Sécurité des Réseaux

Hardening d'un VPS Web

Monroe Samuel

16 novembre 2015

1 Introduction

La tendance actuelle du développement Web est celle du cloud computing, très intéressante tant il est facile de déployer un site dans le Cloud et ce en quelques dizaines de secondes. Plus besoin de devoir monter un serveur chez soi, de se préoccuper de paramètres tels que l'adresse IP ou encore l'achat de matériel spécifique, un débutant sans connaissances hardware ou réseau peut aujourd'hui voir son serveur web fonctionner et accessible en l'espace de quelques minutes.

Ce rapport va détailler ma recherche théorique dans le cadre du cours de sécurité des réseaux, et plus spécifiquement concernant le hardening qu'il est nécéssaire d'effectuer sur un VPS, appliquée à un cas imaginaire d'un utilisateur du cloud computing.

2 Situation Concrète

Michael "Mickey" Riffouille est un étudiant en informatique à l'EPHEC, passionné de web-development, au potentiel assuré d'auto-entrepreneur.
Après un rude hackathon au cours de sa deuxième année et suite aux débuts de projets d'intégration de troisième, il a eu une brillante idée : construire son propre site e-commerce pour **vendre des packs hackathons** pour organiser ses propres soirées coding entre amis.

Le site HackatHome (son site d'e-commerce) propose d'acheter des packs qui permettent d'organiser des hackathons chez soi (boissons énergisantes, tableaux SCRUMS, goodies).

Les utilisateurs seront donc invités à s'enregistrer sur le site, à indiquer leurs informations personnelles et de livraison, ainsi qu'à entrer un numéro de carte bancaire lors du paiement ou de procéder via paypal. Système classique que l'on retrouve sur la plupart des plateformes e-commerce, typiquement Amazon.
Ces informations seront stockées sur base de donnée, afin qu'un utilisateur puisse refaire des achats de packs facilement dans le futur, et même gagner des points de fidélité et obtenir des bonus si il effectue un cetain nombre d'achats.

Mickey a fait un plan de de tout ce qu'il allait mettre en place pour HackatHome :

— Un VPS de chez DigitalOcean avec un plan tarifaire de 10$ par mois.
— Un système d'exploitation FreeBSD 10.2
— Php comme langage pour le backend de son service
— MySQL pour sa base de données
— Projet versionné sur GitHub (repository privé)

Mickey développera son site de manière locale sur son portable, mettra en place l'infrastructure également depuis celui-ci en se connectant à son VPS via SSH.
Il est certain de versionner son code (il a entendu dire un jour : "Du code non versionné n'existe pas."), et le mettra sur GitHub pour peut-être envisager un déploiement automatique dans le futur.
Son VPS tournera avec FreeBSD, avec serveur web Apache et une combinaison Php/MySQL.

Enfin, il est à préciser que Michaël compte vivre de cette activité e-commerce, les retours qu'il a reçu sur sont projet sont très encourageants et il a quitté son job de super codeur HTML5 freelance pour se consacrer au dévelopement de HackatHome.

3 Présentation Technique

Le **hardening** d'un serveur est, comme son nom l'indique, le fait de le **renforcer** afin de préserver un certain niveau de sécurité minimum, en tenant compte d'une analyse de sécurité préalable et des requirements qu'elle aura mise en lumière.

3.1 FireWall

La première étape implémentable est celle de la configuration d'un firewall.
Ce VPS sous FreeBSD a un but de service précis et doit être administrable, il faut donc en limiter l'accès en ouvrant seulement les ports nécessaires à ces buts. Notre serveur n'a par exemple pas besoin d'être accessible via les ports FTP ou même Telnet.

Notre service a donc besoin des ports suivants :

— 22 pour le SSH, pour l'administration du système
— 80 pour le HTTP
— 443 pour le HTTPS
— Tout les autres ports seront simplement bloqués par défaut

Un des firewalls disponibles sur FreeBSD est **PF** (Packet Filter), un pare-feu réputé et d'origine OpenBSD, la syntaxe diffère quelque peu de IpTable de chez Linux.

La commande suivante permettra l'administration SSH sur le port 22 : **pass in on $ext_if inet proto tcp from any to ($ext_if) port 22**

3.2 Configuration de SSH

L'administration d'un VPS dans le cloud est forcément faite à distance.
Telnet n'étant pas une option valable du tout d'un point de vue sécurité, cela se fait donc via le protocole **SSH** ou Secure SHell.

SSH étant sécurisé car chiffrant les échanges de données, il faut impérativement configurer le service un peu plus en profondeur pour se prémunir d'autres risques.

3.2.1 Clés SSH

De base, le serveur SSH tournant sur le VPS propose une authentification par mot de passe.
Forcément, ce mode d'authentification tel quel est vulnérable aux attaques brute-force.

La solution est d'utiliser le système de clés SSH. L'utilisateur doit créer une paire de clés SSH via la commande :

ssh-keygen -t rsa -b 4096 -C "your_email@example.com"

Cette commande va créer une paire de clés publique/privée à protéger par une passphrase (mot de passe fort de préférence).
La clé publique doit être ensuite placée sur le VPS dans le fichier de clés autorisées associé au compte user du VPS.

Enfin, le serveur SSH du VPS doit être configuré pour ne plus accepter l'authentification par mot de passe, de sorte que seul un utilisateur possédant la clé privée puisse se connecter.

3.3 Certificat SSL

Ce certificat permet l'utilisation du protocole HTTPS port 443 pour un service web.
Notre cas étant un site e-commerce, cette mesure est essentielle pour assurer la sécurisation des transactions bancaires.

3.4 Fail2Ban

Fail2Ban est un service qui scan les fichiers de log et bannit une adresse IP s'il détermine que cette adresse adopte un comportement non légitime.

Il peut être configurer par exemple pour détecter des tentatives continues et infructueuses de connexions au serveur, et ainsi ajouter une règle au firewall pour rejetter tout traffic provenant de cette adresse.

3.5 IDS - Système de détection d'intrusion

Un système de détection d'intrusion va permettre à l'administrateur VPS d'agir si les mesures préventives n'avaient pas suffit et qu'une intrusion réussi, et ce en surveillant et analysant les events du système.

3.5.1 Tripwire

Tripwire est un IDS de type **Host-based IDS**, il se charge de surveiller un hôte et les évènements qui occurent sur celui-ci.

Lors de la configuration, cet IDS génère une base de donnée des fichiers système à surveiller qu'il va signer numériquement.

En fonctionnement, Tripwire compare la signature numérique qu'il possède en base de données avec des recalculs de signatures sur les fichiers.

En cas de non-correspondance des signatures, c'est que le fichier a été altéré et Tripwire va alerter l'administrateur.

3.5.2 Psad

Psad (Port Scan Attack Detection) est quant à lui un IDS de type **Network IDS**, surveillant le réseau et protocoles pour détecter une tentative intrusion.

Ce service surveille le firewall pour déterminer si une attaque via scan de ports est en cours, et en alerte l'administrateur.

4 Analyse de sécurité structurée

Cette section présentera d'abord un tableau de l'analyse des impacts d'une faille CIA sur les assets de HackatHome, ensuite une seconde analyse reprendra les risques encourus sur les assets et les contre-mesures qui pourraient être mises en place pour pallier à ces risques.

4.1 Assets et CIA (Confidentiality, Integrity, Availability)

Certains éléments de la triade CIA sont parfois omis car sans intérêt pour le bien analysé.

1. **VPS**
   Cet élément étant un service virtuel qui plus est, la valeur est assez difficile à estimer, j'estimerai donc ici sa valeur au coût mensuel que cela représente, c'est à dire 10$.
   — **A :** Impact **élevé**, une indisponibilité du service représente une perte d'argent vitale pour Michaël qui vit de cette activité.

2. **PC Portable**
   Son portable est un MacBook Pro de 2014 standard, acheté 1359 euros avec le statut étudiant.

   — **C :** Impact **élevé**, l'ordinateur contient énormément de données essentielles de la vie de Mickey et par extension, des données essentielles sur son travail.
   — **A :** Impact **faible**, bien que Mickey tienne à son beau MacbookPro, un terminal et un éditeur de texte sur une autre machien peuvent pallier à toute indisponibilité de sa station de travail.

3. **Système hôte FreeBSD 10.2**
   Ce système fait tourner l'activité de Michaël et est donc très importante, son indisponibilité mettrait un frein sur son chiffre d'affaire potentiel, tandis que sa compromission mettrait en péril les données clients et autre ressources critiques.

   — **C :** Impact **élevé**, le système hôte du service HackatHome contient forcément toutes les données relatives à l'activité, et une faille dans la confidentialité du système est inadmissible.
   — **I :** Impact **moyen à élevé**, tout dépend des éléments systèmes qui seraient touché par un problème d'intégrité.
   — **A :** Impact **élevé**, ce système faisant tourner par extension la plate-forme web, sont indisponibilité n'est pas envisagée.

4. **Plateforme Web**
   Comme pour le système hôte, la plate-forme web rapporte de l'argent à l'entreprise, et fait transiter des données sensibles pouvant représenter beaucoup d'argent.

   — **I :** Impact **faible à élevé**, encore une fois tout dépend de quels éléments sont affectés par la perte d'intégrité, des données corrompues au niveau html purement visuel auraient un impact faible, tandis que du code php altéré pourrait s'avérer catastrophique. Notons également que cela pourrait aussi impacter la réputation de HackatHome.
   — **A :** Impact **élevé**, disponibilité du service obligatoire.

5. **Données clients**
   Ces éléments sont difficilement chiffrables, il faudrait pouvoir calculer la somme qu'engendrerait des exactions sur les comptes clients après un vol des données bancaires, ou encore la somme qui pourrait être demandée en dédomagement suite à des poursuites judiciaires pour la mauvaise protection des données.
   On s'accordera sur le fait que, dans la mesure du raisonnable, tout doit être mis en oeuvre pour protéger ces données contre les risques classiques encourus par ce type de données.

   — **C :** Impact **très élevé**, ces données sont plus qu'importantes, une perte de confidentialité impacterait non-seulement le service mais plus conséquamment les utilisateurs qui auraient entré leurs informations bancaires sur le site, sans parler de la perte de réputation et même les éventuels risques pénaux.

— **I :** Impact **élevé**, ces données sont très importantes pour assurer le service, pas question d'envoyer un HackaPack Deluxe à une adresse erronée si Rémy Groet a payé 125$ pour celui-ci.
— **A :** Impact **moyen à élevé**, les données clés étant les adresses de livraison, leur indisponibilité mettrait au pire temporairement les livraisons en attente ou inviterait l'utilisateur à réessayer plus tard, ce qui impacterait également la réputation. Tout dépend de la portée qu'aurait cette indisponibilité des données sur le fonctionnement de la plate-forme.

6. **Argent - Système de paiement**
   HackAtHome a pour l'instant un chiffre d'affaire de 400 euros mensuels (c'est le début des activités), et le compte relié à celui-ci contient 6500 euros.
   De nouveau, des données clients critiques transitent par ce système et donc, potentiellement, sa valeur est très très importante.

   — **C :** Impact **très élevé**, si les données du système de paiement perdaient en confidentialité, on se trouve dans un cas similaire que pour les données client, l'impact serait catastrophique.
   — **I :** Impact **élevé**, une perte d'intégrité du système pourrait entraîner une sérieuse perte d'argent, par exemple si des commandes pouvaient être effectuées avec des paiements modifiés, etc.
   — **A :** Impact **moyen à élevé**, un peu près comme tout les autre composants du service, une non disponibilité pourrait entraîner une perte de réputation ou de clients potentiels.

4.2 Assets, risques et contre-mesures

Pour chaque bien, sera ici indiqué une structure reprenant les risques et les contre-mesures qui vont s'appliquer à ce risque.
De plus, suivra une liste de risques dits "résiduels", pour lesquels des contre-mesures ne seront pas mises en place mais qu'il importe de répertorier.

1. **VPS**
   — **Panne électrique chez DigitalOcean** :

   — Le risque est très **faible**, DigitalOcean est une société ayant une bonne réputation et son service est sûrement assuré contre ce genre d'évènement.
   Néamoins et selon la durée de cette panne et de la remise en service, l'impact pourrait être **élevé**.
   — Prise régulière de snapshots du serveur afin de backuper le serveur si jamais ce problème venait à arriver, celles-ci ne pèsent pas énormément en terme de données, Mickey pourra donc les stocker sur un disque dur qu'il possède déjà chez lui en plus de chez DigitalOcean. Le service de chez DigitalOcean lui coûterait 20 pourcents du montant du serveur en plus par mois.

   — **Défaut de paiement** :

   — Le risque est très faible, le plan de Mickey étant un plan à 10 dollars/mois, il y a donc très peu de chances qu'il ne puisse s'affranchir de ce montant. L'impact serait néanmoins assez élevé si DigitalOcean venait à lui couper le service.
   — On veillera à garder un approvisionnement automatique du compte facturé d'un montant supérieur à celui de la location du VPS, ce qui représente donc 20 dollars par mois, mais ne représente pas un coût réel. Michaël devrait aussi séparer ce compte afin, par exemple, de ne pas faire des achats via ce compte et passer sous la barre des 10 dollars avant facturation.

— **Risques résiduels sur le VPS** :

— **Destruction des serveurs DigitalOcean** : Risque faible, impact élevé
— **Cessation d'activité chez DigitalOcean** : Risque très faible, impact élevé
— **Perte ou fuite de données chez le fournisseur** : Risque faible, impact très élevé

2. **PC Portable**
— **Vol ou perte** :

— Le risque est **très élevé**, cela pourrait lui arriver n'importe où. L'impact serait lui également **très élevé** dans le sens où le voleur aurait une capacité beaucoup plus grande à devenir un risque potentiel pour d'autres assets.

— Mickey possède un Macbook, Apple permet à l'utilisateur de bloquer de manière assez sûre ses appareils à distance, il veillera donc à configurer ce service de façon à mettre hors service son appareil et empêcher toute utilisation en cas de vol.

— **Accès non autorisé** :

— Ce risque très élevé en cas de vol, plutôt **moyen** en temps normal.
L'impact serait cependant **très élevé**, puisque cet appareil contient des informations sensibles sur la vie et le travail de Michaël.

— Celui-ci veillera donc à utiliser des mots de passe forts pour tout ses comptes quels qu'ils soient, et également à mettre en place le verouillage automatique de son ordinateur après un temps assez court (dans la limite du confortable tout de même) d'inactivité.

— **Destruction** :

— Comme tout autre produit électronique, un ordinateur portable peut tomber en panne sévère, avoir une obsolescence programmée, ou l'utilisateur imprudent qu'est Michaël pourrait éventuellement renverser son Monster (sa boisson favorite) dessus, ou bien même encore le laisser tomber par terre. Le risque est donc **très élevé**.
L'impact est cependant **faible**, outre une perte sentimentale et d'argent assez désagreable, Mickey aime le cloud et a placé son code sur un repository privé Github, il n'a donc réellement besoin que d'un éditeur de texte, un terminal et une connexion internet pour travailler.

— Il est tout de même possible d'éviter ces situations en transportant l'ordinateur dans une housse à 20 euros de bonne qualité protégeant des chocs, en souscrivant à une assurance sur le matériel comme le AppleCare qui coûte tout de même 250 euro, et en évitant les gestes brusques si l'on désire boire en travaillant.

— **Malwares** :

— Le risques est très **faible**, Mickey est un informaticien averti et utilise son ordinateur pour travailler uniquement et n'installe que le strict minimum sur cette machine, l'impact reste quant à lui **élevé**.

— Il devra continuer à utiliser son ordinateur comme poste de travail uniquement et bien appliquer les mises à jours de sécurité de son système ainsi que de ses utilitaires. Il peut également souscrire à un Antivirus qui coûterait une quarantaine d'euros par an, et évidemment mettre en place le hardening de son serveur.

3. **Système hôte FreeBSD**
   - **Malwares** :
     - Michaël étant le seul administrateur sur ce système, le risque est relativement **faible** si on considère surtout les virus, trojans et autres malwares étant souvent dûs à une mauvaise utilisation d'une machine et à un comportement peu prudent.
       L'impact serait cependant très **élevé** puisque le serveur de production de HackatHome serait touché.
     - Il sera important pour Michaël de rester vigilent, de n'installer que le strict nécessaire sur son serveur de prod et d'éviter l'utilisation de plugins douteux pour se faciliter la vie.
       Il faudra aussi installer un Système de détection d'intrusion afin de détecter toute anomalie sur le processus normal de travail.
   - **Exploit** :
     - Un exploit ne peux jamais être écarté, l'impact serait également très **élevé** mais le risque est relativement **moyen**, Mickey a choisi FreeBSD pour sa grosse communauté et sa fiabilité.
     - Il lui importera de rester au courant des dernières mise-à-jour sécuritaires du système et de les appliquer.
   - **Accès non authorisé** :
     - Le risque est ici plus **élevé** car rendu possible via d'autres menaces (vol de son pc, fuites chez DigitalOcean), et l'impact serait bien sûr très **élevé** également.
     - Le système doit donc être rendu accessible via SSH uniquement, et il faut configurer le serveur SSH pour n'accepter qu'une connection via clé publique/clé privée. Plus encore, les passphrases des clés doivent être fortes.
       Comme précédemment, on installera un IDS afin de détecter toute intrusion dans le système.
   - **Risques résiduels** :
     - : Risque , impact
     - : Risque , impact
     - : Risque , impact

4. **Plateforme Web**
   - **Injections SQL** :
     - Le risque est très **élevé**, des formulaires sur une page web sont une invitation à ces injections, surtout que les gains potentiels en données seraient intéressants pour un cracker.
       L'impact d'une attaque par injection SQL serait très **élevé**.
     - Il conviendra de sécuriser au mieux les inputs PhP et les uploads fichiers avec les fonctions fournies par le langage, ainsi que de configurer sa base de donnée MySQL afin d'en limiter les droits en production.

   Risques résiduels :

— **Exploits** : Le risque est plutôt **moyen**, mais l'impact peut être très **élevé**.

— **Defacement** : Le risque dépend d'autres menaces et la manière dont les contre-mesures de celles-ci ont été mise en place, il reste néanmoins **moyen**, avec un impact **moyen** qui touchera surtout à la réputation du service mais laissera intact des pans plus critiques de celui-ci.

En contre-mesure globale, Michaël isolera son application dans une jail FreeBSD. En cas de compromission de celle-ci, l'impact pourra être limité ou du moins contrôlé via cette mesure.

5. **Données clients**
   — **Vol** :

      — Le risque dépend encore une fois des autres menaces et de la manière dont elles sont traitées, il est néamoins **élevé** avec un impact **élevé**.
      — Il est priommordial de sécuriser tout les moyens d'accès à ces données, depuis la machine de travail, base de données, vps tels que susmentionnés.

   — **Destruction** :

      — Le risque est assez **élevé** car cette menace peut même provenir de Mickey lui-même, on pourrait imaginer un **sudo rm -rf /** dans la mauvaise console un jour de très grosse fatigue, l'action d'un malware ou d'une injection SQL. L'impact est évidemment **très élevé**.

      — Les contre-mesures sur ces aspects ont été explorées dans les menaces précédentes.

6. **Argent - Système de paiement**
   — **Récupération des données lors de la transaction** :
      — Le risque est **élevé**, surtout si son système est fait-maison, avec un impact évidemment **très élevé**.

      — En contre-mesures, il conviendra déjà d'utiliser un certificat SSL sur le site pour éviter l'échange d'informations en clair, et pourquoi pas utiliser un système de paiement avec un tiers tel que **Stripe**.

5 Conclusion

Le hardening d'un VPS Cloud est un sujet qui peut s'avérer très vaste.
Il requiert une bonne connaissance du système d'exploitation, beaucoup de recherche et d'apprentissage, ainsi qu'une connaissance des services et technologies qui vont interagir au sein de ce système. Les compétences mises en oeuvre sont multiples, ainsi que les domaines à sécuriser.

L'équilibre entre le coût des mesures, le coût d'une menace potentielle, et le degré de sécurité qu'il est possible d'implémenter sans gêner le processus d'administration ou même l'expérience des utilisateurs est un juste milieu qu'il incombe de trouver.

6 Bibliographie

— **DigitalOcean** - https ://cloud.digitalocean.com/droplets/new
— **FreeBSD - Digital Handbook** - https ://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/
— **Computer Security** - William Stallings & Lawrie Brown
— **Syllabus Sécurité des Réseaux** - Virginie Van Den Schrieck
— **OpenSSH & Ubuntu** - https ://help.ubuntu.com/community/SSH/OpenSSH/Keys
— **Hack et défacement de sites web** - http ://www.hackingloops.com/6-ways-to-hack-or-deface-websites-online.html
— **Stripe - Paiements en ligne** - https ://stripe.com/be
— **PF on FreeBSD** - https ://www.freebsd.org/doc/en_US.ISO8859-1/articles/linux-users/firewall.html
— **SSH** - http ://doc.ubuntu-fr.org/ssh
— **Hardening VPS** - http ://www.hacksonville.com/2013/hardening-a-vps/
— **Certificat SSL** - https ://fr.wikipedia.org/wiki/Certificat_%C3%A9lectronique
— **Certificats SSL** - https ://www.globalsign.fr/fr/centre-information-ssl/definition-certificat-ssl/
— **Fail2Ban** - http ://www.fail2ban.org/wiki/index.php/Main_Page
— **Tripwire** - https ://fr.wikipedia.org/wiki/Tripwire_(logiciel)
— **Psad** - https ://www.digitalocean.com/community/tutorials/how-to-use-psad-to-detect-network-intrusion-attempts-on-an-ubuntu-vps