorisé par hôte.
- Le maximum de vCPUs par hôte est de 128.
- Chaque Virtual Machine (VM) ne peut se voir assigner qu'un maximum de 8 vCPUs.
- La RAM maximum autorisée par hôte physique est de 12 TB.
- Le support n'est pas compris dans la version payante.

Il est possible d'effectuer une mise à jour de licence pour bénéficier de la version payante sans réinstaller l'hôte. L'image ESXi fournie par VMware est identique, que ce soit pour

la version payante ou pour la version gratuite. Les fonctionnalités sont liées aux clés de licence. Il n'est donc pas nécessaire d'installer de nouveaux *packages* pour bénéficier de nouvelles fonctionnalités[3].

La version payante est laissée à l'utilisateur en version d'essai pour un délai de 60 jours. Toutes les fonctionnalités sont activées durant cette période. Après la période d'essai, l'utilisateur peut continuer à utiliser son hyperviseur en introduisant une clé de licence gratuite. Les options payantes ne sont pas nécessaires pour notre utilisation. Les seuls critères importants qu'il faut retenir d'une utilisation gratuite d'ESXi est la limitation au niveau *hardware*. Le serveur ne peut posséder que deux processeurs. Le tableau ci-dessous reprend les différentes fonctionnalités qu'offre ESXi en fonction de la clé de licence[27].

TABLE 3.1 – Licences ESXi [27]

Limitations	Versions of ESXi		
	Free license	Evaluation license	Enterprise plus license
Physical memory	–	–	–
Technical support	Community supported, no VMware commercial support	Community supported, no VMware commercial support	Commercial support from VMware
Ability to manage via vCenter Server	–	–	+
License time	–	60-day time limit	–
Max number of physical CPUs per-host	2 physical CPUs socket per Hosts limitations	–	–
Max number of Virtual processors (vCPU) per-VM	8 vCPUs	128 vCPUs	128 vCPUs
APIs	Product APIs are read-only	–	–

3.2.4 Performances

Pour mesurer les performances des deux systèmes, VirtualBox et ESXi, différentes situations ont été imaginées. Au total, ce sont dix scénarios différents qui ont été mis en place pour pouvoir analyser le comportement des hyperviseurs. Le but est de mettre les hyperviseurs dans des situations similaires à celles qu'ils rencontreront lorsqu'ils seront utilisés par les étudiants.

D'un point de vue *hardware*, les hyperviseurs ont les mêmes caractéristiques. Ils possèdent un CPU de 16 coeurs, une mémoire de 16GB et un disque dur de 500GB. Chaque serveur GNS3 est également provisionné de la même manière, qu'il soit hébergé sous VirtualBox

ou sous ESXi. A savoir, 4 CPU virtuels et 8GB de mémoire.

Les scénarios permettent de voir comment l'hyperviseur gère ses ressources face à des comportements différents. Une partie des scénarios concerne la technologie Dynamips, tandis que l'autre partie des scénarios concerne la partie *nested* VM qui intègre la virtualisation via QEMU.

Chaque scénario est par la suite dupliqué de manière plus lourde et conséquente. Cette duplication est utilisée pour montrer la gestion des ressources de l'hyperviseur en situation de stress.

Les ressources monitorées sont l'utilisation du CPU et l'utilisation de la mémoire. Les mesures ont été prises de deux manières différentes.
La première manière est de relever les mesures directement sur l'hyperviseur en utilisant des sondes SNMP. Les données relatives aux différentes métriques sont stockées dans une base de données SQL. La solution choisie pour effectuer ces mesures est Icinga2. Icinga2 est une version *fork* du célèbre outil Nagios. L'avantage d'utiliser Icinga2 réside dans le fait que la solution propose nativement, le monitoring via SNMP. De plus, différents plugins sont disponibles et permettent une intégration simple de l'hyperviseur ESXi. Pour permettre une visualisation optimale des données, une solution de modélisation des données est utilisée avec Icinga : Grafana. Grafana est une solution graphique qui utilise Apache2 comme serveur web. Cette solution a l'avantage d'être open source et de compter une communauté assez importante (ce qui signifie que le projet est maintenu et évolue). Grafana permet l'intégration de différents *dashboards* en fonction des éléments qu'on monitore. La deuxième manière est de relever les mesures qui sont affichées par le serveur GNS3.

La première manière monitore donc les performances totales de l'hyperviseur alors que la deuxième manière monitore les performances du serveur virtuel.

Pour ce test de performances, on utilise deux *dashboards* qui reprennent les mêmes composants : la charge et la mémoire. Les mesures ont été prises dix fois afin d'éviter de possible effets de *caching*. En effet, il fallait déterminer si GNS3 mettait des données en cache lorsqu'elles étaient utilisées, afin d'améliorer le temps d'accès aux ressources.

Concernant les scénarios, un scénario 0 existe. Il porte le numéro 0 car ce n'est pas vraiment un scénario. Les mesures réalisées lors du scénario 0 sont des mesures prises à vide. Cette mesure permet de visualiser les ressources que le serveur GNS3 utilise (à vide) en fonction de l'hyperviseur. Ce test 0 a pour but de déterminer si les ressources sont utilisées de la même manière sous VirtualBox et ESXi.

Le premier scénario est un scénario orienté sur la technologie Dynamips. Chaque hyperviseur dispose d'un seul serveur GNS3. Le premier scénario GNS3 intègre un routeur. Il s'agit de la seule ressource utilisée lors de ce scénario.

Le deuxième scénario, comme dit ci-dessus, est une duplication du précédent. Le but est d'augmenter la charge dans le même scénario pour visualiser la gestion des ressources sur l'hyperviseur. Chaque hyperviseur dispose d'un seul serveur GNS3. A la différence du scénario précédent, le nombre de routeurs est augmenté. Le scénario compte donc dix routeurs.

Le troisième scénario a pour but de tester les performances des hyperviseurs lorsqu'une machine virtuelle est inclue dans le scénario. C'est donc la technologie Dynamips qui est testée. Chaque hyperviseur dispose d'un seul serveur GNS3. Une machine Metasploitable est déployée dans le scénario.

Le quatrième scénario est basé sur le précédent mais il intègre quelques composants en plus. Il est composé au total de trois machines virtuelles Metasploitable, d'un switch et d'un PC virtuel.

Le cinquième scénario teste également la technologie QEMU mais avec des machines virtuelles plus imposantes. En effet, la machine Metasploitable est une machine assez légère et qui ne consomme pas trop de ressources (1GB de disque et 512MB de mémoire). A l'inverse, la machine virtuelle que l'on intègre dans le cinquième scénario est une machine *custom* qui a été réalisée pour le mémoire. Il s'agit de la machine **PTF**, *Pentester Framework* qui intègre toute une panoplie d'outils destinés au *pentesting*. Cette machine virtuelle possède un disque dur de près de 8GB et est configurée pour utiliser 2048MB de mémoire.

Le sixième scénario est basé sur le cinquième. Trois machines virtuelles **PTF** sont déployées.

Le septième scénario innove et augmente la charge directement au niveau de l'hyperviseur. En effet, on passe d'un serveur GNS3 à quatre serveurs GNS3 par hyperviseur. Le but recherché en augmentant le nombre de serveur GNS3 est de trouver une meilleure stabilité tout en diminuant la notion de *single point of failure*. Si un des serveurs GNS3 a un problème, trois autres serveurs s'exécutent et se répartissent la charge. Aucun scénario n'est exécuté. Le septième scénario est comme le scénario 0 mais avec quatre serveurs par hyperviseur. Les métriques sont récoltées "à froid" pour voir comment l'hyperviseur gère les ressources des serveurs virtuels.

Le huitième scénario intègre toujours quatre serveurs GNS3 par hyperviseur. Chaque serveur GNS3 exécute un scénario qui comprend dix routeurs.

Le neuvième scénario intègre également quatre serveurs GNS3 par hyperviseur. Chaque serveur GNS3 exécute un scénario qui comprend trois machines Metasploitable, un switch et un PC virtuel.

Pour terminer, le dixième scénario intègre quatre serveurs GNS3 par hyperviseur. Chaque serveur GNS3 exécute un scénario qui est composé de trois machines **PTF** et d'un PC

virtuel. Ce scénario n'a pas été réalisé en raison des problèmes rencontrés sous Debian avec les *nested VM*.

Performance de l'hyperviseur

Le relevé des métriques au niveau de l'hyperviseur, a montré qu'en moyenne, pour la gestion de la charge et de la mémoire, l'ESXi était plus performant que Debian. Les valeurs sont cependant proches et nécessitent quelques explications.

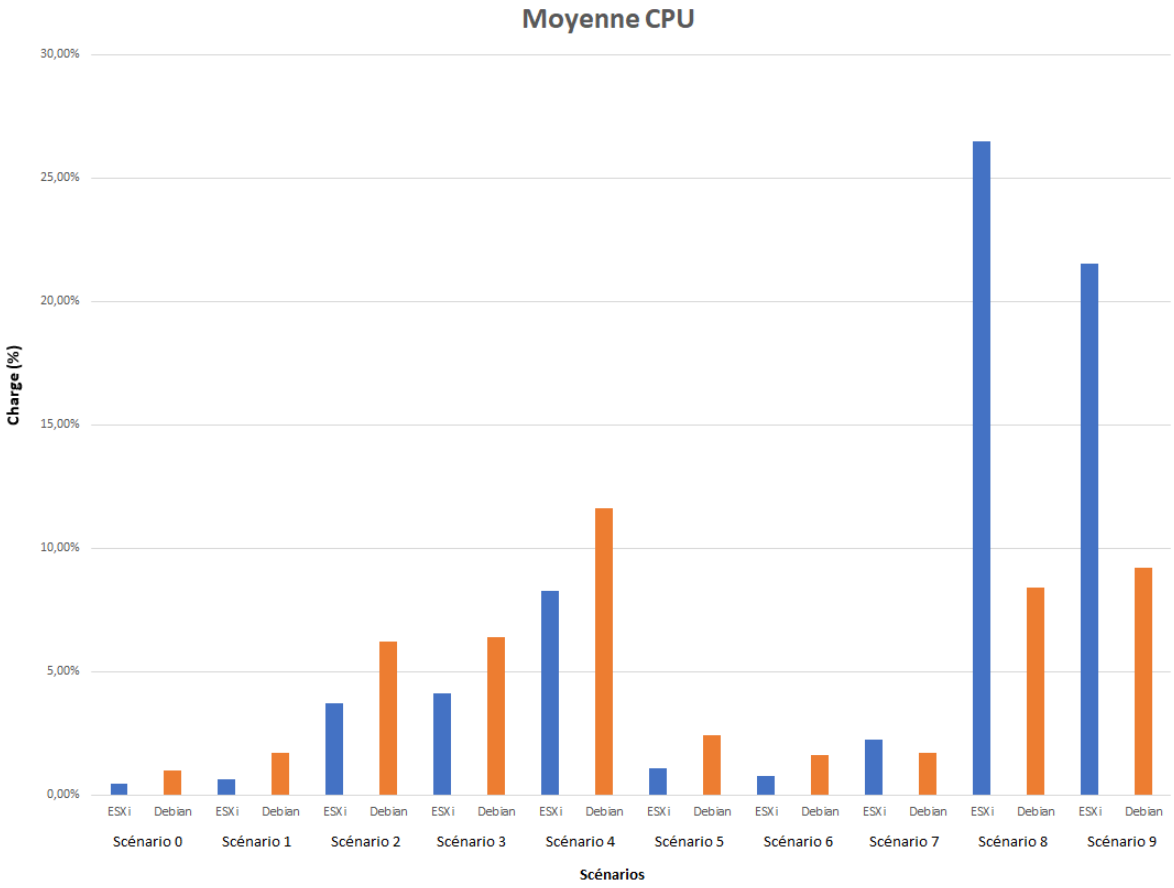


FIGURE 3.4 – Charge moyenne mesurée sur l'hyperviseur.

On observe sur le graphique ci-dessus, que l'ESXi a tendance à avoir une plus petite valeur moyenne que la Debian sur la plupart des scénarios. La charge CPU est en revanche plus importante sur les derniers scénarios quand l'hyperviseur doit gérer plusieurs machines GNS3. La raison de cette différence est assez simple à expliquer. VirtualBox a démarré les scénarios 8 et 9 mais l'utilisateur n'a jamais pu avoir accès aux entités. La charge CPU est retombée beaucoup plus vite que sur l'ESXi pourtant les instances n'étaient pas accessibles. En revanche, la charge CPU est restée élevée sur l'ESXi mais, à la différence de VirtualBox, l'utilisateur a pu accéder aux instances qu'il avait importées dans le scénario.

La représentation des données sous forme de *plotbox* permet une visualisation simple des données. Cette visualisation montre la performance d'un système par rapport à un autre. Pour qu'un système soit considéré comme nettement plus performant, les boîtes à moustaches ne doivent pas être allignées. En visualisant les valeurs des métriques relevées, on obtient le graphe suivant :

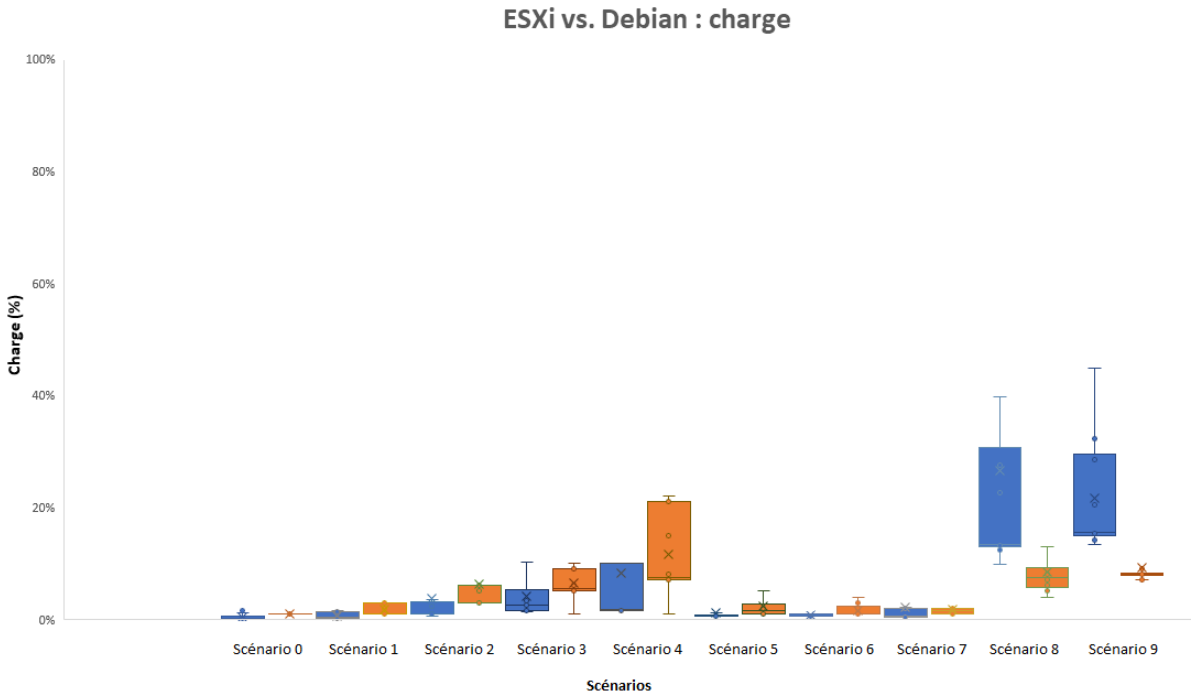


FIGURE 3.5 – Charge mesurée sur l'hyperviseur *plotbox*.

Sur cette représentation des données, on peut observer que les deux hyperviseurs se comportent plus ou moins de la même manière pour certains scénarios. Les performances sont assez semblables pour les scénarios 0, 1, 6 ou 7. En revanche, on constate une grosse différence sur les scénarios 4, 8 et 9. Les résultats liés aux scénarios 8 et 9 ont été expliqués ci-dessus.

La deuxième métrique est analysée de la même manière que la première. En moyenne, la gestion de la mémoire est mieux optimisée sous ESXi que sous VirtualBox. Un pic est observé sur les scénarios 8 et 9 pour les mêmes raisons que celui observé lors de la gestion de la charge.

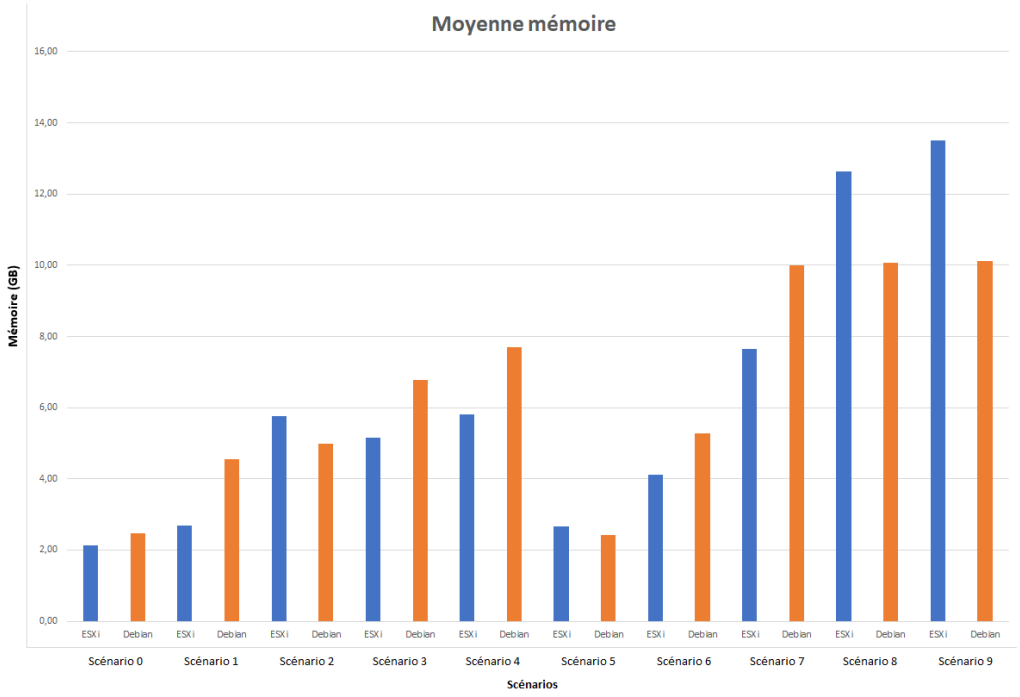


FIGURE 3.6 – Mémoire mesurée sur l'hyperviseur.

Les *plotbox* montrent une gestion de la mémoire totalement différente en fonction du type d'hyperviseur. On remarque sur ce graphique que la gestion de la mémoire varie beaucoup plus sous VirtualBox que sous ESXi.

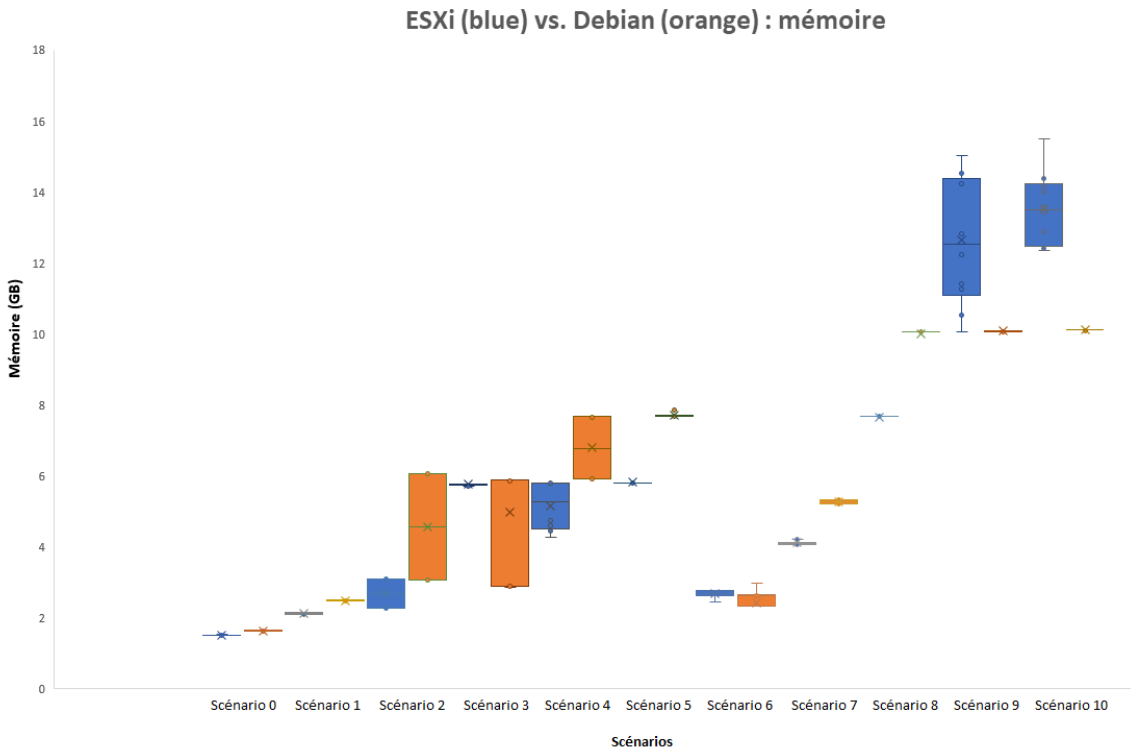


FIGURE 3.7 – Mémoire mesurée sur l'hyperviseur *plotbox*.

La représentation sous forme de *boxplot* étant plus représentative que les bâtonnets de moyenne, les graphiques qui reprennent les valeurs moyennes des deux métriques ont été annexés (D). Seules les *boxplot* seront présentées pour la suite des analyses.

Performance du serveur GNS3

Au niveau de la gestion de la mémoire par le serveur GNS3, on constate que la mémoire est plus ou moins gérée de la même manière sur les deux types d'hyperviseur. La variance est cependant plus élevée lorsqu'on utilise VirtualBox, ce qui signifie qu'il y a une plus grande dispersion des points. Cette dispersion de points peut être le résultat d'un système moins stable puisque les résultats relevés ne sont pas proches.

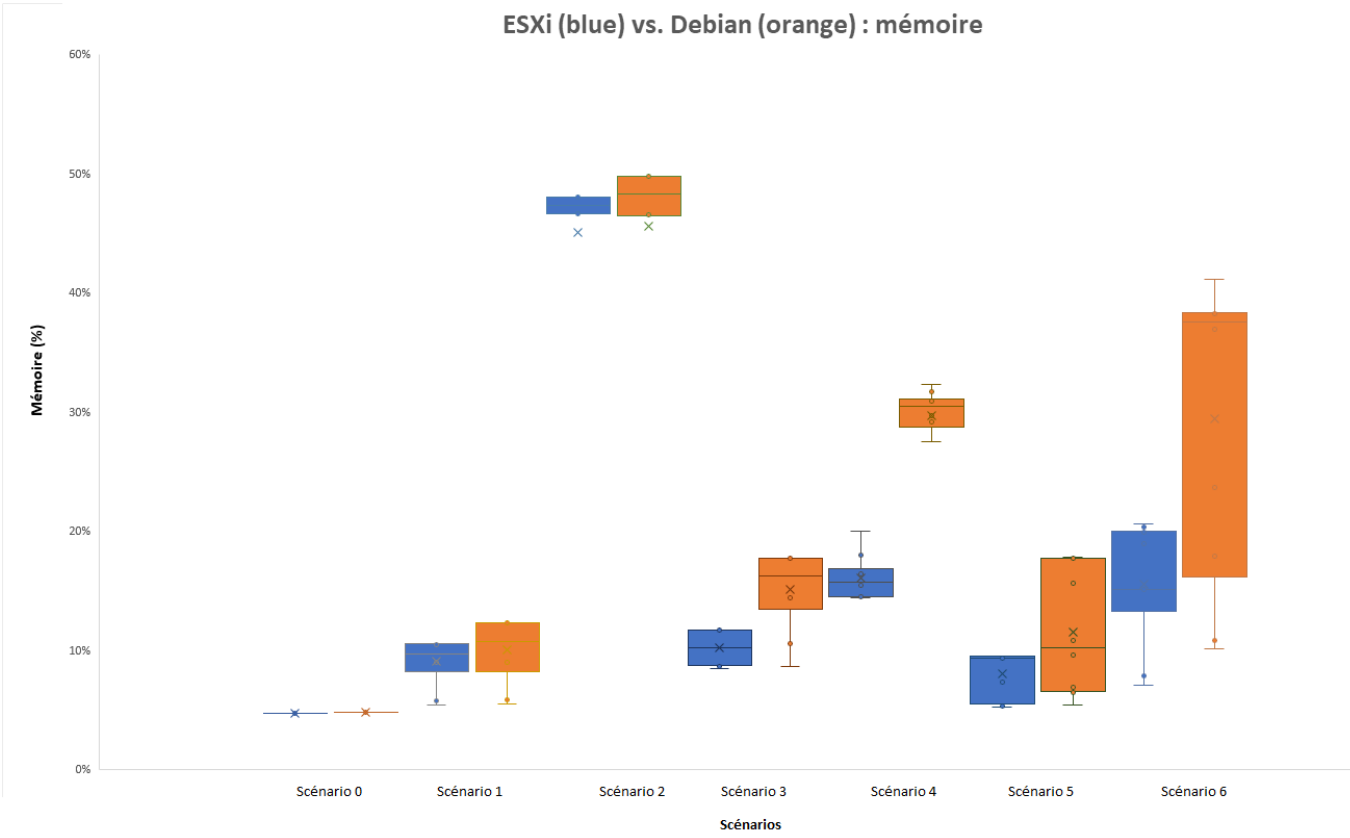


FIGURE 3.8 – Mémoire mesurée sur le serveur GNS3 *plotbox*.

La même remarque peut être faite pour la gestion de la charge par le serveur GNS3. Le relevé des scénarios 3 et 4 montre en revanche une meilleure performance pour l'ESXi.

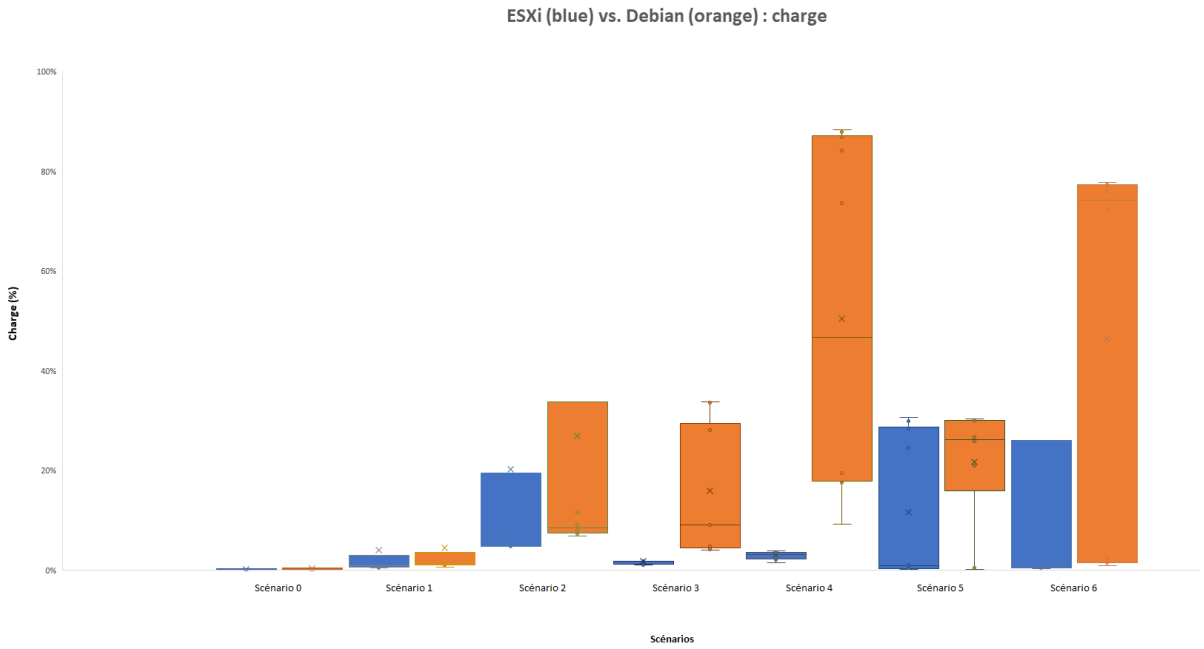


FIGURE 3.9 – Charge mesurée sur le serveur GNS3 *plotbox*.

Les résultats du scénario 7 ne se sont pas montrés intéressants, ils suivent la même logique que le scénario 0. Les résultats n'ont donc pas été inclus dans le graphe suivant. Les résultats du scénario 10 ne font pas non plus partie du graphe puisqu'aucun relevé correct n'a été possible sous Debian. VirtualBox ne gérant pas très bien les *nested VM*.

Plusieurs choses sont à constater pour le scénario 8. La charge est gérée de la même manière par les deux hyperviseurs. En revanche la mémoire ne l'est pas du tout. Les résultats relatifs aux tests de gestion de la mémoire sont assez difficiles à expliquer. Un bref pic sur l'un des hyperviseurs peut en être la cause. Pour le scénario 9, on remarque une petite différence sur la gestion de la mémoire pour le premier ESXi par rapport à la première machine Debian.

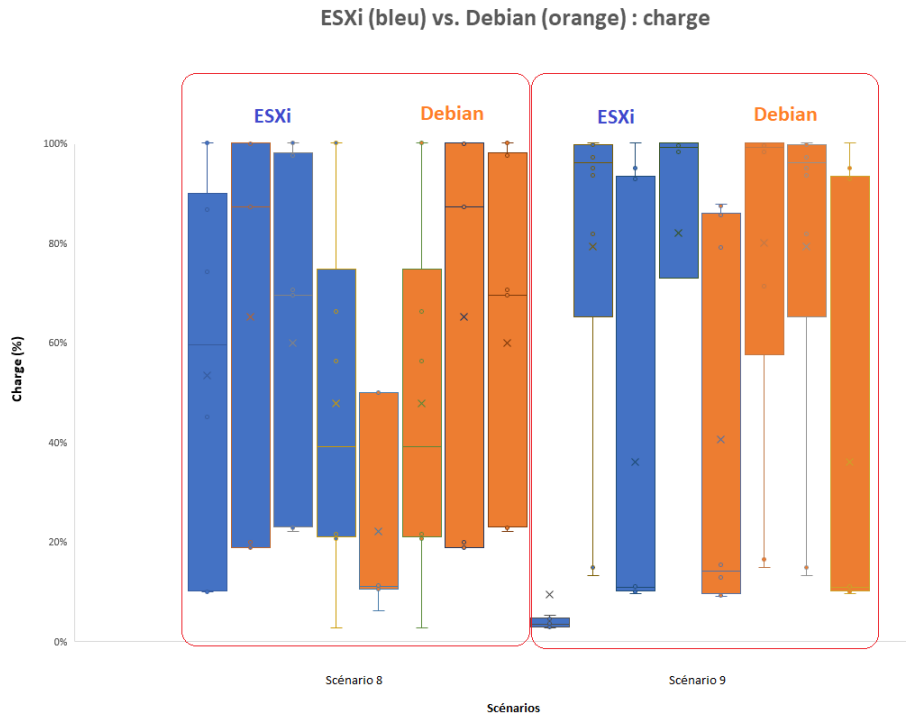


FIGURE 3.10 – Mémoire mesurée sur les 8 serveurs GNS3 *plotbox*.

Malgré ces différences, il est intéressant de remarquer que pour le scénario 8 et le scénario 9, les hyperviseurs de la même catégorie gèrent leurs ressources de manière identique.

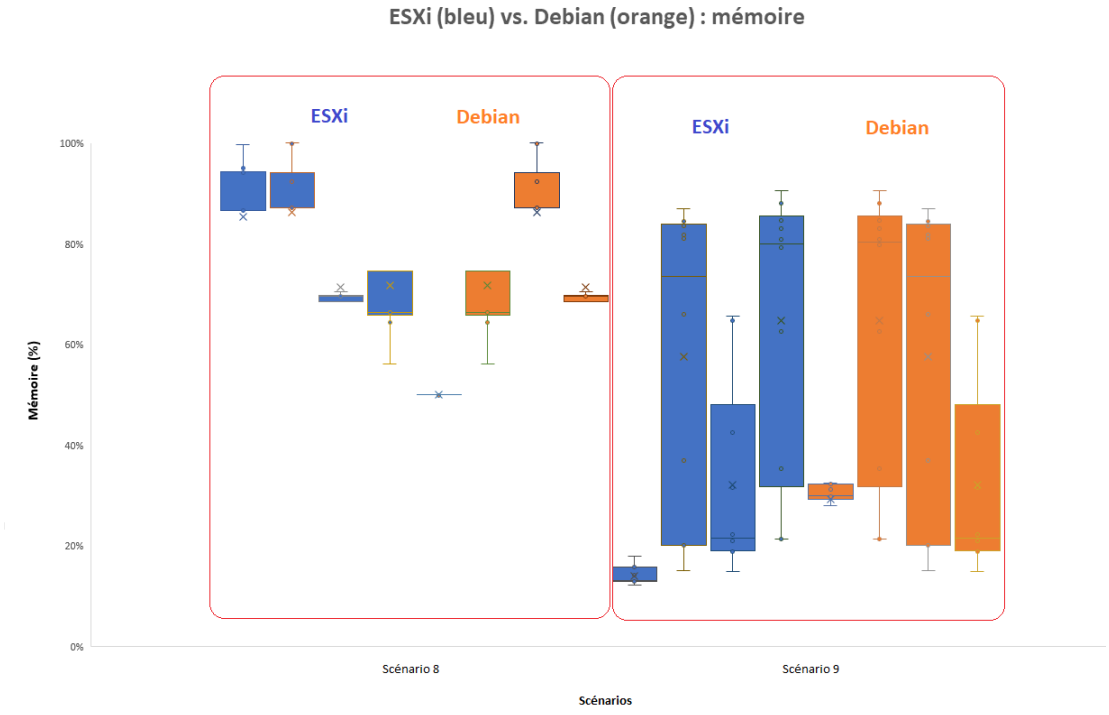


FIGURE 3.11 – Mémoire mesurée sur les 8 serveurs GNS3 *plotbox*.

Notons bien que toutes les mesures précédentes et les analyses qui en découlent, ont été relevées au même moment. Ces résultats sont fiables puisque l'opération a été répétée dix fois. Néanmoins, ces résultats sont soumis à un hasard qui ne peut empêcher qu'un pic survienne sur un des hyperviseurs à un moment différent.

Effet de caching

L'idée qu'il y ait un possible effet de *caching* qui influence les résultats et les performances n'est pas cohérent. En effet, le *timing* entre le démarrage d'une instance et le moment où l'utilisateur peut y accéder a été relevé pour chaque ressource déployée. Il s'avère que ce temps est identique quand une nouvelle resource est déployée et quand une ressource est redémarrée alors qu'elle était déjà présente dans un scénario.

3.2.5 Avantages

Pour de lourdes tâches de virtualisation, à l'heure actuelle et au vu des progrès réalisés technologiquement, il est conseillé d'utiliser un hyperviseur de type 1. VMware s'est imposé dans l'industrie de la virtualisation et la renommée de la firme n'est plus à faire. Au vu des performances relevées ci-dessus, on observe qu'ESXi a une meilleure tolérance avec les *nested VM*. Pour rappel, les serveurs GNS3 qui étaient virtualisés sous VirtualBox ne sont pas parvenus à virtualiser de grosses ressources dans les scénarios GNS3. Les outils de monitoring intégrés directement à ESXi en font une solution qui se démarque d'un hyperviseur de type 2. L'utilisateur peut aisément consulter les ressources utilisées s'il constate un comportement ou une réaction anormale. L'intégration d'une solution de stockage distante (NAS) est plus facile sous ESXi que sous Debian. L'allocation des ressources physiquement disponibles est mieux répartie sur un hyperviseur de type 1 que sur un hyperviseur de type 2. De plus, l'usage d'ESXi pour le laboratoire ne justifie pas une l'achat d'une licence chez VMware. La solution est donc gratuite.

3.2.6 Choix de l'hyperviseur

Sur base des observations et des comparatifs réalisés plus haut, l'hyperviseur de type 1 VMware ESXi a été retenu pour virtualiser le simulateur.

Chapitre 4

Mise en place du laboratoire

4.1 ESXi

Afin de garder une trace de la procédure d'installation de l'hyperviseur, tous les aspects techniques nécessaires au déploiement ont été documentés et sont disponibles dans la partie annexes au point A.

4.1.1 Image *custom*

Les serveurs du labo ne sont pas totalement compatibles avec l'image ESXi fournie par VMware. Après une première installation, les cartes réseaux ne sont pas détectées par l'hyperviseur, empêchant toute interaction avec le serveur. La solution mise en place consiste à intégrer directement les *drivers* des cartes réseaux des serveurs dans l'image ESXi. Ce procédé est réalisé en utilisant PowerShell et VMware CLI. VMware CLI est un outil léger en ligne de commandes basé sur Windows PowerShell. VMware CLI intègre des *cmdlet* pour gérer les produits VMware.

Utiliser VMware CLI oblige de changer le niveau de sécurité lié à l'exécution des scripts dans PowerShell.

```
Get-ExecutionPolicy -list
Set-ExecutionPolicy Unrestricted

Scope ExecutionPolicy
----- ---------------
MachinePolicy       Undefined
UserPolicy          Undefined
Process             Undefined
CurrentUser         Undefined
LocalMachine      Unrestricted
```

Les différents fichiers nécessaires à la création de l'image sont récupérés : image, drivers, packages... Un nouveau profil est créé en se basant sur le clone d'un profil existant. Le niveau d'acceptance doit être défini sur *community* car les drivers qu'on ajoute à l'image

sont signés par la communauté VMware. Les drivers sont ensuite ajoutés à l'image et l'ISO est généré.

```
Add-EsxSoftwareDepot
https://hostupdate.vmware.com/software/
VUM/PRODUCTION/main/vmw-depot-index.xml
Add-EsxSoftwareDepot http://vibsdepot.v-front.de
New-EsxImageProfile -CloneProfile
ESXi-6.7.0-8169922-standard
-name ESXi-6.7.0-8169922-standard-RTL -Vendor ULB
Set-EsxImageProfile -ImageProfile
ESXi-6.7.0-8169922-standard-RTL8111
-AcceptanceLevel CommunitySupported
Get-EsxSoftwarePackage | Where {$_.Vendor -eq "Realtek"}
Add-EsxSoftwarePackage -ImageProfile
ESXi-6.7.0-8169922-standard-RTL
-SoftwarePackage net55-r8168

Export-EsxImageProfile -ImageProfile
ESXi-6.7.0-8169922-standard-RTL
-ExportToIso -filepath
C:\Users\$user\Desktop\ESXi\VMware-ESXi-6.7.0-8169922-RTL.iso
```

4.1.2 Stockage

Le laboratoire de l'ULB est équipé d'un NAS. Une des cartes réseau des serveurs est directement reliée au NAS. ESXi intègre la gestion d'espace disque sur un NAS de manière facile. L'espace dédié à l'ESXi sur le NAS est ajouté en tant que deuxième datastore. Les images et les machines virtuelles créées sont stockées sur le NAS.

Le NAS est relié à l'ESXi en utilisant le protocole iSCSI. Le protocole iSCSI est un protocole qui utilise le protocole IP pour faire du stockage réseau. La création d'un iSCSI LUN sur le NAS se fait avant d'ajouter le *datastore* à l'ESXi (cf. E.5).

Le LUN créé précédemment sur le Synology est ajouté en tant que deuxième datastore sur l'ESXi. Un nouvel adaptateur iSCSI doit être créé au niveau de l'espace de stockage pour permettre la communication entre les deux machines. L'adaptateur iSCSI doit être liée à une interface réseau. Le laboratoire utilise le réseau 172.16.4.0 pour la communication entre les serveurs et le NAS. Cette plage d'adresse n'est pas routable pour des raisons de performances.

La première étape est de créer un switch virtuel. Le MTU dépend des équipements du réseau et doit être configuré en connaissance de ceux-ci. Le nouveau switch doit être visible depuis l'interface web.

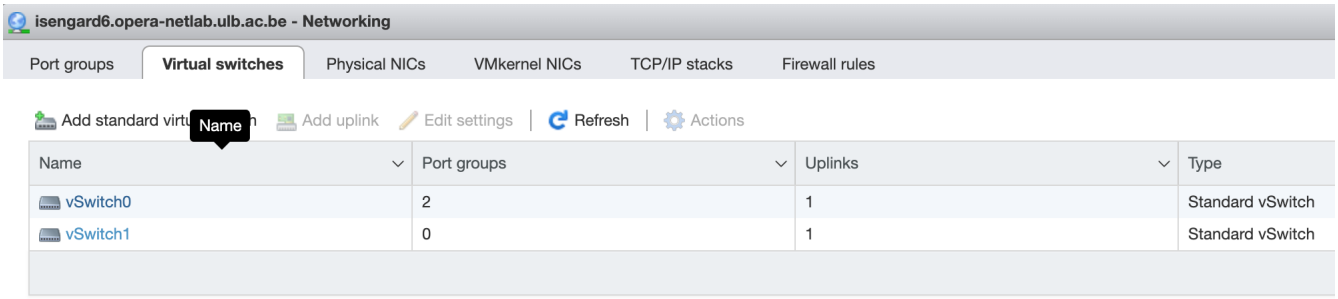


FIGURE 4.1 – Activer la *Switchs virtuels.*

Le nouveau switch virtuel doit être lié à un réseau. Cette liaison se fait dans le *VMKernel Network Interface Controller*. Le réseau du VMKernel NIC correspond ici au réseau sur lequel est notre Synology. Le VMKernel NIC doit être lié sur le bon switch virtuel. Il est possible de spécifier une adresse IP statique lorsqu'il n'y a pas de DHCP sur le réseau (tout en s'assurant que l'adresse que l'on assigne n'est pas utilisée).

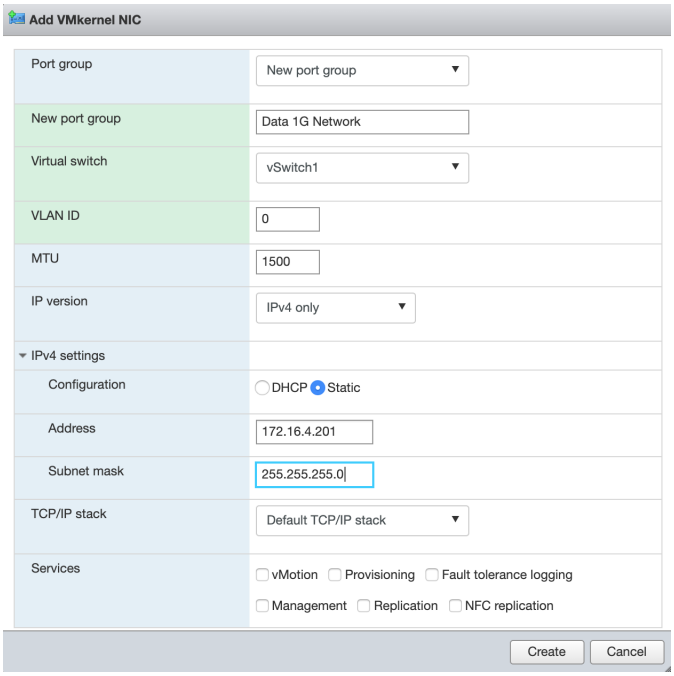


FIGURE 4.2 – Activer la *Configuration VMkernel NIC.*

Le nouveau VMkernel NIC est disponible dans l'interface de gestion. Il est maintenant possible de connecter le NAS Synology à travers l'adaptateur iSCSI.

La configuration diffère en fonction de ce que l'on désire monter. Il est possible de monter des *iSCSI Targets* ou un *iSCSI LUN*. Pour la configuration d'un LUN iSCSI l'administrateur doit ajouter un *datastore*. Le datastore ajouté est le LUN précédemment créé sur le Synology. Le volume doit correspondre à celui réservé sur le Synology. Les options de partitionnement sont soit VMFS5 ou VMFS6. Il est conseillé de sélectionner VMFS5 si on sait que dans le futur, un hôte installé avec la version 6.0 d'ESXi sera amené à se

connecter sur ce datastore.

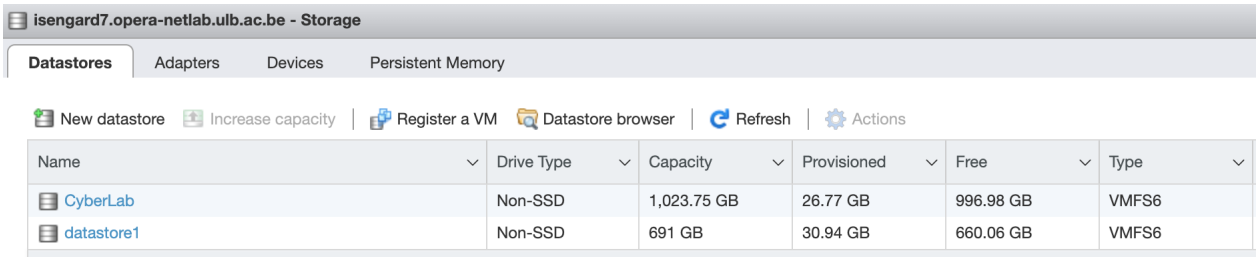


FIGURE 4.3 – CyberLab datastore Synology.

4.1.3 GNS3 pour ESXi

Une appliance de GNS3 pour ESXi est proposée sur le site de GNS3. Le fichier compressé de GNS3 pour ESXi contient quatre fichiers :

- Un fichier qui porte l'extension .mf et qui contient le hash des trois autres fichiers présents dans l'archive. Le hash est utilisé pour valider l'intégrité des fichiers téléchargés.
- Les deux disques virtuels de la machine au format .vmdk
- Le fichier de métadonnées de la machine au format .ovf

L'intégrité des fichiers peut être vérifiée en se basant sur les hash fournis.

KVM doit être activé pour que les *appliances* puissent utiliser QEMU. Pour activer KVM, il faut se rendre dans les paramètres de la machine virtuelle et autoriser la *nested virtualization*, disponible dans les options du CPU.[7] La *full CPU virtualization* permet au système d'exploitation invité de virtualiser des composantes hardwares sans passer par une translation binaire ou par une paravirtualisation.[11]

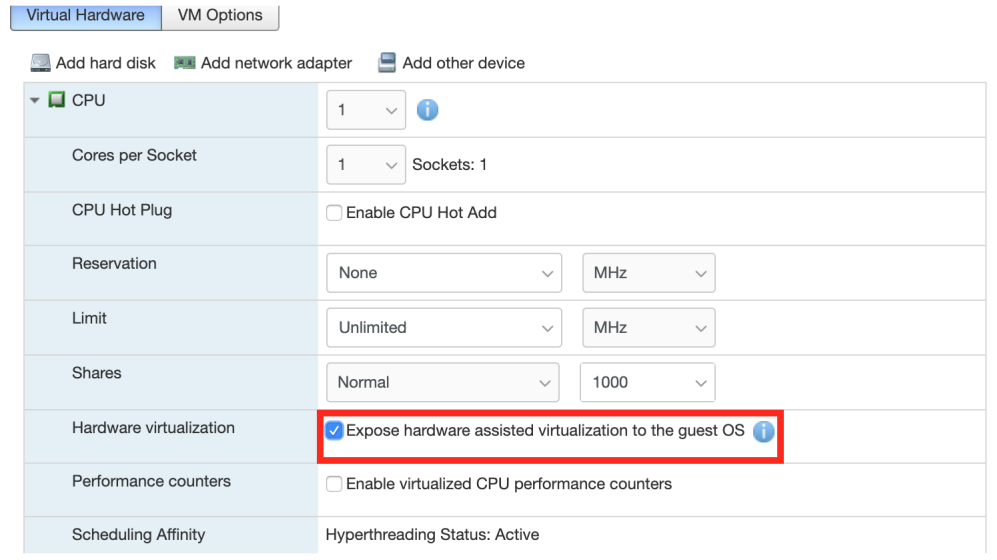


FIGURE 4.4 – Activer la *nested virtualization*.

L'ESXi offre la possibilité de découper une carte réseau physique en différents VSwitch. Il est possible de créer un VSwitch entièrement dédié aux VM GNS3.

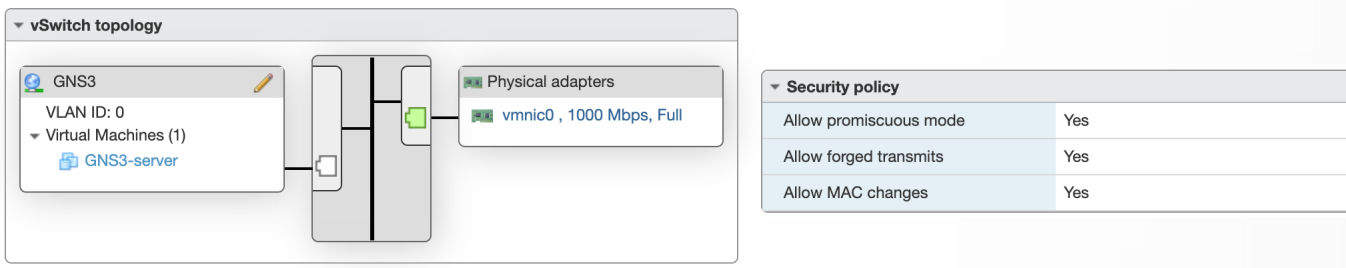


FIGURE 4.5 – *Network policies* GNS3.

En ce qui concerne les différents modes, le mode promiscuité permet à la carte réseau de recevoir toutes les trames même si celles-ci ne lui sont pas destinées. Pour la virtualisation, ce mode permet de faire un pont entre la carte réseau physique et la carte virtuelle créée par l'ESXi.

L'option *Forged Transmissions* affecte le traffic qui est transmis depuis une machine virtuelle. Si l'option est définie sur la *policy* accept, l'ESXi ne compare pas la source avec la véritable MAC adresse.[12]

L'option *MAC changes* affecte à l'inverse le traffic qui est destiné à une machine virtuelle. Si cette option est définie sur accept, l'ESXi accepte les demandes de modification de l'adresse MAC effective en une adresse autre que l'adresse MAC initiale.[13]

4.2 Allocation des ressources sur base des tests précédents

Une division des ressources de manière efficace permet aux étudiants de pouvoir effectuer les manipulations sans craindre un problème matériel. Les ressources du laboratoire ont été complètement réinstallées sur base des observations faites lors de la relève des métriques. Les valeurs qui ont été retournées par les différents équipements ont premièrement permis de définir la meilleure solution entre l'hyperviseur de type 1 et l'hyperviseur de type 2.

L'interprétation des résultats montre que VirtualBox en tant qu'hyperviseur de niveau 2 gère les ressources de manière moins optimisée lorsqu'on intègre des *nested VM* dans les scénarios GNS3.

Sur base de ces observations, les *blades* présents dans le laboratoire ont été réinstallés sous VMware ESXi. La découpe des ressources des serveurs a été pensée en fonction du nombre de participants qu'il pourrait y avoir pour suivre les laboratoires du cours, mais aussi de l'importance des scénarios qu'ils auront à réaliser.

Cette découpe s'est faite sur base des critères suivants :

- Nombre de personnes minimum
- Nombre de personnes maximum
- La formation de groupe
- vCPU disponibles
- Mémoire disponible
- La gestion des ressources de l'hyperviseur
- La gestion des ressources sous GNS3 serveur
- Les résultats des tests de métriques
- Quelques suppositions

Le nombre de personnes minimum reprend le nombre de personnes pour lesquelles les serveurs du laboratoire ne doivent avoir aucune difficulté à faire tourner les ressources demandées. Ce nombre a été fixé à 12 personnes.

Le nombre maximum de personnes reprend le nombre de personnes qui peuvent être admises à utiliser les ressources du laboratoire de manière simultanée sans provoquer de problème sur l'infrastructure. Ce nombre a été fixé à 16.

La formation de groupes suppose que les étudiants seront regroupés par 2, permettant dans le cas où un nombre d'élèves trop important se présenterait aux séances de travaux pratiques, une continuité du travail.

Le nombre de CPU virtuels regroupe le nombre de vCPU disponibles par serveur. Ce nombre a été évalué à 64 coeurs, chaque serveur disposant de 16 coeurs.

La mémoire disponible a été évaluée à 64GB pour les quatres serveurs, chaque serveur disposant de 16 GB.

La solution est hybride, elle permet donc l'interconnexion du matériel de l'étudiant avec les ressources du laboratoire. Afin de ne pas surcharger les serveurs, chaque gain de ressources est à prendre. C'est dans cette optique que la machine virtuelle PTF sera mise à disposition sur un *share folder* pour les étudiants. Les étudiants qui ont déjà une machine de pentesting ne seront pas obligés de la télécharger. Dans le cas extrême où l'étudiant ne saurait faire tourner cette machine virtuelle, celle-ci sera disponible et pourra être exécutée directement à l'intérieur de la topologie GNS3.

Voici un récapitulatif de l'infrastructure physique du laboratoire :

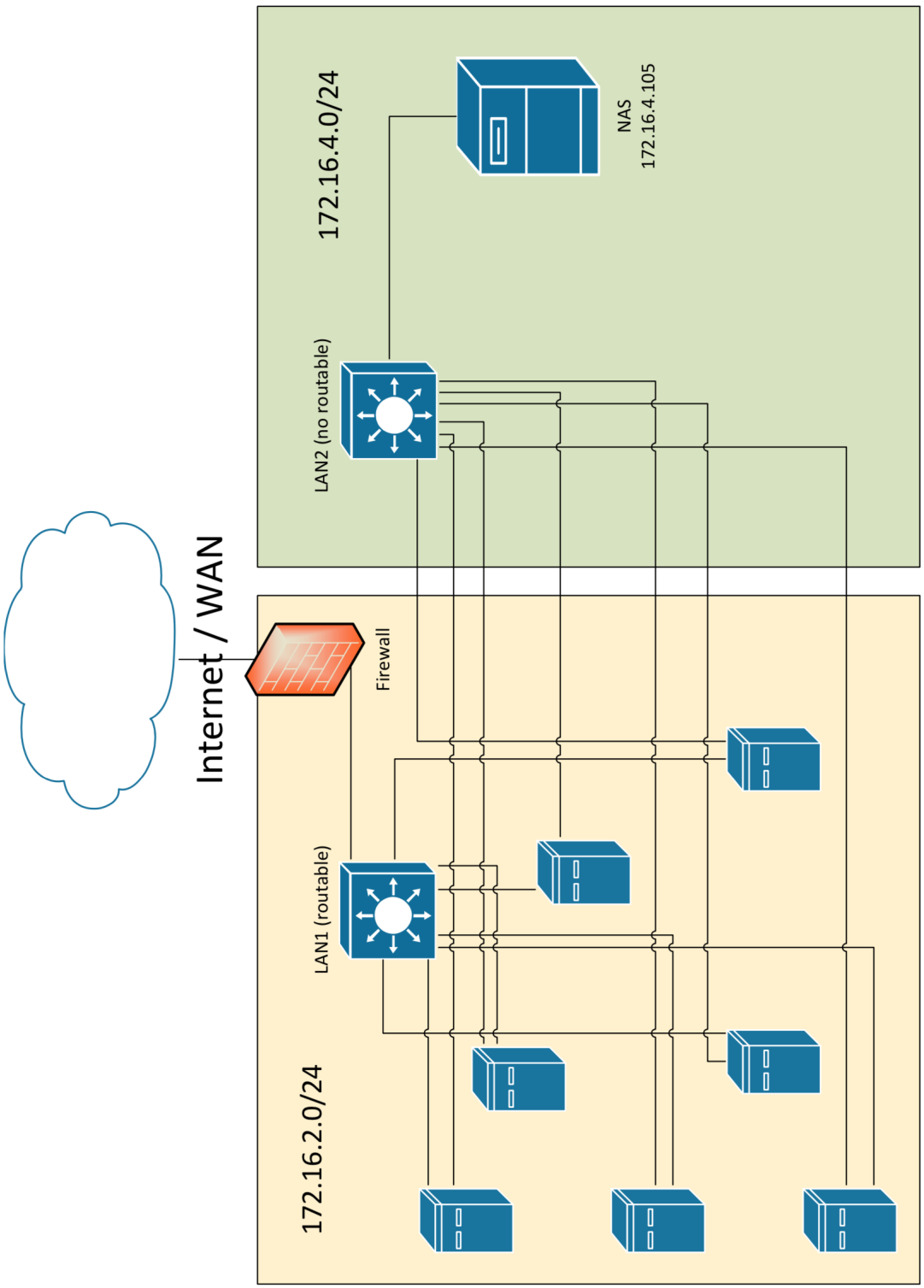


FIGURE 4.6 – Infrastructure physique du laboratoire.

Sur base des observations énumérées ci-dessus, voici la topologie logique du laboratoire :

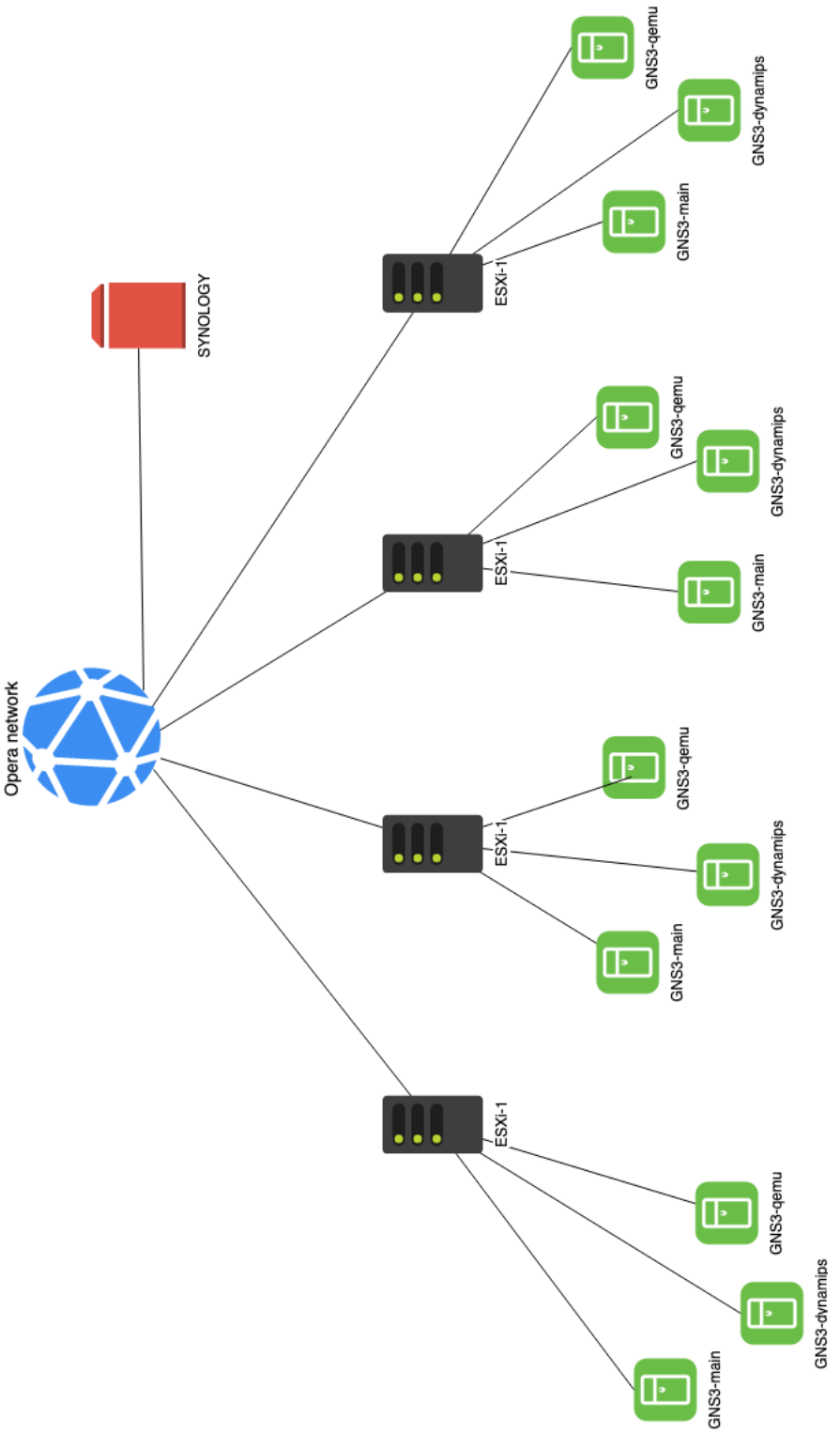


FIGURE 4.7 – Topologie laboratoire Opera pour les manipulations sous GNS3.

Les quatre serveurs ESXi sont connectés au réseau du laboratoire. Ces serveurs hébergent trois serveurs GNS3 virtualisés. Le premier serveur GNS3 est le serveur principal. Le second est utilisé pour émuler les routeurs en utilisant Dynamips. Le dernier est dédié à la *nested virtualization*.

	vCPU	RAM	Remarques
Hôte	16	16	Tous les serveurs sont identiques et se composent de 16 CPU virtuels et 16 GB de mémoire RAM.
GNS3-main	4	4	Le main serveur est celui qui nécessite le moins de ressources. Il génère la topologie et gère les actions. Il faut cependant lui en attribuer assez pour qu'il puisse gérer plusieurs projets simultanément.
GNS3-dynamips	6	8	Un routeur consomme 256 MB de mémoire (maximum). En attribuant 8 GB on permet le fonctionnement de 24 routeurs de manière simultanée. La consommation CPU n'est pas très importante d'après les métriques relevées.
GNS3-qemu	6	6	Il s'agit de la machine qui doit avoir le plus de puissance pour faire tourner les machines virtualisées dans GNS3.

Chapitre 5

Scénarios

Les scénarios de ce travail sont basés sur la matière théorique du cours de *Network Security*. Différents concepts relatifs à la cybersécurité sont présentés dans chaque scénario. La liste de ces scénarios n'est pas exhaustive et ne couvre pas toutes les thématiques qui ont été vues en théorie. Chaque scénario couvre un sujet spécifique.

La procédure technique qui sera remise aux étudiants pour découvrir GNS3 est disponible dans les annexes au point B.

5.1 Scénario 1

5.1.1 Objectifs

L'objectif de ce scénario est de familiariser les étudiants aux techniques de routage et à la configuration des équipements réseaux.

5.1.2 Compétences à acquérir

A la fin de ce labo, l'étudiant devra avoir acquis les compétences et savoirs suivants :
- Comprendre une topologie réseau
- Maîtriser le simulateur
- Interconnecter des éléments de couches différentes
- Interconnecter des réseaux physiques et des réseaux logiques
- Configurer un routeur
- Mettre en place un protocole de routage
- Localiser et résoudre un problème réseau

5.1.3 Déroulement

Cet exercice se déroule en trois parties. Premièrement l'étudiant doit se familiariser avec le simulateur et avec les technologies utilisées pour la suite des exercices. Par la suite, il devra interconnecter un réseau physique avec un réseau logique créé sous GNS3.

Le but est de permettre une communication parfaite entre les différents équipements. L'étudiant devra ensuite configurer quelques routeurs pour permettre à des clients qui se trouvent dans des réseaux différents, de pouvoir se joindre. La configuration des routeurs comprend la configuration des interfaces et du protocole de routage. Une fois que l'étudiant se sera familiarisé avec le matériel, une topologie lui sera remise. Son but sera de déterminer pourquoi les clients n'arrivent pas à se joindre. La solution se trouve dans la configuration des routeurs.

5.1.4 Scénario

Network security

ELEC-H-504

Introduction

This first lab aims to be more familiar with the tools used for this course. GNS3 is the solution chosen to run instances and make cybersecurity scenarios. The resources you will need are available on the share folder (ask to the teacher). Make sure you download the right version for your computer.

The benefits for using GNS3 is to allow students who have not enough power resources to use the servers computation power of the lab. You normally should receive the **GNS getting started** guide.
You are invited to use the computation power if you have not enough resources.

The sessions will take place by group of two students. Please, insert the IP address of the main server which is assigned to you.

The lab sessions will cover different subjects seens in the theoretical course like routing, firewalling, ARP spoofing, DNS and vulnerability exploitation.

Getting started

Once you are connected to your resources pool the first goal is to get more familiar with GNS3 environment.

GNS3 is an hybrid emulation platform solution. It means all virtualized equipments can communicate with the physical equipments present in the lab LAN.

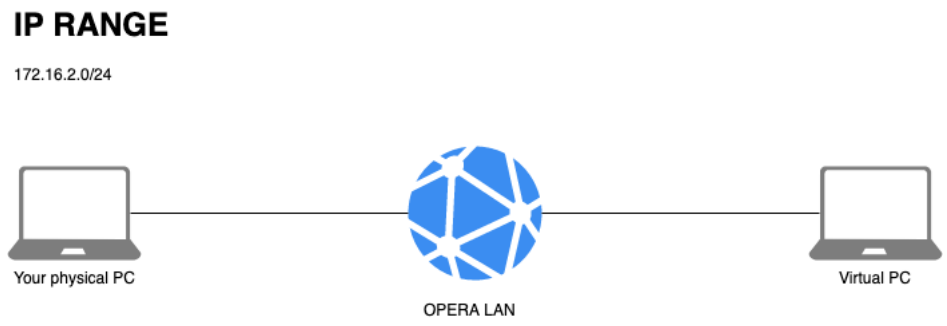


Figure 5.1 – Hybrid connection

The node cloud in GNS3 allows the connection between the virtual and the physical network. For the first test, try to make a simulation where your physical device can communicate with a virtual PC (cf. Figure 1).

Router configuration

Using the router resources available on GNS3 and the virtual PCs, build this following topology.

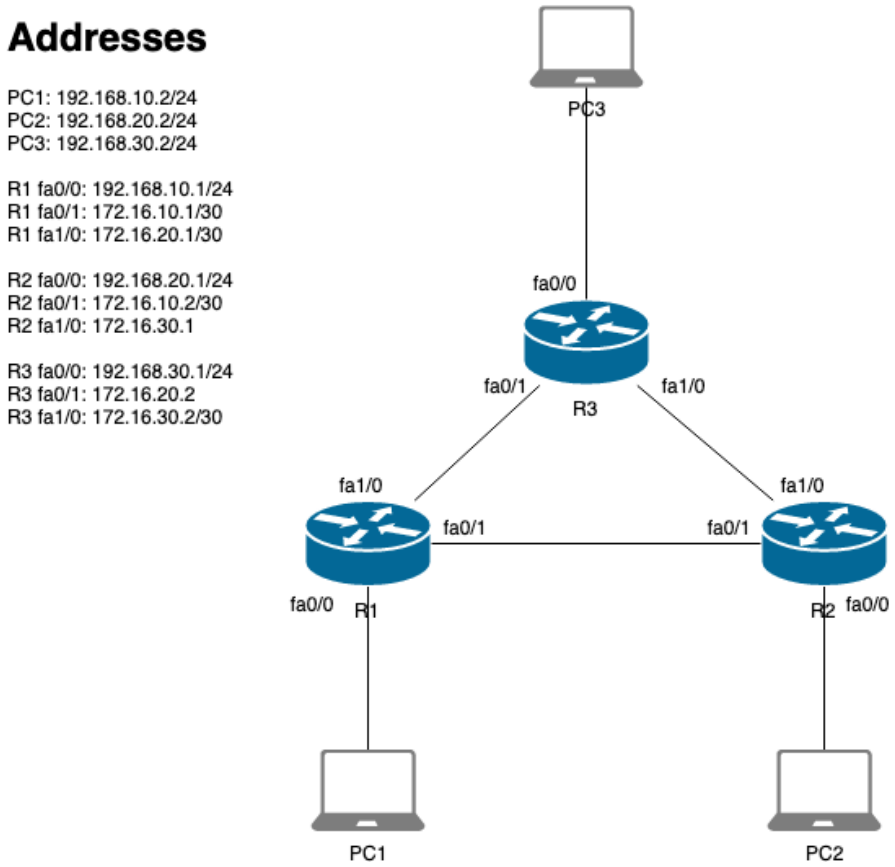


FIGURE 5.2 – Router initiation

At the end of the configuration of the equipments, PC1, PC2 and PC3 would be able to ping each other.

Follow this methodology to configure the routers. Autocompletion works in Cisco ios. All commands are in bold.

Enter in the enable mode.
enable
This mode can be used to see information on the router like routing table or all configuration.

Enter in the configuration mode.
configure terminal
In this mode you can configure network interfaces, routing procotols and many other options.

Access ethernet interface for configuration.
interface fastEthernet x/x

When you are on the desired inteface, assign IP address and netmask
ip address x.x.x.x x.x.x.x

Execute the command to no shutdown the interface.
no shutdown

Repeat operation for all router interfaces.

Once all interfaces are configured you need to set up the routing protocol. Chosen routing protocol is RIPv2 for its simplicity. Go back in the configuration mode.

Specify the routing protocol.
router rip

Because RIP has more than one version you need to specify the version used. **version 2**
Add all networks described on the topology.
network x.x.x.x

After this configuration on each router all PCs should be able to ping any other PCs.

Routing

Prerequisite : you must have read the Cisco Memento.

For this point you need to import the GNS3 file located on the share folder in /lab/scenario1-routing.gnsa3.

The imported topology has an issue. Virtual PC which is in the zone 2 and zone 3 cannot communicate with the physical network (zone 1).

Your mission is to troubleshoot and find the problem. All PC have to be reachable from the physical network and from the different zones. Information needed are put on the diagram below.

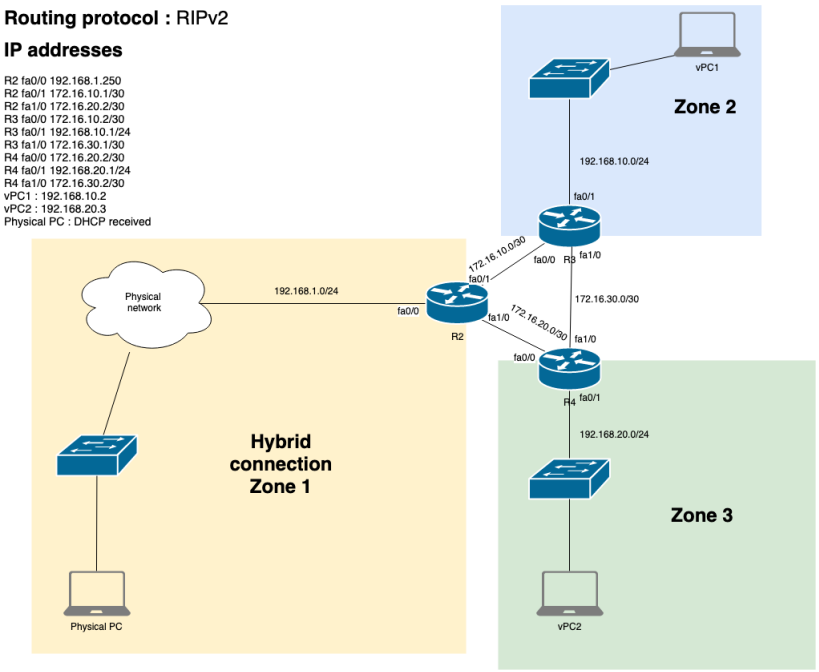


FIGURE 5.3 – Troubleshooting network

Hints

— Routing tables of the routers can be displayed by typing
— Configuration of routers can be displayed by typing
— You can add ip address on the router interface
— Perhaps, OS firewall blocks ping requests

Personal notes

5.2 Scénario 2

5.2.1 Objectifs

L'objectif de ce scénario est d'utiliser les compétences apprises dans la première scéance et d'y ajouter la configuration d'un *firewall*. Le *firewall* doit être configuré en utilisant la *best-practice whitelisting*.

5.2.2 Compétences à acquérir

A la fin de ce labo, l'étudiant devra être capable de configurer un réseau entier en y intégrant des *firewalls*. L'étudiant sera également apte à déterminer la meilleure manière de configurer un *firewall*.

5.2.3 Déroulement

L'étudiant est amené à dessiner une topologie d'entreprise classique. Cette topologie se compose d'un accès WAN, d'une DMZ et de deux sous-réseaux privés. Des machines simulant un serveur mail et un serveur web sont placées dans la DMZ. L'étudiant doit configurer les différents équipements réseaux pour minimiser au maximum le risque d'une fuite de données ou d'une cyber attaque.

5.2.4 Scénario

Network security

ELEC-H-504

Introduction

For this lab you will reuse the topology created in the first lab. On the basis of the first topology you will integrate (some) firewall(s).

The goal is to provide to your company a fully trustable infrastructure where "less trustable" components have to put in a DMZ.

For the example, just use virtual PC to play the role of the potential servers.

Topology

Normally, you should have a topology similar to the topology below. If you have other topology in mind do not hesitate to test your solution.

Entities

To draw this topology you will need routers (same than in first scenario), virtual PC and firewall.

You will use pfSense as firewall. They are available in GNS3 resources. The most part of the policies configuration for the firewall are done through the web interface of pfSense.

Do not hesitate to try to apply rules to realize concrete actions and impacts on the network.

Hints

Before accessing the web interface of pfSense you need to configure the firewall. The first configurations have to be made using the CLI of the machine. The CLI is accessible from the terminal. If you are running Windows you probably should modify some configuration in GNS3 to lunch a terminal client with the resource (Putty for example).

When you have finished the basic configuration the firewall should be accessible from the browser. If it does not work, be sure you are in the same network. You can also access the web interface from your physical machine if you configured properly the cloud node and the different gateways.

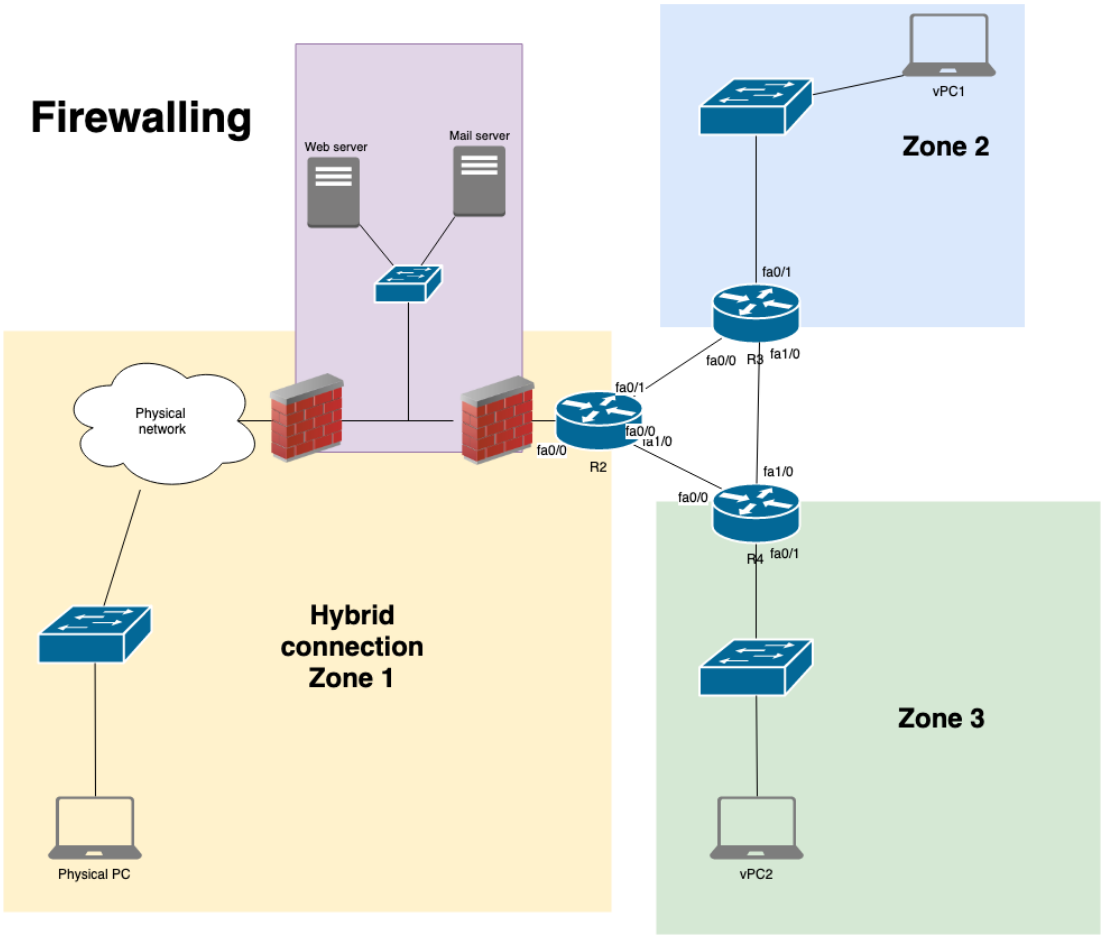


FIGURE 5.4 – Network topology

Personal notes

5.3 Scénario 3

5.3.1 Objectifs

L'étudiant est mis face à une machine qui est remplie de vulnérabilités. Le but de cette manipulation est d'amener l'étudiant à utiliser des outils incontournables dans la recherche de vulnérabilités. L'étudiant aura l'occasion de prendre en main des outils de découverte et d'exploitation.

5.3.2 Compétences à acquérir

A la fin de ce labo, l'étudiant devra être capable d'utiliser des outils de reconnaissance pour déterminer les services qui tournent sur une machine. Il sera également capable :

- d'utiliser le framework Metasploitable et de mener un *exploit* du début à la fin sur trois services différents.
- d'escalader les privilèges afin de devenir le super utilisateur d'une machine.
- de revéler des informations auxquelles il n'a normalement pas accès.
- de tirer profit d'une faille CSRF.
- de tirer profit d'une faille XSS.
- de tirer profit d'une injection SQL.
- d'utiliser une URL afin de révéler des informations sensibles.
- d'exécuter une attaque FTP.

5.3.3 Déroulement

Toutes ces attaques seront exécutées sur une machine virtuelle qui a été importée au préalable dans GNS3. Pour ceux et celles qui ne disposent pas des outils de pentesting sur leur ordinateur et qui ne savent pas l'installer, une machine virtuelle créée pour l'occasion et qui regroupe de nombreux outils est disponible dans GNS3[21].

L'accès à ces machines se fera en utilisant un client VNC (au choix).

5.3.4 Scénario

Network security
ELEC-H-504

Introduction

This lab aims to make you more familiar with some pentesting tools.

Your job is to perform and understand the following vulnerabilities on the Metasploitable machine.

- Exploit : find and exploit three different running services.
- Escalation privileges : you are a simple user. Your goal is become a super user !
- Leaking information : try to leak information by all possible means (e.g. display *passwd* file content).
- Execute a CSRF vulnerability.
- Perform an XSS attack.
- Perform a SQL injection.
- Divert the main goal of URL...
- Find and exploit a FTP vulnerability

Topology

No need specific topology for this lab. Just drag and drop the Metasploitable resource contained in GNS3.

If you do not have enough computer power on your laptop you can use the lab resource. A Linux virtual machine named *PTF* is available in GNS3. This machine embeds all necessary pentesting tools to perform the above tasks.

If you have enough computer power, just put the Metasploitable machine on an hybrid node to have access from your physical computer.

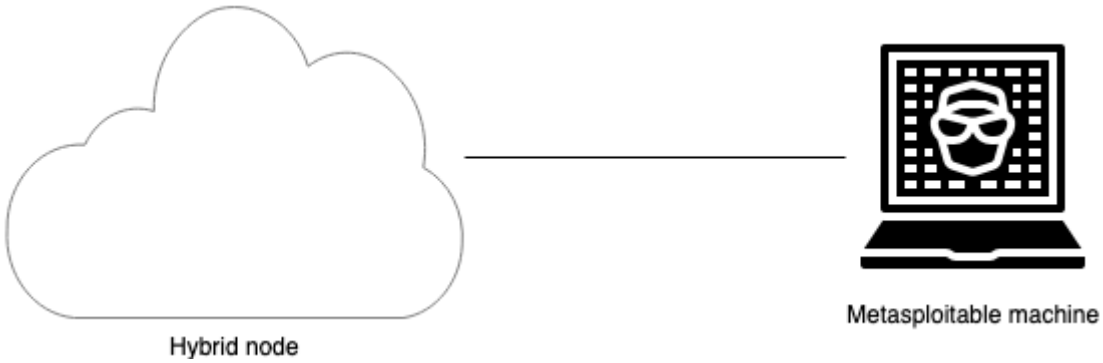


Figure 5.5 – Network topology

Personal notes

5.4 Scénario 4[15]

5.4.1 Objectifs

L'objectif de ce scénario est de montrer à l'étudiant comment fonctionnent les attaques de type *DHCP et ARP spoofing*

5.4.2 Compétences à acquérir

A la fin de ce labo, l'étudiant devra pouvoir utiliser les outils nécessaires pour effectuer une attaque de *DHCP et d'ARP spoofing*. L'étudiant saura manipuler les flux réseaux afin de rediriger le trafic vers sa machine.

5.4.3 Déroulement

Pour réaliser cette manipulation, l'étudiant sera amené à déployer un réseau virtuel comprenant un client, un serveur et une machine destinée à faire du *man in the middle*. Le premier type d'attaque à mettre en place est le *DHCP spoofing*. L'étudiant devra prendre en main l'outil *Etercap* afin de changer la passerelle par défaut du client pour rediriger le trafic ciblé vers sa machine.

La deuxième étape repose sur la même topologie. Un client, un serveur et un utilisateur malicieux. Le but de l'étudiant est, en utilisant *Etercap*, de faire de l'*ARP spoofing*. La technique d'*ARP spoofing* aura le même but que la technique du *DHCP spoofing*. En faisant croire à la victime que l'utilisateur malicieux possède l'adresse IP qu'elle essaye de joindre, le flux de données sera envoyé vers la machine malicieuse.

5.4.4 Scénario

Network security
ELEC-H-504

Introduction

During this lab you will learn to perform spoofing attacks. More precisely DHCP spoofing and ARP spoofing attacks.

You will need *Etercap* tool to perform these attacks. You can download the tool inside the VM using the node cloud.

Topology

The topology should have three participants. The victim machine, the malicious machine and the target server. In our case, the victim machine makes requests to a webserver located outside the LAN. Obviously, the trafic is first forwarded to the gateway of the network.

For DHCP spoofing, the goal is to spoof DHCP trafic in order to change the network configuration of the victim. DHCP uses the UDP protocol to send requests to and from a DHCP server. There are four types of packets when the connection is negociated :

- Discover : sent to all nodes on the IP network to discover a DHCP server.
- Offer : sent by the DHCP server to the node that sent the discovery packet to offer a network configuration.
- Request : send by the originating node to confirm its acceptance of the offer.
- Acknowledgement : send by the server to confirm completion of the configuration.

Etercap will spoof the DHCP configuration process to redirect trafic.

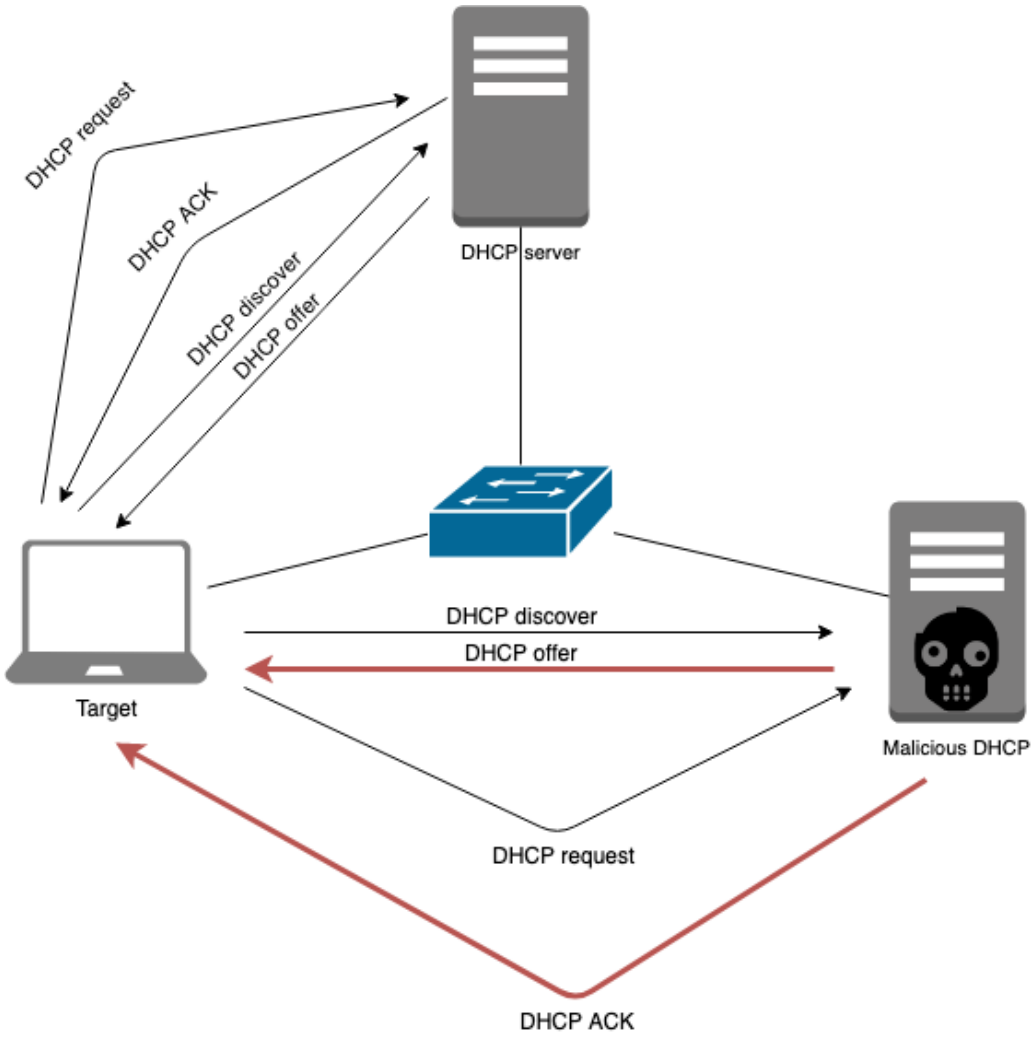


FIGURE 5.6 – How DHCP spoofing works

ARP spoofing has the same goal than DHCP spoofing : change the destination trafic. *Etercap* can also be used to perform ARP spoofing (based on the same topology).

Hints

If you try to forward communication on your machine, think to configure routing !

A successful attack has to contain acknowledgement DHCP spoofing log. The successful log looks like **DHCP spoofing : fake ACK [some numbers] assign to [IP]**.

Personal notes

5.5 Scénario 5[30]

5.5.1 Objectifs

Le but de ce scénario est de montrer à l'étudiant comment tirer parti d'applications qui n'ont pas été mises à jour par l'utilisateur. La machine cible est une machine Windows 7 qui fait tourner un serveur web et une base de données SQL. Les versions choisies de la base de données et du serveur web sont volontairement obsolètes.

Le but de cette manipulation est de familiariser l'étudiant avec les outils de pentesting utilisés pour identifier les vulnérabilités web.

5.5.2 Compétences à acquérir

A la fin de ce labo, l'étudiant sera capable d'identifier les vulnérabilités d'une application web. Il sera également capable d'utiliser des logiciels de pentesting de vulnérabilités web tels que *Burp, SQLMap, Beef....*

Pour terminer, il saura effectuer des injections ou tirer profil de failles telles que *les local file inclusion, injections SQL, XSS, XPath injection....*

5.5.3 Déroulement

Pour ce scénario, une machine virtuelle préconfigurée sera démarrée sur l'hyperviseur. Il ne s'agit pas d'exécuter la machine dans GNS3 pour ce scénario, la machine étant assez lourde, les performances seront meilleures à même l'hyperviseur. Les étudiants travailleront par groupe et l'adresse IP de la machine Windows 7 sera donnée au groupe.

Les étudiants utiliseront leurs connaissances afin de tirer parti des vulnérabilités présentes sur la machine. Pour les étudiants qui n'ont pas la puissance nécessaire sur leur machine, la machine PTF disponible sous GNS3 pourra être utilisée en connexion hybride.

5.5.4 Scénario 5

Network security
ELEC-H-504

Introduction

This manipulation is focused on the damages that could be caused by non updated applications. The resource used is a Windows 7 machine which hosts a web server with a SQL database. You do not need GNS3 for this manipulation.

Your goal is to break the security of the web server and stole database information. You can access the website on the IP given by the teacher. The website is available in the LAN lab.

Look at the Hints section for more information about the existing vulnerabilities and possible exploitations.

Topology

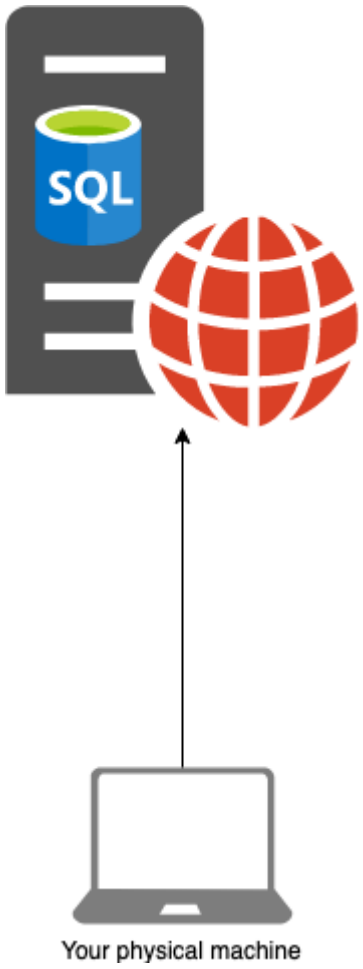


Figure 5.7 – Web server topology

Hints

— Try to find version of the application used.
— You can find related vulnerabilities.

— Do you know XPath Injection and Local File Inclusion (LFI) ?

— Burp could be your friend.

— Perhaps you could try SQL injection or SQLMap ?

Personal notes

Chapitre 6

Conclusion

Les techniques de virtualisation ont évolué, entrainant avec leur évolution une remise en question des méthodes actuelles utilisées. Le but de ce travail porte sur l'élaboration d'un simulateur hybride qui pourrait accueillir des scénarios d'entrainement pour les étudiants du nouveau master en cybersécurité. Afin de mener cette tâche, les solutions actuellement utilisées ont été comparées à d'autres solutions.

Au niveau du simulateur, les recherches effectuées ont montré la limite d'Hynesim en tant que solution gratuite, privilégiant le déploiement de GNS3. La virtualisation se faisait sur un système Debian, sans interface graphique (pour maximiser le gain de performances) en utilisant un hyperviseur de niveau 2 : VirtualBox. VirtualBox a également montré ses limites lors de la virtualisation de machines virtuelles dans une machine virtuelle (*nested virtual machine*). Les analyses statistiques ont montré que cette solution était moins stable dans certaines situations qu'un hyperviseur de niveau 1. Pour cette raison, et pour la stabilité relevée dans les scénarios, un hyperviseur de niveau 1 a été choisi pour remplacer l'hyperviseur de niveau 2.

La solution mise en place doit être à la fois portable et flexible. Grâce à GNS3 et aux instances spécialement créées pour l'entrainement des élèves dans le cours de *Network Security*, les scénarios peuvent être exportés. L'utilisateur a le choix d'exporter soit sa topologie sans intégrer les images des machines, soit la topologie en intégrant les images de machines. Un scénario peut être exporté en intégrant les machines qui le composent. Les machines ont été créées spécialement dans le but de consommer le moins de ressources possibles, favorisant ainsi le temps d'exportation.

Une fois le simulateur et l'hyperviseur mis en place, une partie du travail consiste à imaginer des scénarios dans lesquels l'étudiant pourrait mettre en exergue les compétences enseignées lors du cours théorique. Cinq scénarios différents ont été pensés sur base des notions théoriques et sur base d'une expérience personnelle académique et professionnelle. Ces scénarios ont été créés en prenant en compte une méthodologie d'enseignement enseignée à l'Université Catholique de Louvain.

Les ressources déployées répondent aux besoins qui avaient été identifiés au début de ce

travail.

Pour conclure, ce travail a abouti à la mise en place d'un simulateur et d'un hyperviseur dans le laboratoire de l'ULB sur lequel se trouve des scénarios d'entrainement, qui ont été spécialement conçus pour les étudiants du master en cybersécurité.

Chapitre 7

Déliverables

Ce mémoire est accompagné de tous les logiciels, images, IOS, licences, appliances et ressources qui ont été nécessaires à sa construction. Ces ressources comprennent :

- L'image VMWare ESXi.
- L'image *custom* d'ESXi faite pour les serveurs de l'ULB.
- La licence VMWare ESXi destinée à l'ULB.
- La machine virtuelle vierge GNS3 destinée à l'ESXi.
- La machine virtuelle GNS3 configurée et prête à être déployée. Celle-ci contient les scénarios et les différentes images des équipements.
- L'IOS du routeur intégré à GNS3.
- L'appliance pfSense du market place GNS3.
- L'image pfSense version 2.4.4.
- La machine virtuelle PTF.
- Le client GNS3 Windows compatible avec la version de GNS3 de l'ESXi.
- Le client GNS3 MacOS compatible avec la version de GNS3 de l'ESXi.
- Le client GNS3 Linux (archive GitHub) compatible avec la version de GNS3 de l'ESXi.
- La machine Metasploitable utilisée pour les travaux.
- Les différents scénarios destinés aux étudiants au format LaTeX.
- La documentation d'ESXi destinée aux assistants, au format LaTeX.
- La documentation de GNS3 destinée aux assistants, au format LaTeX.

Chapitre 8

Travaux futurs

Une des contributions futures basée sur ce travail serait, en fonction du temps, d'analyser les nouvelles solutions qui seraient susceptibles de permettre une nouvelle manière d'apprentissage pour les étudiants. Une *review* des solutions existantes qui comparerait celles-ci à des innovations technologiques pourrait se montrer bénéfique pour l'enseignement.

Ce travail reprend quelques scénarios. Ces scénarios ne sont pas exhaustifs. La création de nouveaux scénarios qui utiliseraient la capacité hybride de la plateforme serait intéressante.

Un partenariat avec l'Ecole Royale Militaire ayant pour but l'intégration d'une solution homogène entre leur CyberLab et celui de l'ULB peut être envisagée.

Pour terminer, il est possible d'imaginer l'intégration de Metasploitable 3 dans un scénario GNS3. Metasploitable 3 utilise de nouvelles techniques pour générer une image. L'intégration dans GNS3 pourrait être considérée comme une nouvelle contribution afin de tenir la solution à jour.

Annexe A

Prise en main d'ESXi

Cette section de l'annexe est dédiée à la prise en main de la solution ESXi. La documentation est écrite en anglais.

Getting started ESXi

Pierre Michaux

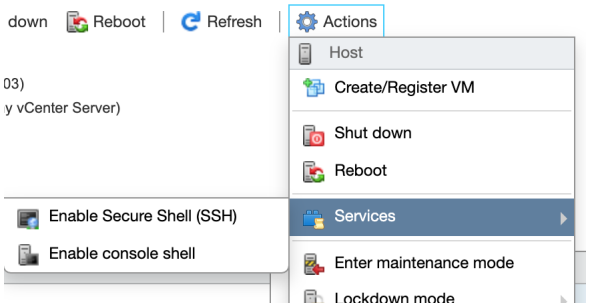
ESXi is a virtualization plateform developped by VMWare. This is a guide explanation of how this solution works.

Credentials to log in are: assistant / password

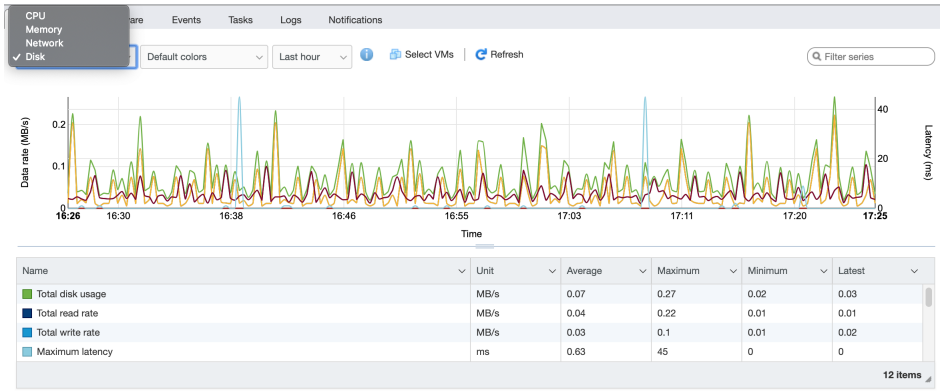
1 Management

ESXi has to be managed by the webinterface accessible via the IP address of the server.

By default SSH is disabled for security reason. You can manualy enable it.



The host performances are monitored (CPU, memory, network flows and disk) by the ESXi itself. If you thing something wrong go check the sensors values in the monitor tab.

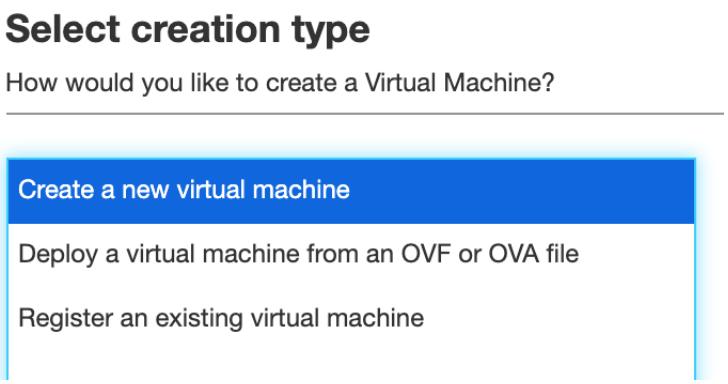


2 Virtual machines

Creation of virtual machines is simple. On the Virtual Machines tab click on "Create/Register VM".

Three posibilities are abailable. You can create totaly new virtual machine or import it from OVA files. The difference between the second and the third option is that importation from OVA files upload directly the needed files to the ESXi from another machine.

The third option is to register a virtual machine. It is used to register a virtual machine where the files are already on the ESXi datastore.



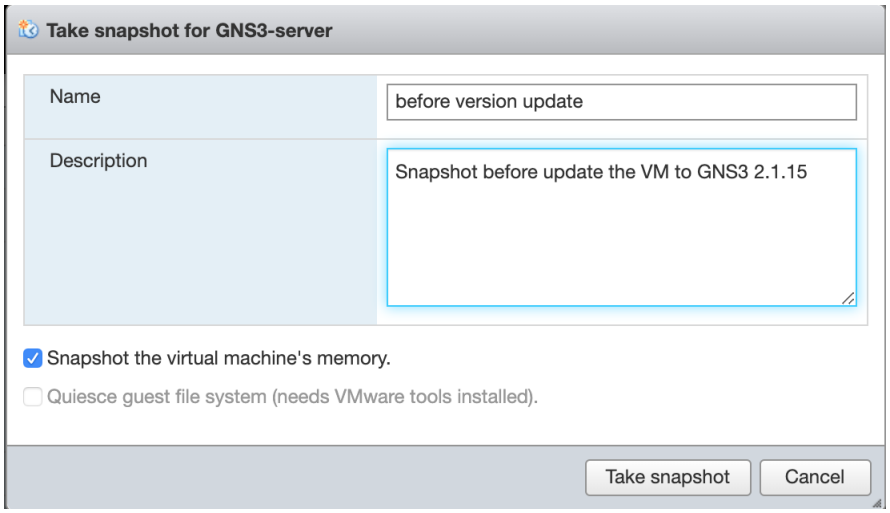
3 Snapshot

Like other virtualization solutions, ESXi has option to make a snapshot of the different virtual machines.

When you take a snapshot of virtual machine you have to specify the name of snapshot and it is possible to write a short description (like "before update components").

The snapshot can include the virtual memory state of the machine or another option called quiescing. Quiescing is another option, more "clean", to perform a backup of a virtual machine. It needs VMWare Tools installed to work.

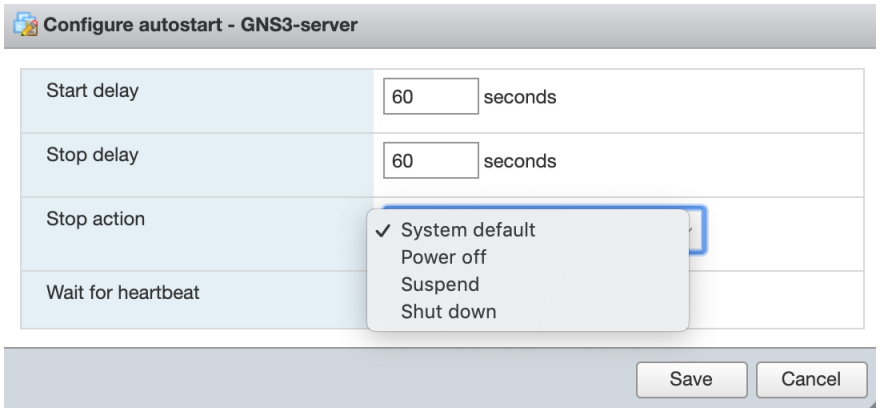
The snapshot of a GNS3 virtual machine does not need quiescing to work. This technic is used when you try to avoid rollback numbers for different services running inside a virtual machine.



Consolidation of the disk is an option after performing a snapshot. After some checks, these option erase the redundant disks to provide more space and increase performances of the virtual machine.

4 Autostart

Virtual machine can be programmed to start automaticaly when the ESXi power on after component initialization. By default, five virtual machine will boot with the ESXi. Different actions are available for autostart.



5 Add devices

5.1 USB devices

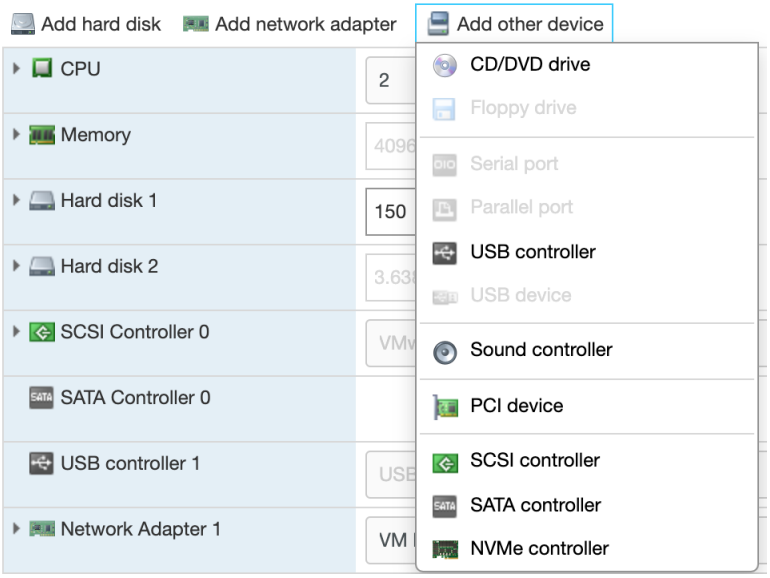
You can add more USB controller if needed in the virtual machine configuration.

Just select *Add other devices -> USB controller*.

These additional USB controller can be useful if you need to plug, for instance, USB key and the other USB controller are busy with peripheral device like keyboard or mouse.

5.2 PCI devices

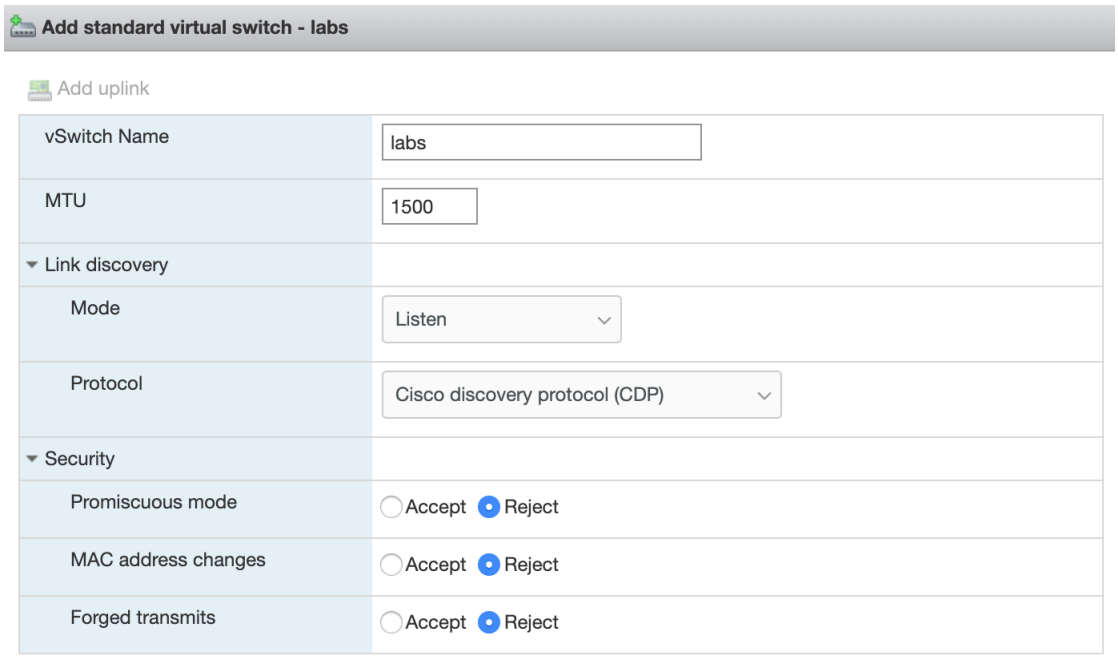
Same process as to add an USB controller. PCI devices allows you to plug for example a graphical card to perform more instructions on a virtual machine.



5.3 Networking

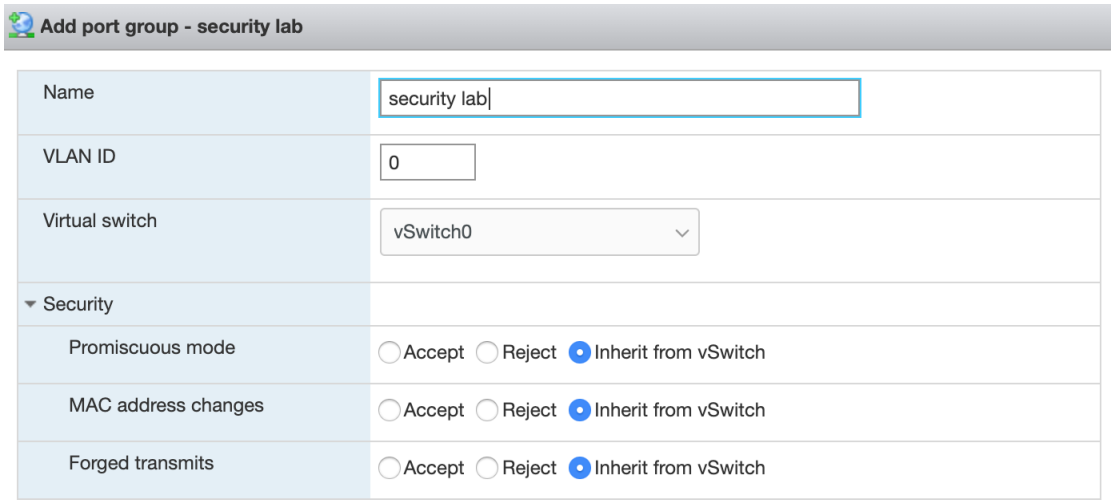
The network part of an ESXi is entirely customizable. You can divide a physical network card in several virtual switch. You can create a total of nine virtual switches.

During the creation process of a new switch you can choose some options: obviously the name of the virtual switch, the MTU (max size of packet that can be transmit in one time), link discovery and security policy.

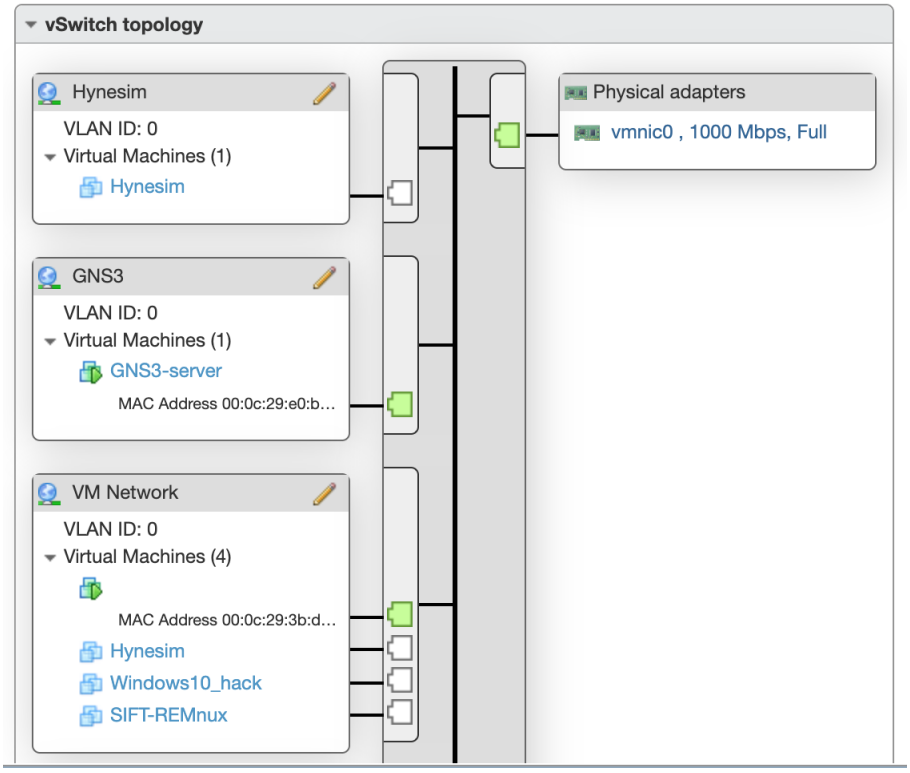


One physical adapter must be free for each virtual switch.

For each virtual switch you can create *port group*. Its creation is almost the same as for virtual switches. A port group is linked to a virtual switch. You can set a VLAN ID and apply a custom security policy or inherit from virtual switch.



A view of the topology of the virtual switch is available when you click on the corresponding virtual switch.



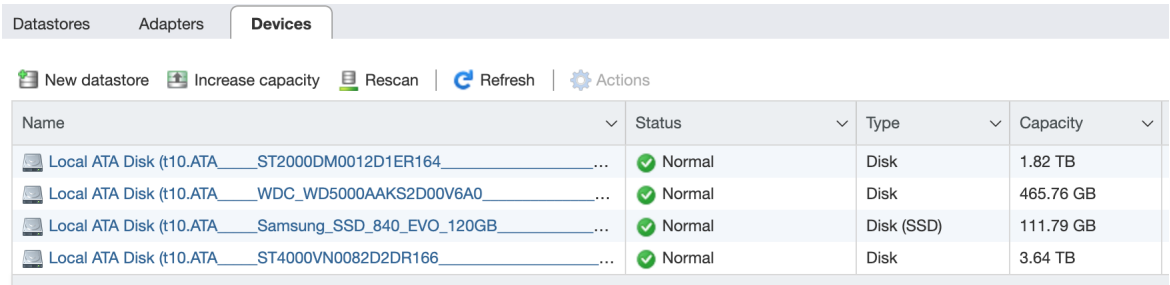
Some rules are available and can be modified in the firewall tab.

6 Useful hints - good to know

6.1 Health disks

ESXi provides probes to check the health of the different disks inside the server.

Health is available in the *storage -> devices* tab. By default all devices should have normal status. If the ESXi meets issues with the device (physical, logical, smart problem, ...) the status will change.



Do not forget to check the health if you think there is a problem running virtual machines.

6.2 Command line

If you access the ESXi using the shell you will notice that traditional linux commands do not work.

Indeed, ESXi is not Linux. The kernel is developped by VMWare. It is not recommanded to install any third party software directly on the hypervisor. However there are few useful commands base on this syntax[1]:

```
esxcli [dispatch_option] <namespace> [namespace, ...]
<cmd> [cmd_options]
```

- **localcli**: contains a set of troubleshooting commands.
- **esxcli network ip connection list**: list active TCP/IP connection.
- **esxcli vm process list**: list processes (possibility to use options like kill to kill a process).
- **esxcli software vib list**: list all VIB packages installed on the host.
- **esxcli network firewall get**: check if the firewall is active (possibility to interact with the firewall via the command line).

The command line is very powerful but needs to catch the VMWare ESXi syntax. Most of main options are available via the webinterface.

6.3 Update

By experience, avoid to update your host to a newer version except if you are totaly sure to avoid all possible compatibility issues. Some features are depreciated on different version and it can provide virtual machine issue configuration.

However there is absolutely no problem to update the webinterface of the host. When an update is available you will see a notification after your connection to the webinterface. Proceed only by clicking install the new update and the ESXi will download the VIB package related to the interface update.

You you need to update a functionnality of one of the virtual machines you can snapshot it to avoid bad surprise.

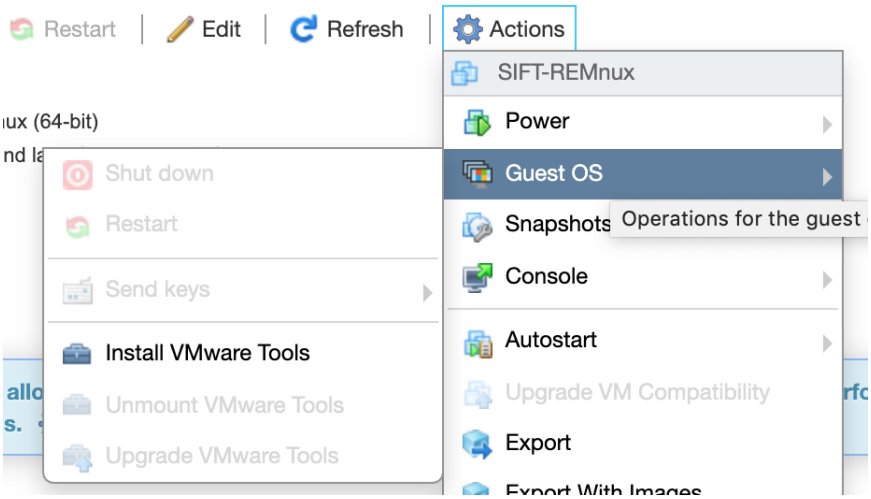
6.4 VMWare tools

VMWare ESXi provides VMWare Tools. It is about a very powerful tools which can send information used for monitoring to the main host. Do not forget to install these tools when you create a new virtual machine.

[1] From VMWare ESXi documentation

These tools are available via a live CD which is plugged to the virtual machine.

Install the VMWare Tools increase the virtual machine performances and allows basic stuffs like copy paste inside the virtual machine.



6.5 Power off

Normally, if the power off button of the server that runs the ESXi is pressed the ESXi does not shutdown immediately but try to send shutdown command to virtual machine that already running.

To avoid virtual machine corruption or any other problem it is advice that the virtual machine should be shutdown manually using the webinterface. You can select all the machine running and shutdown all of them in the same action.

6.6 DNS

Be careful, DNS server plays a crucial role for the ESXi and all of its services. If one day you have the need to change the default DNS server and you have problems, think to check your DNS configuration.

6.7 Log in issue

Sometimes after introducing your credentials it is possible that the webpage panel administrator does not load. If it is the case, just refresh your web browser.

Annexe B

Prise en main de GNS3

Cette section de l'annexe est dédiée à la prise en main de la solution GNS3. La documentation est écrite en anglais.

Getting started GNS3

Pierre Michaux

GNS3 is a virtualization solution used to emulate networking scenarios. Many options are available and make it an powerful tool.

This guide provides tips and configuration concerning this virtualization solution.

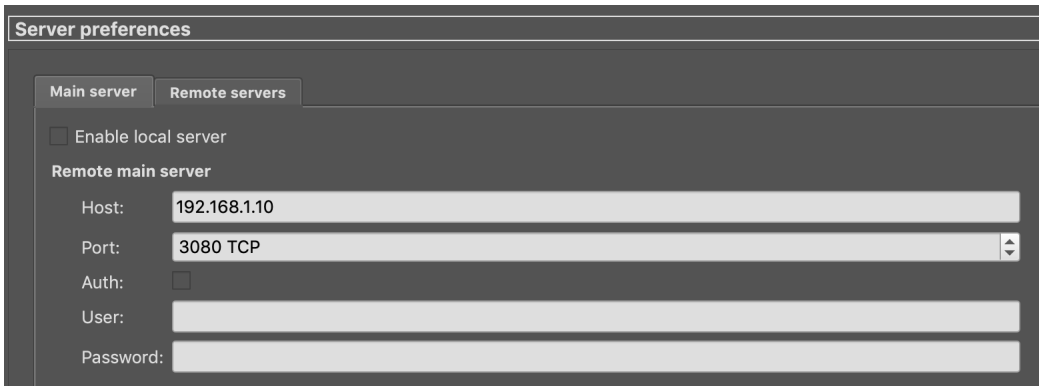
Remark: if you want to use GNS3 with remote server to use its resources the client version has to be the same as the version of the server.
You keep your parameters after updating your software.

1 Configuration

Many aspect can be configure for GNS3. Here is a not-exhaustive list of settings configuration.

1.1 Standalone or remote server

GNS3 works locally or using the resources of a remote server. The type of configuration can be done inside the preferences of GNS3.



Just indicate the IP address of the server that runs GNS3 virtual machine.

By default you do not need username or password to connect to the server.

If you do not want to work with the resources of the remote server just click *Enable local server*. Equipments will use the resources of your local computer.

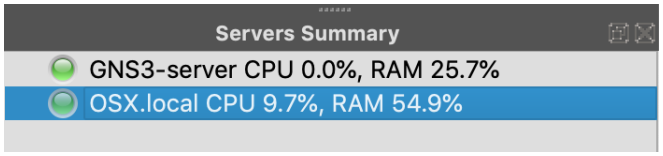
In the remote servers tab you can add several other servers. Indicate another server is used when you want to split used resources in your scenario. For example, you can start a device using the resources of the first server and start another instance using the resources of the second server.

A second remote server works only if you mention the main server.

To add a second server, just click on Add button and specify the IP address of the server.

On the server summary you can see the resources you are using. A huge benefit for GNS3 is the capacity to work in hybrid mode.

Hybrid mode allows the user to use the resources of his personal equipment to run devices but also use the resources of the server. Your scenario can use your physical machine and if it is not enough powerful you can ask the resources of the server for other devices.



1.2 Virtual Network Computing

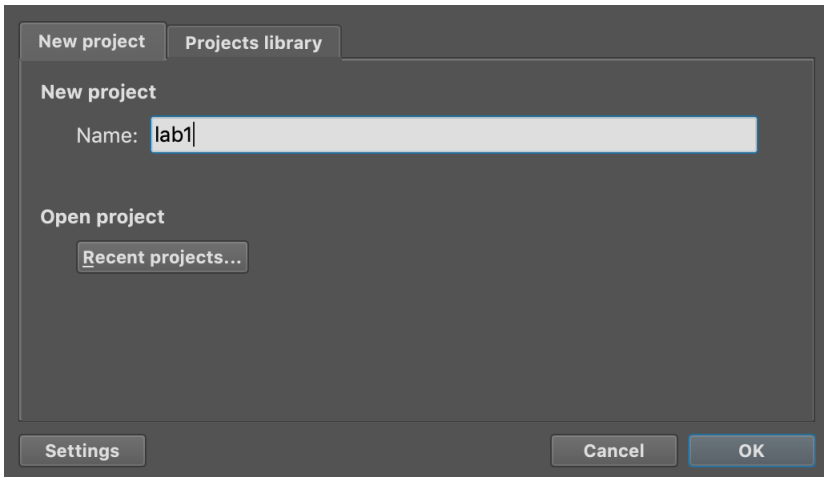
Depending of the user operating system you must change the parameters of the VNC used by GNS3.

You may have problem to access to the interface of an imported custom virtual machine.

The settings for the VNC are in General tab - VNC subtab where you have to indicate the path of your VNC application. If you have no favourite VNC application you should use Chicken of the VNC. It works very well.

2 New project

To create a new project just click on the New Blank Project button. Settings can be modified according to the new project.



3 Default devices

GNS3 provides default devices like virtual PC, switch, cloud connection, frame relay and more.

These devices allow a very basic configuration but there are very useful in many situation.

3.1 Virtual PC

Virtual PC are used to perform test and verify the connectivity between network.

You can configure the virtual PC by using the command line or by editing the configuration file.

For example, to modify the IP address by the command line:

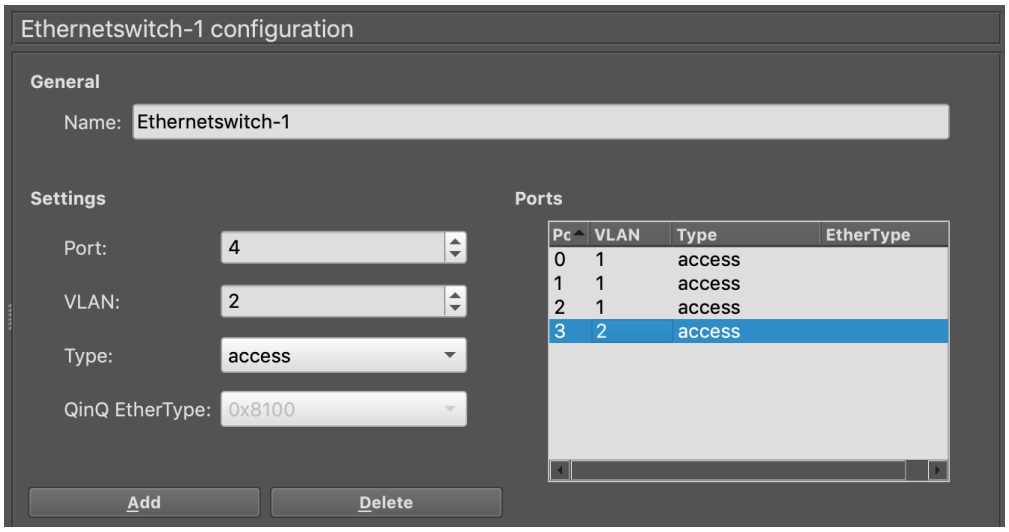
```
ip [ip][CIDR] [gateway]
ip 10.0.0.2/24 10.0.0.1
```

If you prefer the graphical method, right click on the VPC and select modify configuration file. You need to uncomment some line to force static IP configuration.

```
# This the configuration for PC-1
#
# Uncomment the following line to enable DHCP
# dhcp
# or the line below to manually setup an IP address and subnet mask
# ip 192.168.1.1 255.0.0.0
#

set pcname PC-1
```

3.2 Switch

GNS3 provides switch with basic functions. You can add ports on the switch. You can create VLAN and assign ports to the VLAN. It is possible to specify the type of the port: access, dot1q or qinq.



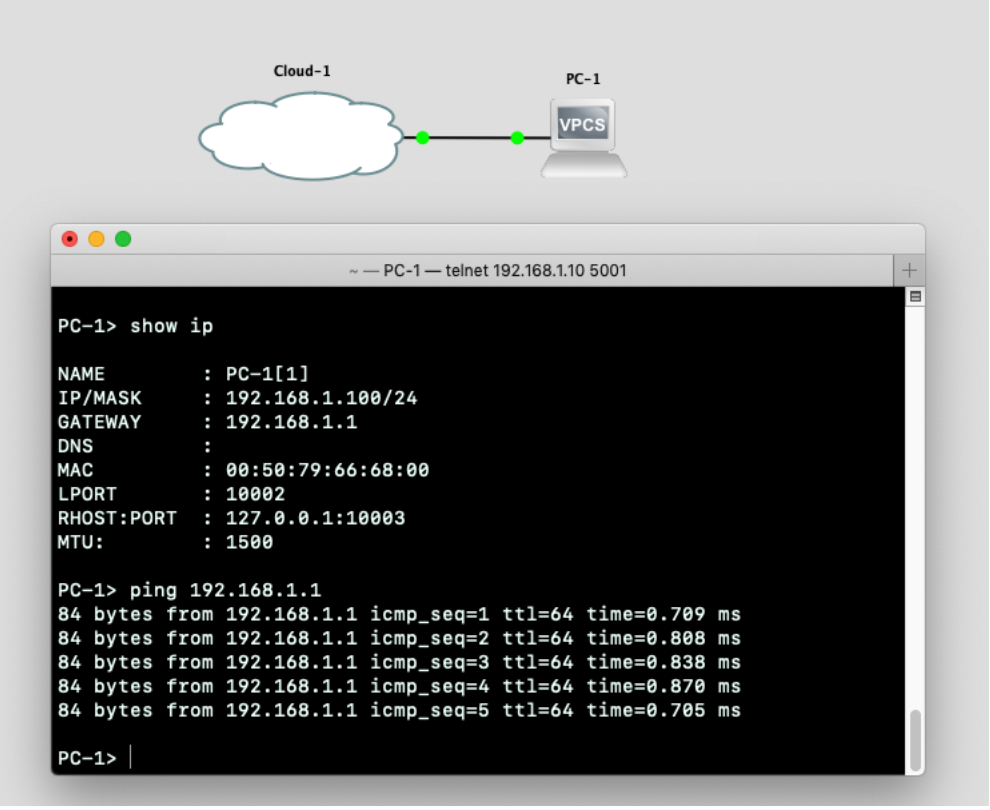
It is not possible to use the command line to configure basic switch. The configuration is only possible through the graphical interface.

Many configuration options are missing.

3.3 Cloud

The cloud node is a crucial equipment. It makes GNS3 to create hybrid scenario. By deploying the cloud node you create a kind of gateway to interact with physical equipments which are present in the LAN.

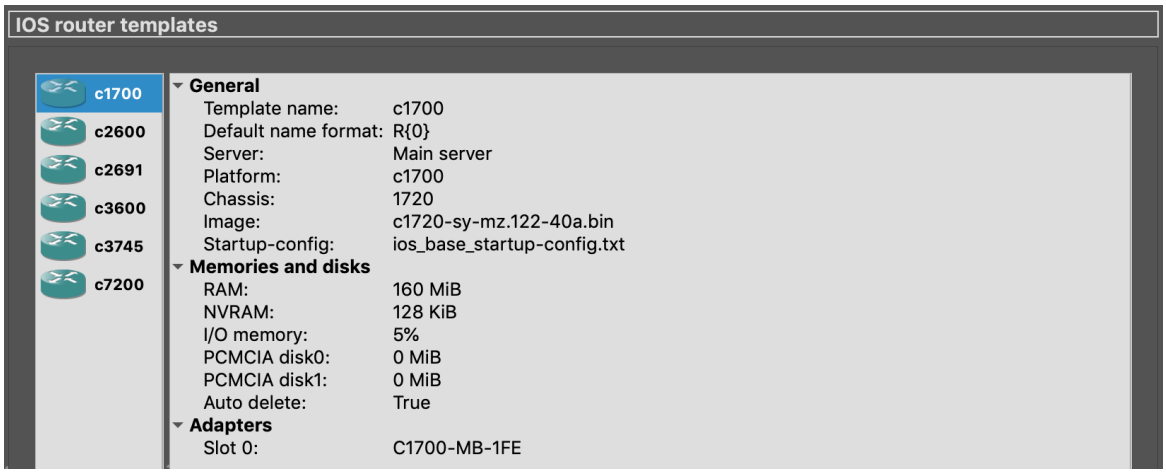
A project with a cloud node cannot be exported. You have to remove it before exportation. Basically you do not have to do anything to interconnect your virtual network with the physical network. In the example below the virtual PC named PC1 with IP address 192.168.1.100/24 is able to ping the physical default gateway at IP address 192.168.1.1/24



4 Router management

Dynamips is an emulator program used to emulate the IOS of Cisco routers.

To add, delete or configure IOS routers go to the preferences of GNS3 and select IOS routers under the Dynamips section. You will see the list of all routers (and their specifications) available on the server.

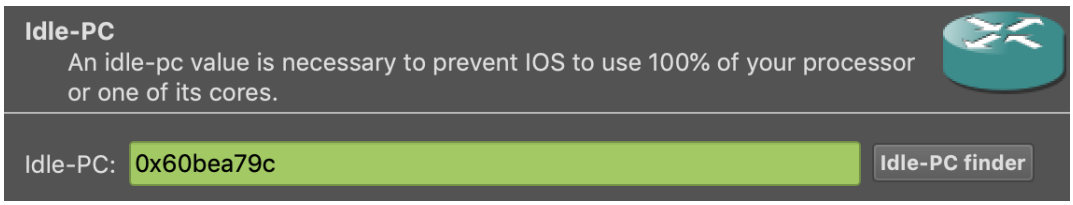


4.1 Create new router

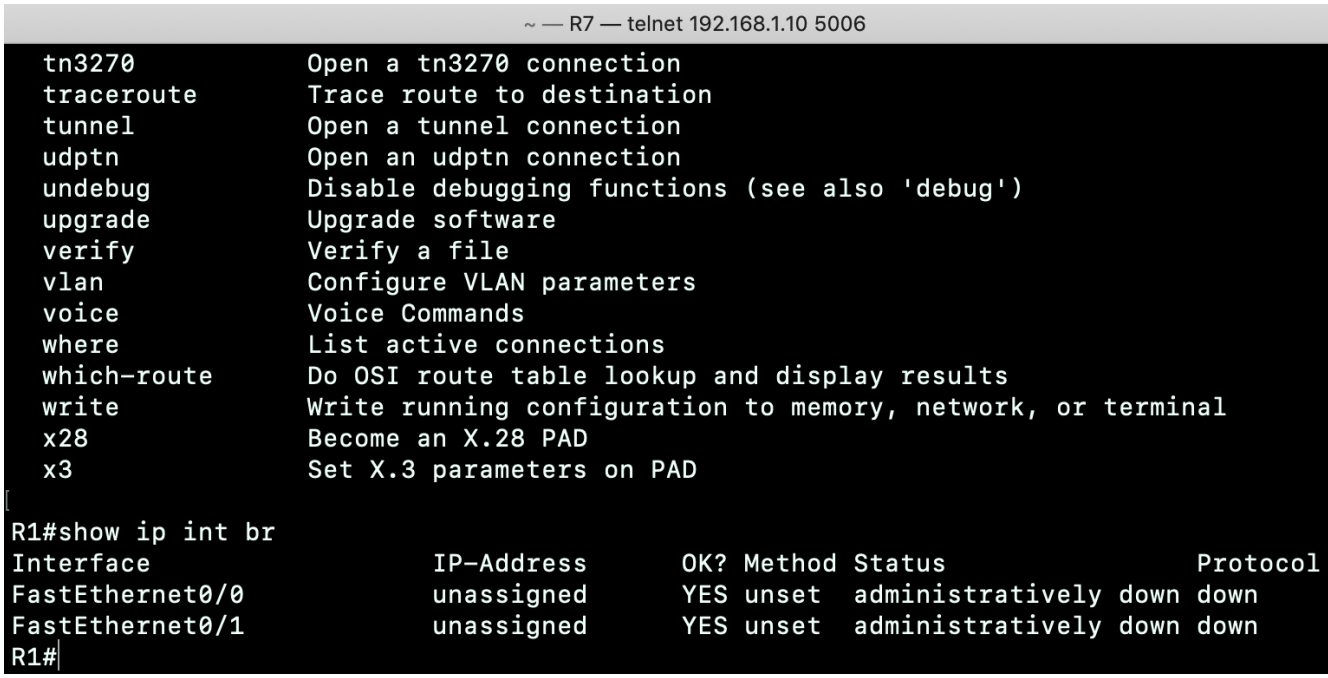
To add a new router click on the add button. Select your IOS image. GNS3 ask you to decompress the IOS image. Click no. GNS3 supports compressed and uncompressed files. The image is uploaded on the server.

Check the specifications of the new router in the documentation. You have to specify the maximum RAM, default network adaptaters, wic modules and the idle pc value. This last value is essential for the router to work. It calculates the time when the IOS image is idle (not performing any task) and puts the router into an idle state (sleep mode).[1]

This value can be found automatically by clicking on the Idle-PC finder button.



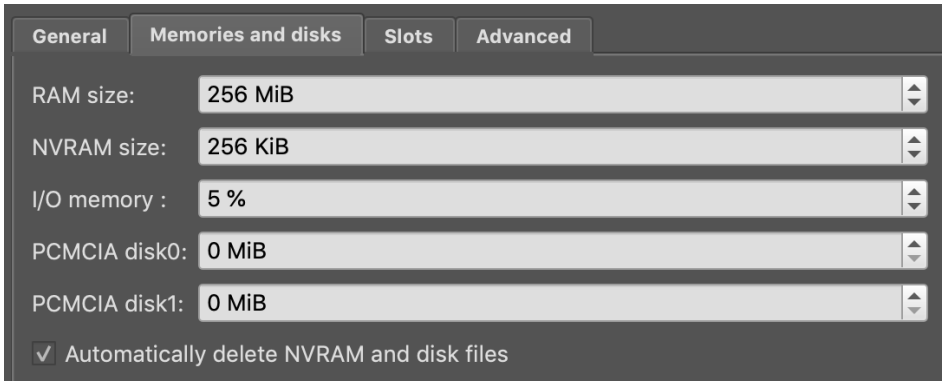
Once the router template has been created you can drag and drop the router inside your project. Start the router and double click on it to open the configured terminal to access to your router.



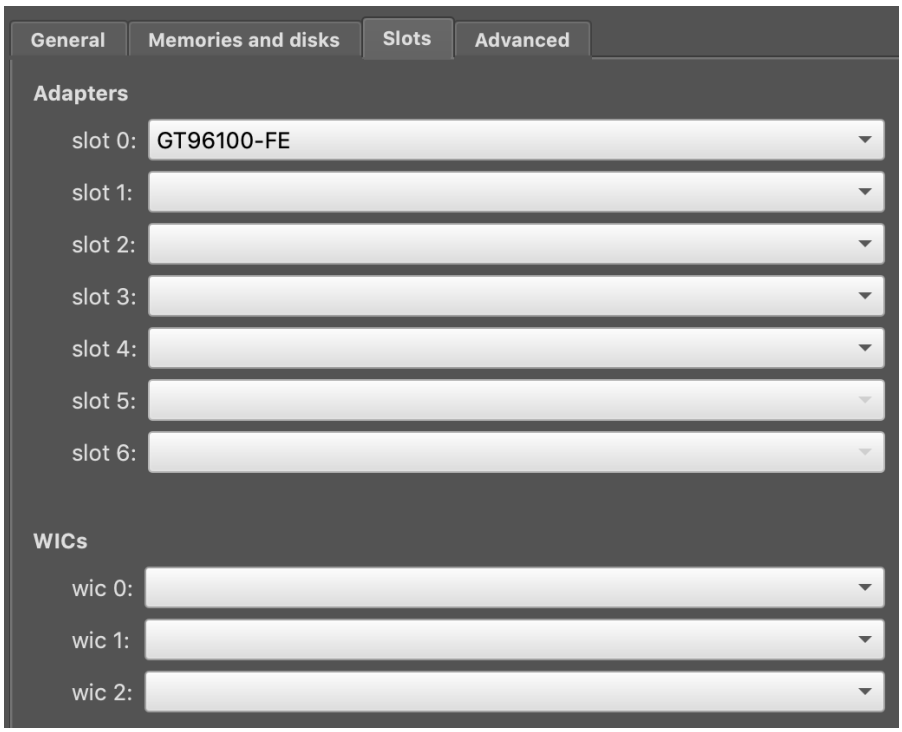
[1] http://www.smartpctricks.com/2014/05/calculate-idle-pc-value-in-gns3.html

4.2 Modify existing router

To modify an existing router click on it and select modify. You can modify all atributes you defined during the creation of the router.



It can be very useful if you need to add more slots or interfaces.



WIC cards are used for the router to transmit data over WAN.

4.3 Delete existing router

To delete a router just select it and press the delete button. Be careful, there is no confirmation.

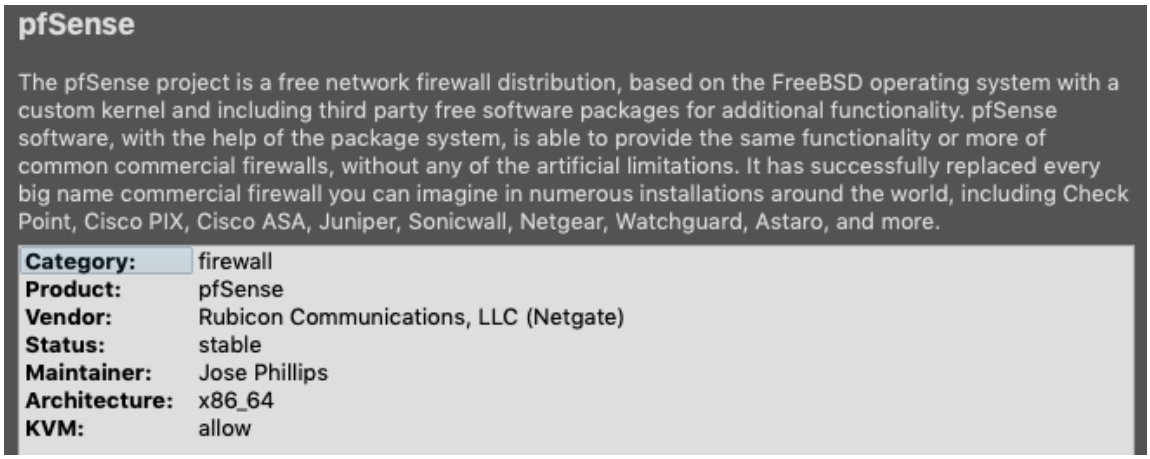
5 GNS3 market place

The strenght of GNS3 is the wealth of its marketplace. The marketplace is a location where different companies can upload their own instance compatible with GNS3.

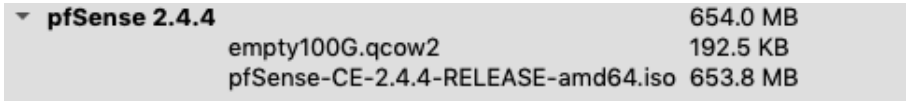
Check `https://www.gns3.com/marketplace`

For instance, firewalls are essential components in network security flows. Here comes the utility of the marketplace. You can find a Pfsense appliance developped by Electric Sheep Fencing LLC on it.

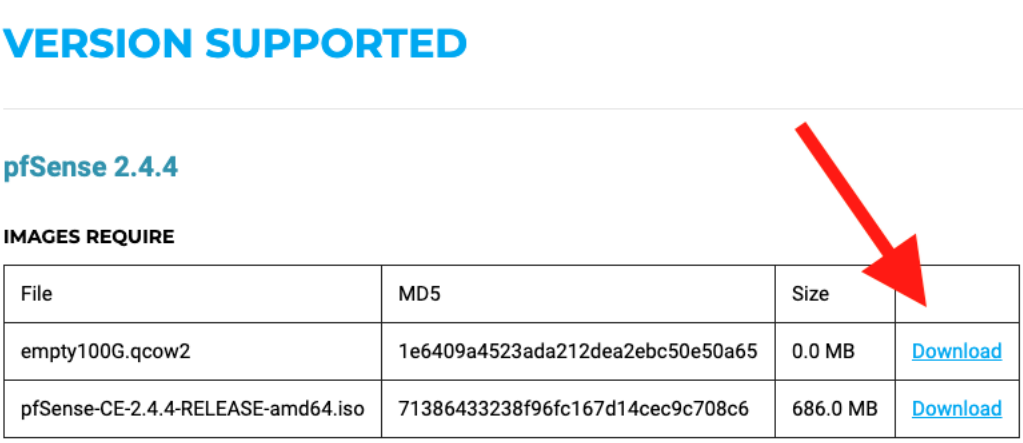
Just download the appliance, you got a .gns3a (GNS3 appliance) file and need to import it on the server. Automatically GNS3 will check if the GNS3 server requirements are ok or not and continue the installation.



Obviously you might have the required image of the Pfsense. GNS3 will show you the version related to the appliance available on the market place.



If you do not have the image do not worry! GNS3 provides on its website the link to the needed files.



It is a very simple process.

6 Import custom images

Import custom images can become quickly a very very tricky task. On GNS3 you have four posibilities to import your custom images. You have the choice between VirtualBox, VMware, Docker or Qemu. For compatibility and homogeneous configuration of the equipment you have to use Qemu.[2]

And that is where the difficulty could happen. First of all you have to create your custom virtual machine on your workstation before sending it to the server.

If you are a Linux-fan et you already use Qemu to create and run your virtual machines there will be no problem during the migration to the server.

However if you use one of the three solution mentioned above you might have some troubles.
The first thing is the compatibility with the virtual hard drives. Qemu uses the **qcow** file format for virtual machines. It is not the case of the other solutions. To avoid compatibily problems on the server the trick is to convert this hard drive file format to the file format used by Qemu.

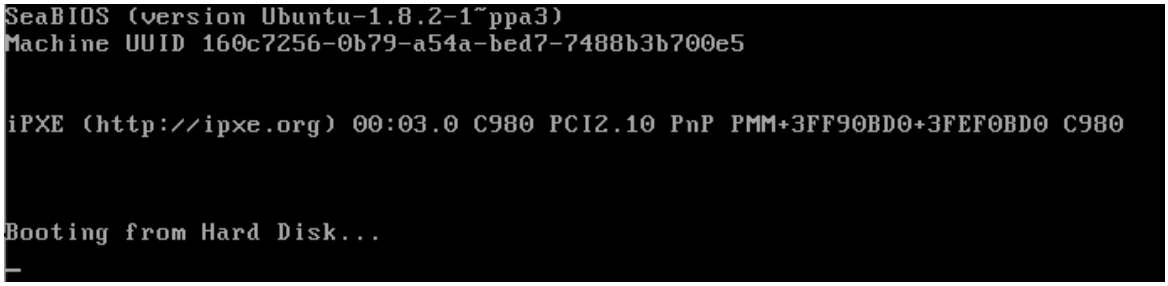
On Linux you have to install the **qemu-utils** package. It is available using *apt* command. Good new for MacOS users! Qemu-utils is also available using *brew*. Only the name of the package is different on MacOS. You only need to install **qemu**.

[2]Find the reason to this choice in my master thesis.

Here is some interesting commands you might need[3]:

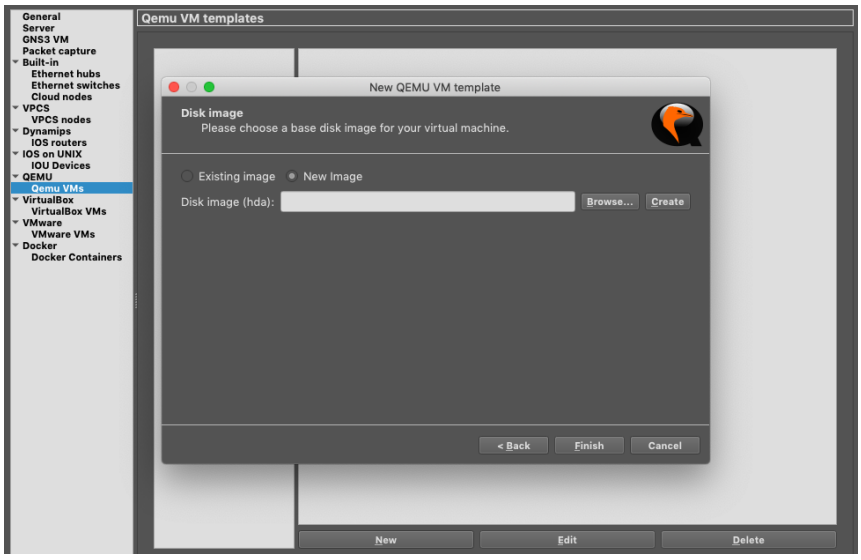
```
To convert raw image file to qcow2 format:
$ qemu-img convert -f raw -O qcow2 image.img image.qcow2

To convert vmdk image file to qcow2 format:
$ qemu-img convert -f vmdk -O qcow2 image.vmdk image.qcow2

Global syntax:
$ qemu-img convert -f [input_file_format] -O
[output_desired_file_format] [input_file] [output_file]
```

A second problem that might occur concern the boot firmware. You will probably have the choice between Legacy Bios or UEFI. The difference between these two systems will not be describe here. It is important to notice that UEFI will not work with Qemu natively causing this kind of screen when the virtual machine starts up.



Obviously, the importation of virtual machines on the server has to be made from the export of the previous virtual machines created locally on your workstation.

The import is made inside the GNS3 preferences. In the Qemu tab click on new and follow the instructions. Select new image to import your custom image to the server. Depending of the network connection, uploading can be slow or fast.



[3]From Openstack, Converting between image formats

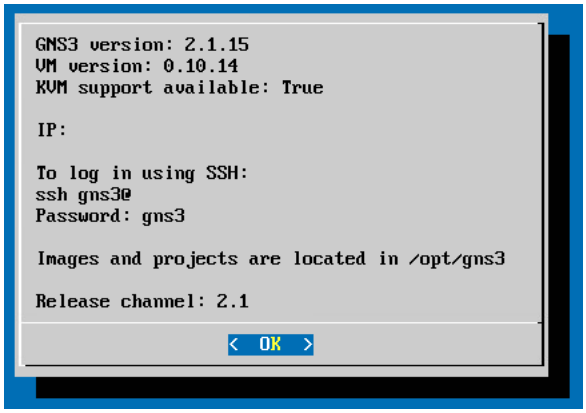
7 Export scenario

GNS3 is truly a powerful tool. You have the possibility to choose to export all iamges containing in a scenario.

8 Command line

You can access to the shell of the GNS3 virtual machine using SSH. Default credentials are

```
username: gns3
password: gns3
```



If you want to sort or delete old project or uploaded images (IOS/QEMU): projects and images are located in the repository /opt/gns3

If you remove an appliance from the graphical user interface of GNS3 the images will not be removed from the /opt/gns3/images repository.

Good to know: pfSense installation takes about 2 or 3 minutes.

```
Pré-requis :
Avoir curl sur la machine
  -> apt-get install curl
Désactiver le gestionnaire réseau de la machine
  -> systemctl stop NetworkManager
  -> systemctl disable NetworkManager
Suppression des packets qui interfèrent
  -> apt-get purge avahi-daemon libnss-mdns

Installation du système :
Ajouter les dépôts dans sources.list
  -> deb [arch=amd64] http://repository.hynesim.org/
     debian jessie backports qemu 2.5 2.4
  -> deb http://archive.debian.org/debian
     jessie-backports main contrib non-free
Ajouter  les clés de signature du dépôt
  -> curl -o - https://repository.hynesim.org/debian/hynesim.asc
     | apt-key add -
Mise à jour des paquets
  -> apt-get update
Mise à jour du noyau
  -> apt-get install -t jessie-backports linux-image-amd64
  -> reboot

Installation d'Hynesim :
Installation du paquet
  -> apt-get -y install -t jessie-backports hynesim
Création de l'arborescence
  -> for i in import export shared resources catalog/topologies
     catalog/entities catalog/guestfoundry;
     do mkdir -p /data/hynesim/$i; done
Configuration des daemons
  -> vi /etc/default/hynesim-glacier
  -> HYNESIM_GLACIER_ENABLE=true
  -> systemctl start hynesim-glacier
  -> vi /etc/default/hynesim-master
  -> HYNESIM_MASTER_ENABLE=true
  -> systemctl start hynesim-master
  -> vi /etc/default/hynesim-node
  -> HYNESIM_NODE_ENABLE=true
Activer KVM/QEMU
  -> Décommenter des lignes in /etc/libvirt/qemu.conf
     user = "root"
     group = "root"
  -> Dé-commenter le bloc cgroup_device_acl, puis rajouter
     dans la liste la valeur « /dev/net/tun »
     cgroup_device_acl = [
     "/dev/null", "/dev/full", "/dev/zero",
```

```
      "/dev/random", "/dev/urandom",
      "/dev/ptmx", "/dev/kvm", "/dev/kqemu",
      "/dev/rtc","/dev/hpet", "/dev/vfio/vfio",
      "/dev/net/tun"
      ]
Rédémarrer le service libvirtd
  -> systemctl stop libvirtd
  -> systemctl start libvirtd
Configuration de LXC (Linux Container)
  -> Éditer le fichier /etc/default/grub
  -> Remplacer la ligne GRUB_CMDLINE_LINUX=        par
     GRUB_CMDLINE_LINUX="cgroup_enable=memory swapaccount=1"
Mettre à jour Grub
  -> update-grub
Installer les paquets LXC nécessaires aux déports d'écrans
  -> apt-get -y install screen libspice-server1
  -> reboot
Activation des services dans le fichier de configuration
  -> vi /etc/hynesim/hynesim-node.ini
     ;VirtualBox/dummy=1
           KvmQemu/dummy = 1
           LXC/dummy = 1
  -> systemctl stop hynesim-node
  -> /etc/init.d/hynesim-node cleanup
  -> systemctl start hynesim-node
Support des cartes hybrides
  -> vi /etc/netwrok/interfaces
     auto eth1
     iface eth1 inet manual
  -> vi /etc/hynesim/hynesim-node.ini
     HybridNetcard/dummy = 1
     HybridNetcard/cards/1/device = eth1
     HybridNetcard/cards/1/plug = "Plug 1"
     HybridNetcard/cards/size = 1
  -> systemctl stop hynesim-node
  -> systemctl start hynesim-node
Activation de l'accès à distance
  -> vi /etc/hynesim/hynesim-glacier.conf
     Ice.Default.Host = IP_MACHINE
  -> vi /etc/hynesim/hynesim-master.ini
     Ice.Default.Host = IP_MACHINE
  -> vi /etc/hynesim/hynesim-node.ini
     Ice.Default.Host = IP_MACHINE
 Support des consoles des routeurs
  -> apt-get install dynamips-utils
```

La procédure de mise en place est disponible sur le site d'Hynesim. La liste des commandes ci-dessus et de leur descriptif sont tirés de leur guide d'installation. Certaines

commandes ont été ajoutées pour permettre d'utiliser Hynesim.

Annexe C

Vue globale

C.1 Hynesim

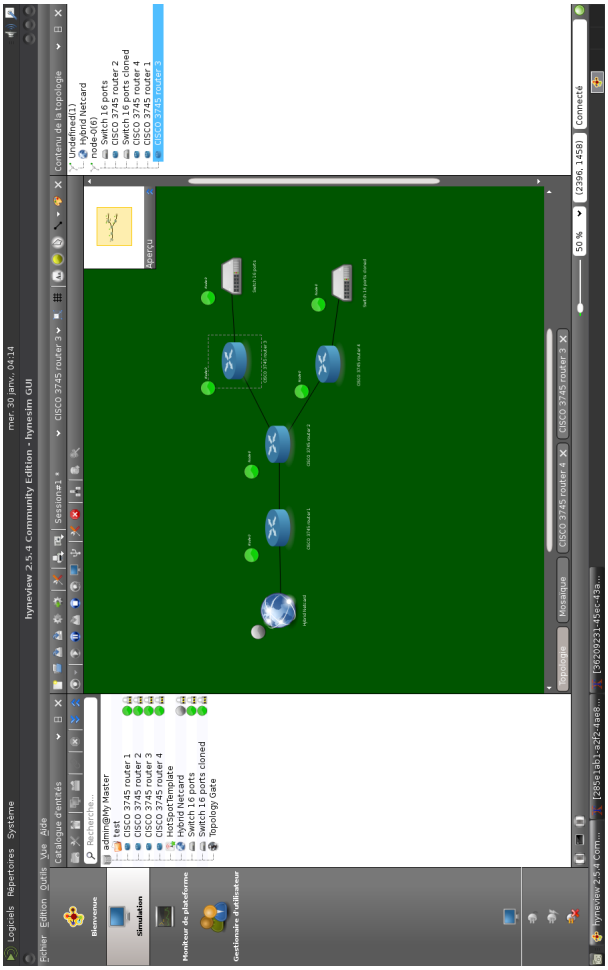


FIGURE C.1 – Vue globale Hynesim.

C.2 GNS3

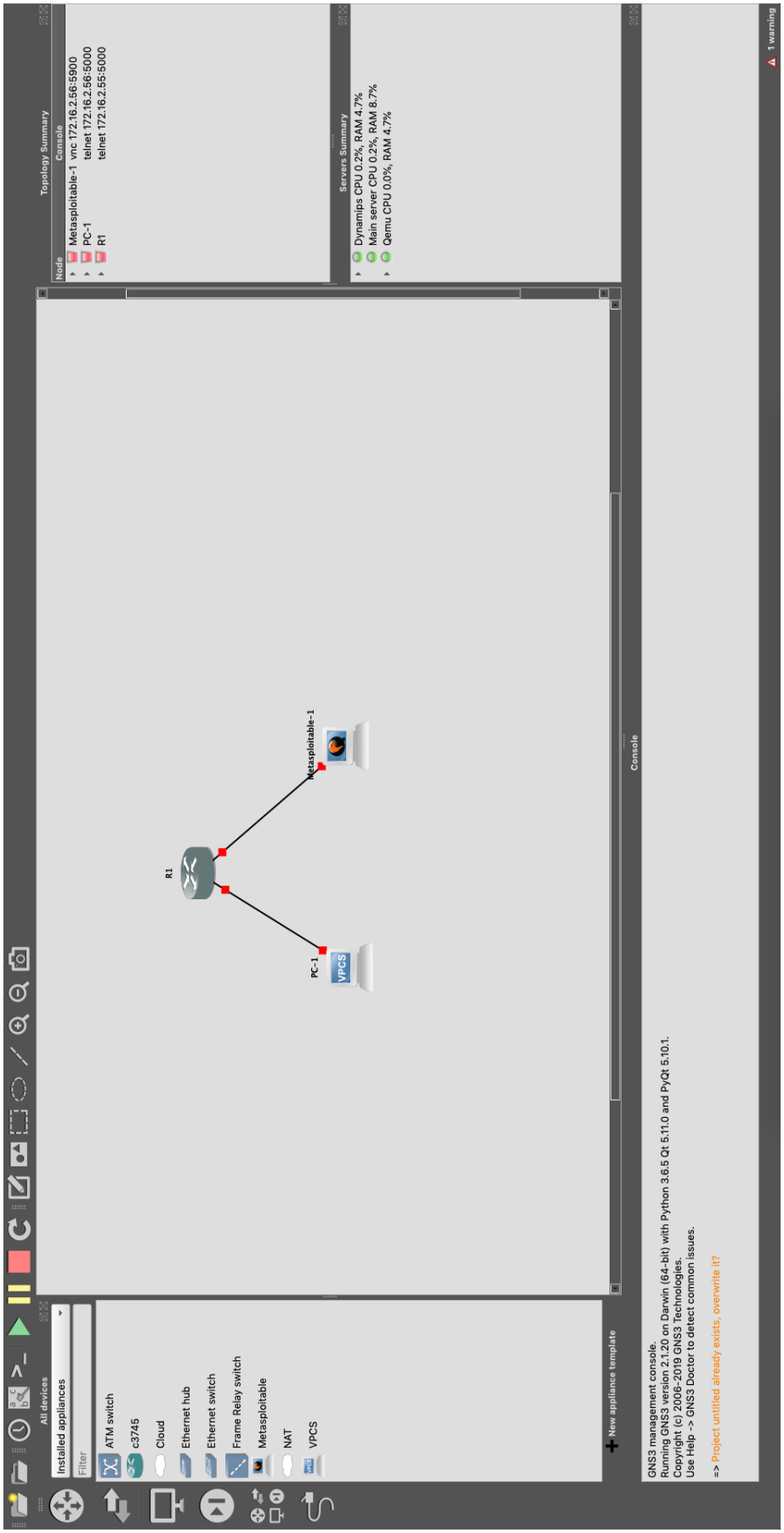


FIGURE C.2 – Vue globale GNS3.

Annexe D

Graphiques

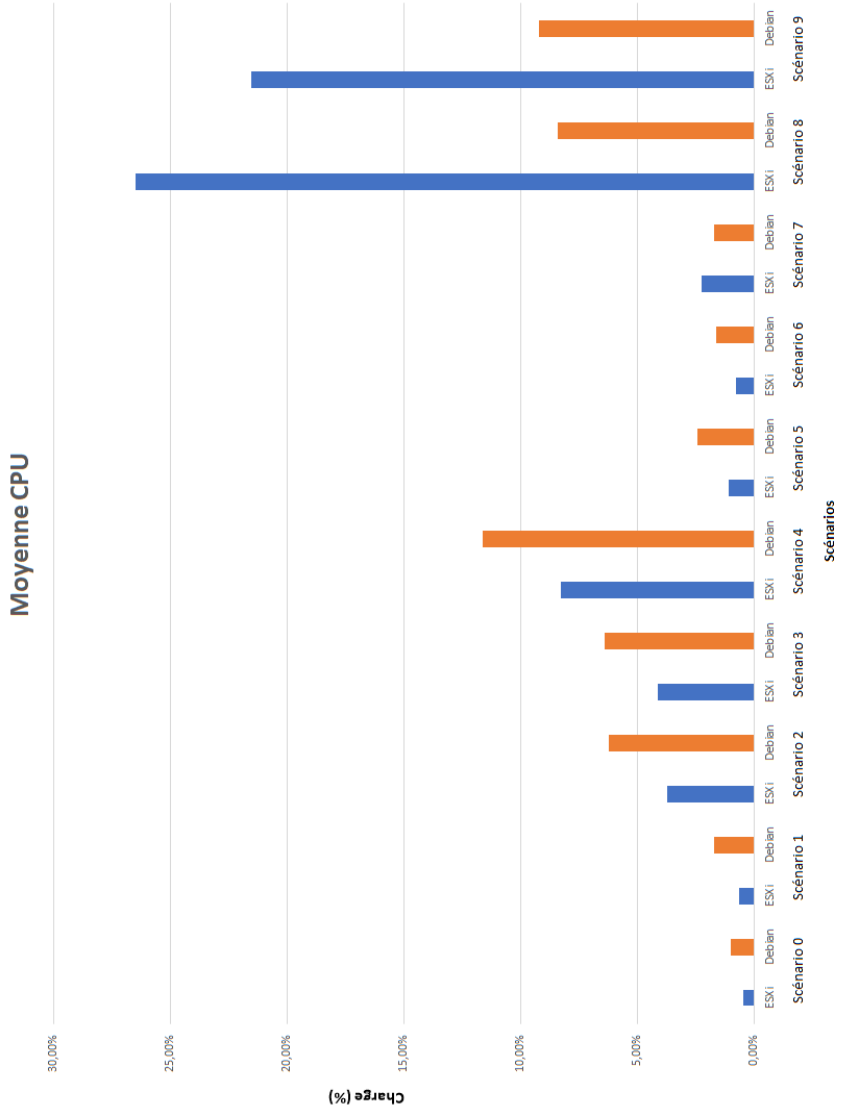


FIGURE D.1 – Moyenne de la charge des hyperviseurs.

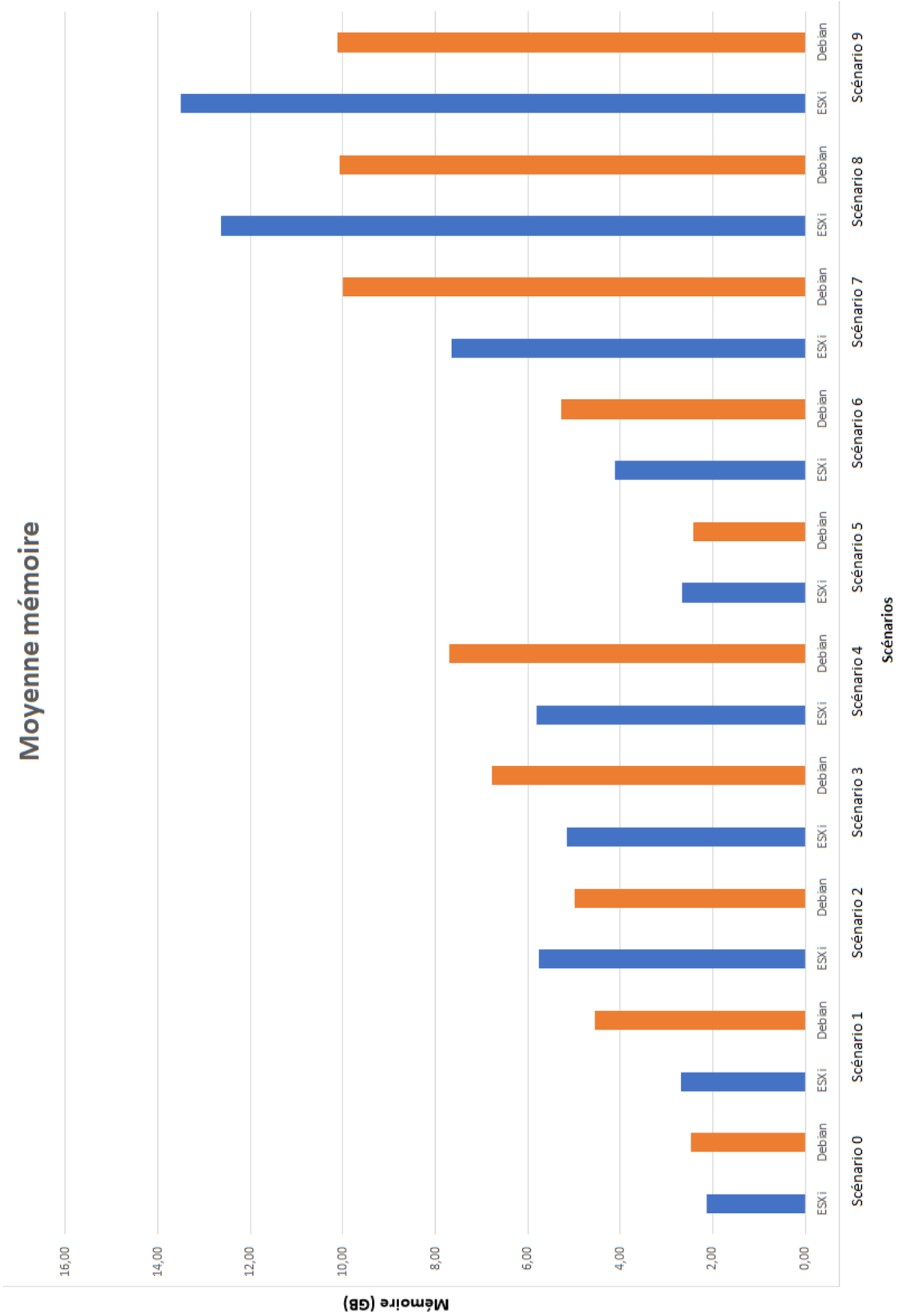


FIGURE D.2 – Moyenne de la mémoire des hyperviseurs.

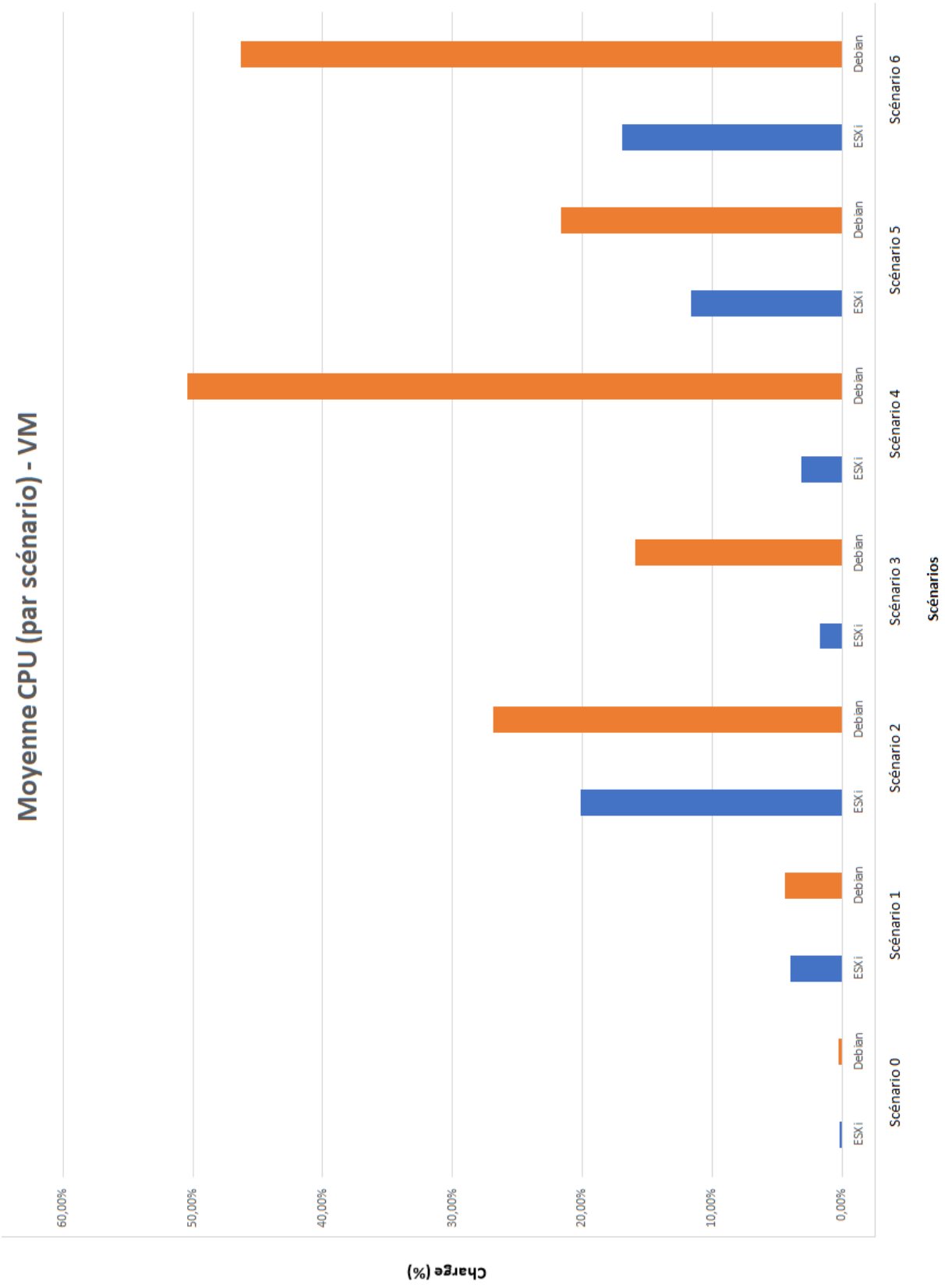


FIGURE D.3 – Moyenne de la charge des machines virtuelles.

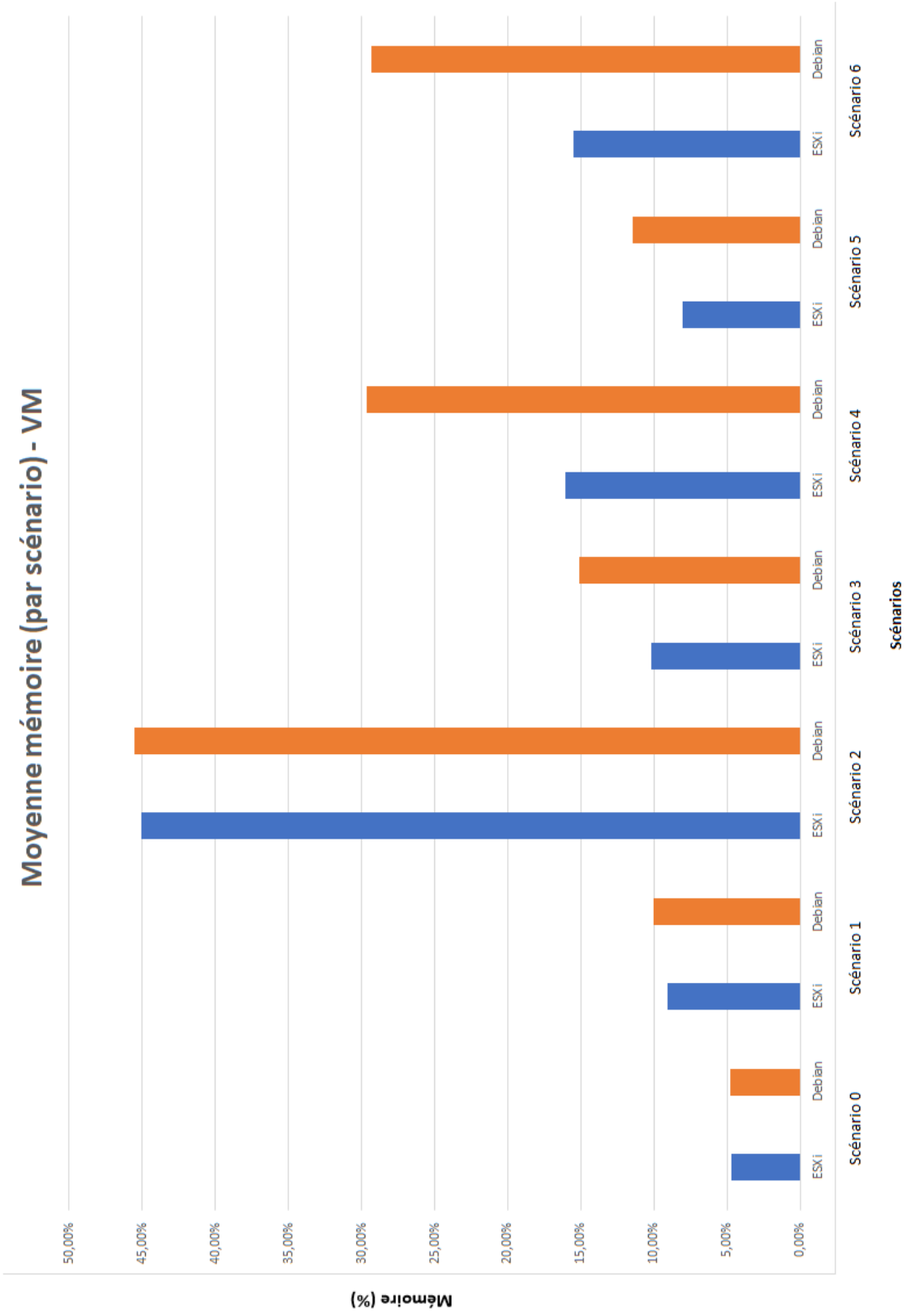


FIGURE D.4 – Moyenne de la mémoire des machines virtuelles.

Annexe E

Autres

E.1 *Windows error boot*

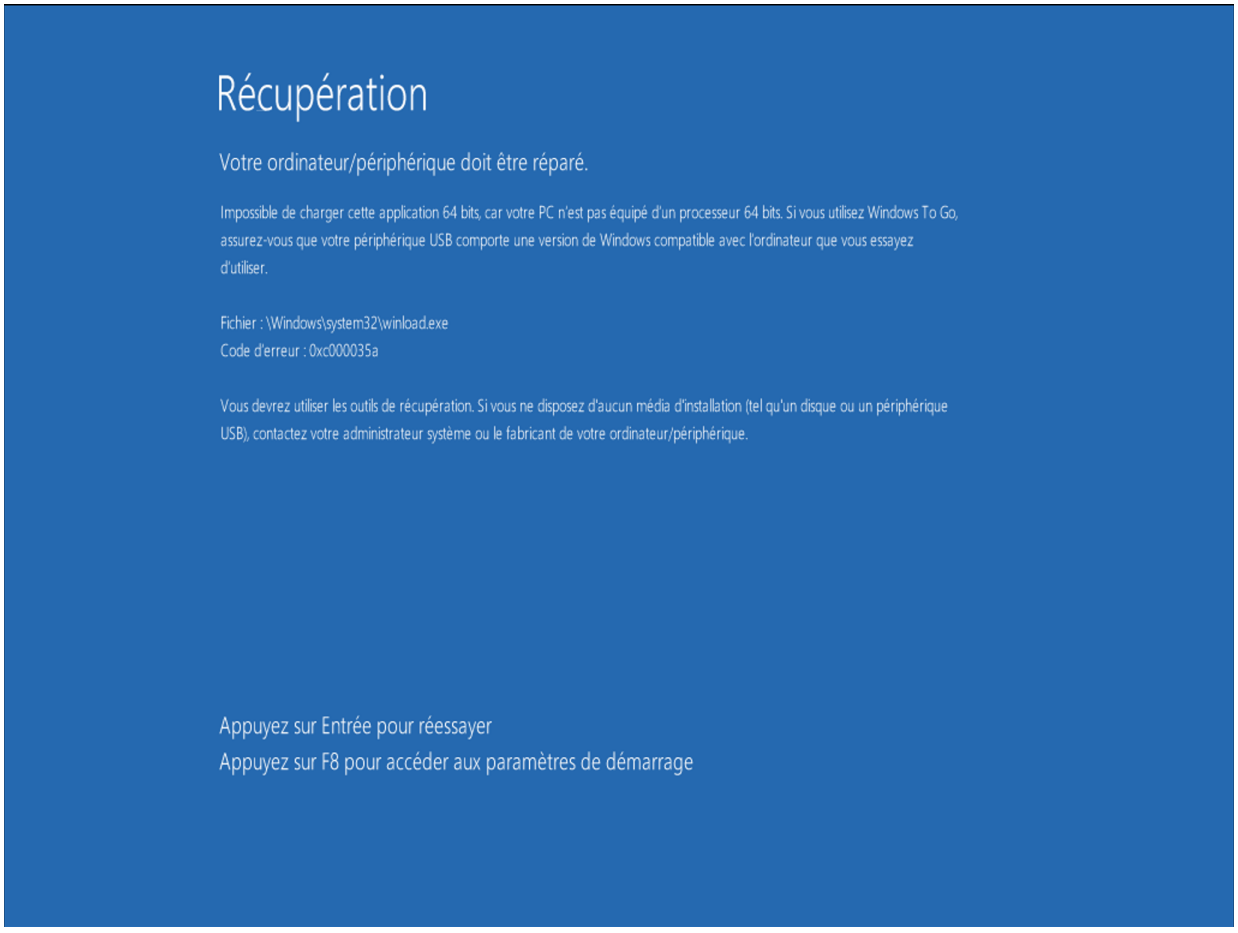


FIGURE E.1 – Erreur Windows après l'importation sur une solution de virtualisation différente.

E.2 VirtualBox CLI

Cette section de l'annexe est dédiée aux commandes de base qu'il faut connaître si on désire gérer des machines virtuelles en utilisant exclusivement la ligne de commande de VirtualBox.

```
Par défaut, les fichiers des machine virtuelles
sont stockés dans $USER/VirtualBox/VMs/

Obtenir la liste des machine virtuelles
enregistrées
        -> VBoxManage list vms

Démarrer une machine virtuelle
        -> VBoxManage startvm    vm      type   headless

Obtenir des informations sur une
machine virtuelle
        -> VBoxManage guestproperty enumerate <vmname>
        -> VBoxManage showvminfo    vm

Arrêter une machine virtuelle
        -> VBoxManage controlvm    vm    poweroff

Supprimer une machine virtuelle
        -> VBoxManage unregistervm    vm

Démarrer la machine virtuelle et la
rendre accessible via VRDE
        -> VBoxManage --startvm "vm" --vrde on

Voir les adaptateurs réseaux
        -> VBoxManage list bridgedifs

Changer la configuration des cartes réseaux
        -> VBoxManage modifyvm "vm" --nic1 bridged
  --bridgeadapter1 ens1

Activer le VRDE sur une machine
        -> VBoxManage modifyvm    vm    --vrde on
```

E.3 Nagios screenshots

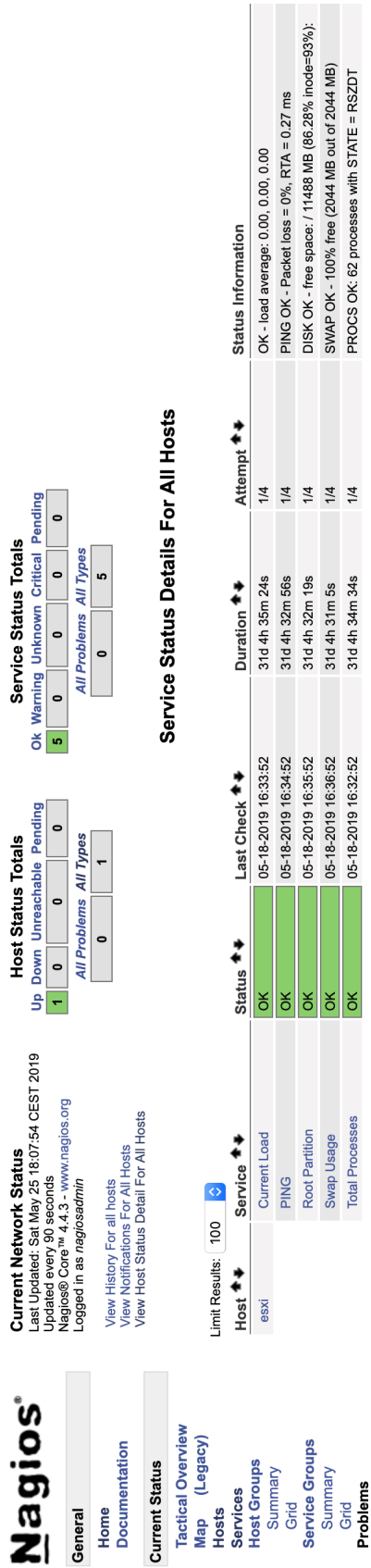


FIGURE E.2 – Vue globale Nagios.

Service State Information

Current Status:	**OK** (for 31d 4h 36m 52s)
Status Information:	PROCS OK: 62 processes with STATE = RSZDT
Performance Data:	procs=62;250;400;0;
Current Attempt:	1/4 (HARD state)
Last Check Time:	05-18-2019 16:32:52
Check Type:	ACTIVE
Check Latency / Duration:	0.001 / 0.019 seconds
Next Scheduled Check:	05-25-2019 18:12:46
Last State Change:	04-24-2019 13:33:20
Last Notification:	N/A (notification 0)
Is This Service Flapping?	**NO** (0.00% state change)
In Scheduled Downtime?	**NO**
Last Update:	05-25-2019 18:10:05 (0d 0h 0m 7s ago)

Active Checks:	**ENABLED**
Passive Checks:	**ENABLED**
Obsessing:	**ENABLED**
Notifications:	**ENABLED**
Event Handler:	**ENABLED**
Flap Detection:	**ENABLED**

FIGURE E.3 – Informations sur un service.

E.4 Grafana screenshot

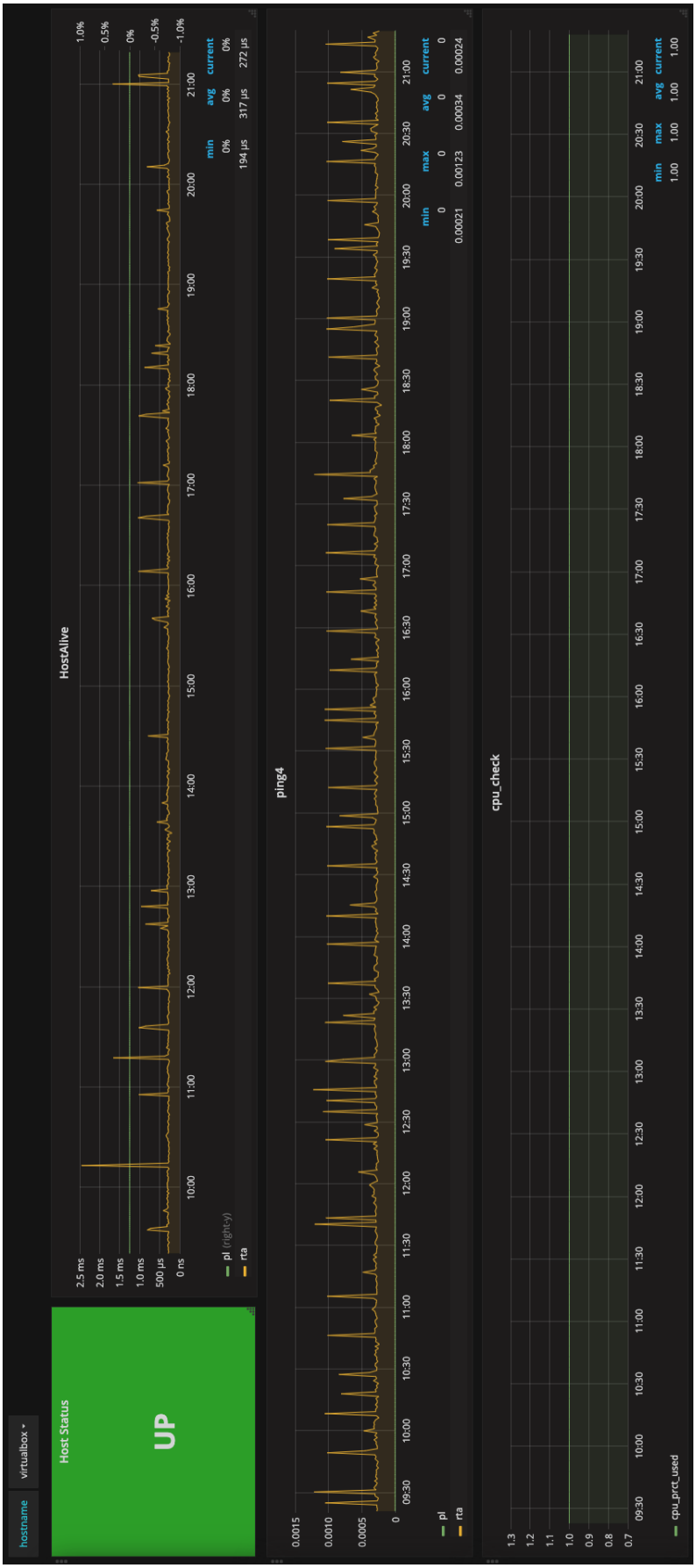


FIGURE E.4 – Vue du dashboard Grafana.

E.5 Stockage

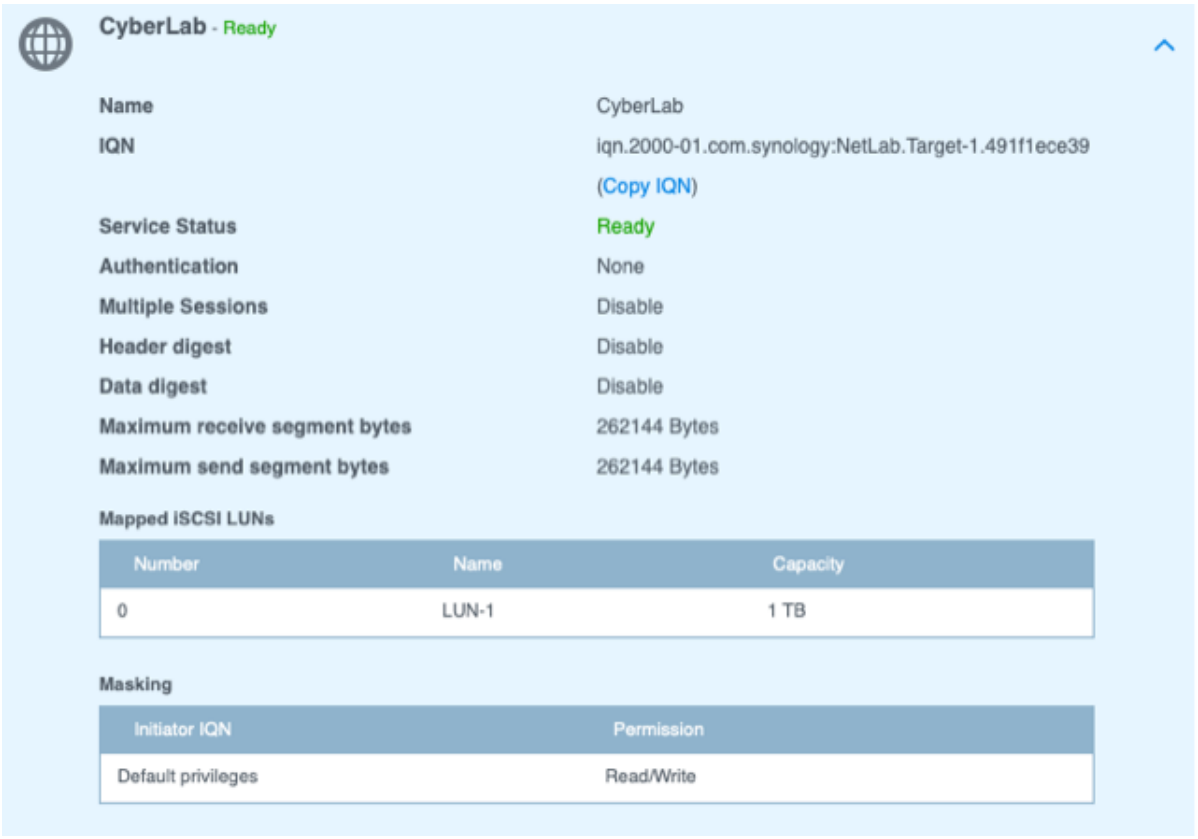


FIGURE E.5 – LUN Synology.

Bibliographie

[1] Introduction to packer. https://www.packer.io/intro/index.html.

[2] Introduction to vagrant. https://www.vagrantup.com/intro/index.html.

[3] Yavuz Aydin. Vmware vsphere 6.x – free and paid editions, 2019. https://www.snel.com/support/vmware-vsphere-6x-free-and-paid-editions/.

[4] David Bombal and Julien Duponchelle. Which emulator should i use ? https://docs.gns3.com/1o4IX8nXISl5gb4BwoSFrUht3MeTjkzHM1TCeWAe669g/index.html.

[5] Brendan. Metasploitable3. https://github.com/rapid7/metasploitable3/wiki.

[6] Yogesh Chandra and Pallaw Kumar Mishra. Design of cyber warfare testbed. *Institute for Systems Studies and Analyses*, 2019.

[7] Andrew Coleman and Julien Duponchelle. Install the gns3 vm on esxi, October 7th 2018. https://docs.gns3.com/1hEoK0rmtdBRnMaUaVoMHUbYwtDAltXYShiMJUp1GMxk/index.html.

[8] Jon Davis and Shane Magrath. A survey of cyber ranges and testbeds. *Cyber Electronic Warfare Division*, October 2003.

[9] Thibault Debatty and Wim Mees. Building a cyber range for training cyberdefense situation awareness. *Cyber Defence Lab*.

[10] Fabien Devaux, Christophe Fillot, MtvE, Gordon Russell, Jeremy Grossmann, and Flávio J. Saraiva. Dynamips (cisco router emulator). https://github.com/GNS3/dynamips/.

[11] VMWare Documentation. Expose vmware hardware assisted virtualization in the vsphere web client. https://pubs.vmware.com/vsphere-51/index.jsp?topic=%2Fcom.vmware.vsphere.vm_admin.doc%2FGUID-2A98801C-68E8-47AF-99ED-00C63E4857F6.html.

[12] VMWare Documentation. Forged transmissions. https://pubs.vmware.com/vsphere-4-esx-vcenter/index.jsp?topic=/com.vmware.vsphere.server_configclassic.doc_41/esx_server_config/securing_an_esx_configuration/c_forged_transmissions.html.

[13] VMWare Documentation. Mac address changes. https://pubs.vmware.com/vsphere-4-esx-vcenter/index.jsp?topic=/com.vmware.vsphere.server_configclassic.doc_40/esx_server_config/securing_an_esx_configuration/c_mac_address_changes.html.

[14] Joshua Eckroth, Kim Chen, Heyley Gatewood, and Brandon Belna. Alpaca : Building dynamic cyber ranges with procedurally-generated vulnerability lattices. *Proceedings of the 2019 ACM Southeast Conference*, pages 78–85, April 2019.

[15] James Forshaw. *Attacking Network Protocols*. No Starch Press, 2018.

[16] Waldemar Graniszewski and Adam Arciszewski. Performance analysis of selected hypervisors (virtual machine monitors - vmms). *INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS*, Volume 62(Issue 3) :231–236, September 2016.

[17] Jeremy Grossmann. Gns3 2.2.0 alpha 4 released !, April 5th 2019. `https://gns3.com/news/article/gns3-2-2-0-alpha-4-released?mkt_tok=eyJpIjoiWXpNNU9ERTRNbVpoWlRWbSIsInQiOiJXQ3dZVzNoZmdrWUFkaFVnbWNEeW1vYzVrV21JWW4zTTl6c3D`.

[18] Bill Hallaq, Andrew Nicholson, Richard Smith, Leandros Maglaras, Helge Janicke, and Kevin Jones. Cyran : A hybrid cyber range for testing security on ics/scada systems. *Security Solutions and Applied Cryptography in Smart Grid Communications*, pages 622–623, July 2016.

[19] David Kennedy. The pentesters framework (ptf), 2018. `https://github.com/trustedsec/ptf`.

[20] David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni. *Metasploit The Penetration Tester's Guide*. No Starch Press, 2011.

[21] Peter Kim. *The Hacker Playbook 3*. Secure Planet LLC, 2018.

[22] Peter Kim. *THE HACKER PLAYBOOK 3 : Practical Guide To Penetration Testing*. Red Team Edition, May 2018.

[23] Roberto Morabito, Jimmy Kjällman, and Miika Komu. Hypervisors vs. lightweight virtualization : a performance comparison. University of Washington.

[24] Ph. Marc Romainville. Didactique générale. Cours de didactique enseigné à l'UCLouvain.

[25] Amir Sharif. Esxi vs. esx : A comparison of features, June 1st 2009. `https://blogs.vmware.com/vsphere/2009/06/esxi-vs-esx-a-comparison-of-features.html`.

[26] Bohar Singh, Jagdeep Singh, and Sahil Kumar. Virtualization techniques and virtualization challenges in cloud computing : A review. *International Journal of Computing and Technology*, Volume 2(Issue 6) :175–180, June 2015.

[27] Kevin Soltow. Esxi is free... but, why do you buy esxi anyway ?, November 29th 2018. `https://www.vmwareblog.org/esxi-free-buy-esxi-anyway/`.

[28] Bhanu P Tholeti. Learn about hypervisors, system virtualization, and how it works in a cloud environment. September 2011. `https://developer.ibm.com/articles/cl-hypervisorcompare/`.

[29] Jack Wang. Why cisco virl is better than gns3 ?, October 16th 2016. `https://www.speaknetworks.com/cisco-virl-better-gns3/`.

[30] Georgia Weidman. *Penetration Testing*. No Starch Press, 2014.