Projet sécurité et réseaux
Bilans personnels

Sommaire

Victor BONIN

Sur ce projet d'attaque réseau, mon rôle a été plus théorique que technique. Je me suis surtout renseigné sur le concept de l'attaque, son histoire, et les failles qu'elle implique.

Cela m'a permis de découvrir les failles potentielles historiquement liées aux serveurs web et les moyens développés pour répondre à ce type d'attaque.

Si le Malicious File Upload n'est plus une attaque couramment utilisée de nos jours, elle a été bien plus importante et dangereuse il y a quelques années. Aujourd'hui, l'immense majorité des frameworks à jours sont suffisamment stables pour ne plus être vulnérable à ce genre d'attaque, mais elle reste un classique intéressant à connaître et à comprendre et dont certaines parties (par exemple l'élévation des privilège) restent d'actualité.

Concernant les moyens de contrer l'attaque, je retiendrai la vérification du type du fichier, via l'extension ou via des fonctions php comme getimagesize() permettant de s'assurer de la conformité du fichier uploadé, la configuration du firewall pour vérifier les requêtes sortantes permettant le reverse shell, ou encore la restriction des droits dans les dossiers où sont uploadés les fichiers.

Finalement, j'ai trouvé le TP enrichissant car il permettait de découvrir par nous même des méthodes récentes non seulement théoriquement mais en plus d'expérimenter l'attaque, ce qui permet de bien mieux en retenir les concepts. Je regrette néanmoins qu'une séance supplémentaire n'aie pas été mise en place afin de permettre une meilleure synchronisation du groupe et un accompagnement par les enseignants.

Mathieu POURBAIX

Pendant ce TP j'ai participé aux choix de l'attaque, en effet ayant une forte affection pour le développement, les failles de sécurité que peut connaître un serveur web dus au code de l'application sont un domaine qui m'intéresse. Pendant la conception, concentré sur les autres PLD j'ai eu moins l'occasion d'implémenter moi même l'attaque mais j'ai pu m'intéresser au concept et aux outils que le groupe a utilisés. Ainsi j'ai pu m'intéresser à l'alternative plus simple qu'offre vagrant sur des outils plus modernes mais aussi plus complexe tel que Docker.

D'un point de vue apport personnel, j'ai pu apprendre beaucoup sur l'importance de toujours garder en tête pendant le développement de n'importe quel projet l'aspect sécurité de l'application. Des réflexes simples tel que la consultation régulières des bases de données d'exploit et l'utilisation d'outils maintenus. En effet, cela peut éviter simplement de mettre en péril toute la sécurité d'un projet à cause d'une faille déjà connue.

Romain GOUTTE-FANGEAS

Pendant ce TP, j'ai effectué un travail de recherche afin de pouvoir valider les étapes de l'attaques et construire un diaporama en accord avec l'attaque réalisée. J'ai pu tester l'attaque afin de valider les étapes et l'installation des logiciels. Je n'ai eu que peu de temps pour participer à la construction des machines virtuelles à cause des autres PLD en parallèle, ce que je regrette, mais j'ai pu aborder tous les concepts en testant cette attaque.

Ces démarches m'ont permis d'apprendre beaucoup sur ce domaine, notamment sur les exploits et les manières d'exploiter des failles de sécurité. J'ai pu voir que des nouveaux exploits sont publiés régulièrement sur internet, et sont corrigés au fil du temps. L'attaque que nous avons réalisé est donc obsolète car les technologies ont été mises à jour, mais cela reste intéressant pour découvrir les concepts qui sont derrière et participe à notre culture informatique.

Matthieu VALVIN

Durant ce projet d'attaque d'un serveur vulnérable, j'ai tout d'abord effectué un travail de recherche afin de déterminer vers quel type d'attaque nous allions nous diriger, quitte à prendre un certain nombre de connaissances sur des notions qui ne seront pas exploitées pour le projet final.

J'ai eu une montée en compétence assez significative sur ma vision générale du monde de la cybersécurité en générale. J'ai pu apprendre et exploiter mes connaissances sur de nombreuses technologies web. Pour ce qui est de l'attaque proposée dans le rendu final, mon rôle je suis resté assez éloigné des travaux de mise en place à proprement parlé. En effet, il a été décidé, pour des raisons d'efficacité et afin de respecter les deadlines des différents projets, que les tâches d'implémentations serait réparties.

J'ai cependant retenu la grande puissance et personnalisation de Vagrant qui permet de créer des environnements personnalisés et adaptés, ce qui est plus qu'utile dans le cas de l'implémentation de TP de cybersécurité. De plus j'ai trouvé ce TP enrichissant car il permet d'aborder le sujet de la cybersécurité sous un angle nouveau.

Paul-Emmanuel SOTIR

Ce TP m'a permit d'en apprendre plus sur comment mettre en place une attaque réaliste de bout en bout et prendre mieux conscience de la vulnérabilité de certaines technologies web ou de l'OS.

Ce projet a également été l'occasion de découvrir diverses technologies de réseaux, sécurité et de devop, comme MetaSploit, Zap, Chef ou Vagrant.

Malheureusement, je n'ai pas eu l'occasion de beaucoup contribuer à l'élaboration de l'attaque étant donné la répartition des rôles qui m'a amené à travailler majoritairement sur le PLD Comp.

Cependant, le projet de cybersécurité auras clairement une influence sur mes futurs choix devOps, notamment concernant les efforts pour s'assurer que les diverses technologies nécessaires pour la sécurité d'un serveur soient à jour et réputées sûres.

Baptiste DUBARRY

Configuration et mise en place des machines virtuelles

Bien qu'ayant compris les principes mis en oeuvres pour la configuration et le déploiement des machines virtuelles, je n'ai pas participé à cette phase du projet. Vagrant est un outil puissant qui nous permet de déployer des machines virtuelles préconfigurées mais cependant volumineux.

J'ai pu néanmoins effectuer les premiers tests de déploiement et pu modifier et détailler la procédure de configuration pour la rendre le plus claire possible.

Exploitation d'une vulnérabilité via l'upload de fichier

Ne m'étant jamais intéressé à la sécurité informatique auparavant, je me suis longuement documenté sur les "exploits" RFI en suivant plusieurs tutoriels afin de comprendre le fonctionnement global. J'ai ensuite testé sur l'environnement qui était préparé (serveur web vulnérable + attacker) et j'ai pu de nouveau déterminer certains bugs et permettre leur résolution. Malgré la vulnérabilité volontaire de notre serveur web, la simplicité du script.php uploaded qui nous permettait d'exécuter n'importe quelle commande shell directement depuis le serveur m'a surprise.

Découverte des payload et des générateurs

J'ai pu, grâce à notre attaque, découvrir et me familiariser avec des outils extrêmement puissant comme Metasploit qui sont incontournables dans le domaine de la sécurité informatique.

Dans notre cas, nous avons utilisé ce framework pour nous générer un payload python servant à initialiser un reverse shell. Je ne connaissais pas le concept du reverse shell et son utilisation, même si je n'ai pas pu me pencher en détail sur le code généré par Metasploit (payload ou même l'utilisation du reverse_tcp) j'ai pu appréhender les notions générales.

Privilege escalation

Cette dernière partie a été réalisée relativement tardivement comparé aux autres. Durant nos différentes recherches (RFI, LFI, Metasploit etc..) j'ai sans cesse vu les termes de "Privilege escalation" revenir. Notre attaque étant globalement fonctionnel nous nous sommes dit qu'il pourrait être intéressant d'implémenter un privilege escalation.

Après de nombreuses recherches nous avons réussi à trouver un exploit sur le kernel Linux utilisé par la VM de notre vulnerable web server ! Nous avons été en mesure de modifier un exploit trouvé il y a moins de 2 semaines afin de l'intégrer dans notre attaque.

Ce privilege escalation nous a permis d'obtenir un remote shell avec les privilèges root ce qui signifie le contrôle complet du web server. L'utilisation d'une faille aussi récente et l'adaptation à notre attaque a été réellement passionnante.

Conclusion

Les deadlines des autres PLD approchant énormément, je n'ai pas participé à la mise en place de l'environnement (vagrant, CDK, Berkshelf etc..) ni à la découverte des vulnérabilités RFI. Cependant j'ai pu jouer le rôle de testeur afin de détecter les bugs, le manque de clarté dans certaines explications. En me documentant et en effectuant un vrai travail de fond j'ai pu participer à la découverte, à la modification et à l'intégration de la partie Privilege Escalation.

Cette attaque, en tant que novice, m'a permis de me rendre compte de la puissance des attaques et des vulnérabilités existantes et de me plonger dans un monde inconnu et passionnant.

Corentin GIRAUD

Introduction

Au premier abord, je me suis dis que ce TP aller être difficile à réaliser de part la liberté de sujet à laquelle nous étions confrontés. En effet, le cadre du sujet était (volontairement ?) très libre ce qui nous a permi de choisir quelque chose qui nous intéressait réellement.

Comme indiqué lors de la présentation, notre principale volonté était de montrer l'utilisation d'outils professionnels de pentest afin de réussir à exploiter des vulnérabilités sur un système inconnu. Nous nous sommes donc mis d'accord sur le vecteur d'attaque et les différentes étapes que nous souhaitons réaliser à savoir:

1. L'explication basique du principe de la vulnérabilité exploitée (faille RFI)
2. La complexification de la vulnérabilité pour expliquer le fonctionnement d'un proxy local
3. L'utilisation du framework *metasploit* pour la génération de payload

4. Et enfin, l'exécution d'un exploit **extrèmement** récent pour réussir un élévation de privilèges

La découverte des des environnements de développement virtuel

Après avoir lu le mail décrivant le sujet, nous nous sommes attardés sur la manière dont nous allions rendre le projet. En effet, il nous a paru impossible de rendre les images des machines virtuelles à cause de leur taille trop importante (20 Go environ).

Nous avons donc dû mettre en place un un logiciel libre et open-source pour la création et la configuration des environnements de développement virtuel appelé *Vagrant*.

J'étais initialement parti sur la configuration de plusieurs *docker* mais le fait de ne pas connaître le concept de *container* à rendu la tâche bien trop complexe et j'ai donc dû abandonner l'idée rapidement à cause du manque de temps.

Je me suis occupé de cette partie et j'ai vraiment perdu du temps afin de trouver les bonne box vagrant sur lesquelles le projet allait se baser. Il a ensuite fallu comprendre ce qu'était le *provisionning* d'un machine virtuelle: le fait de la configurer (installer les outils nécessaires, type java, metasploit …). J'ai donc dû comprendre le fonctionnement d'un outil comme *Chef* avec le principe de recettes.

Enfin, il a fallu développer une application web volontairement vulnérable pour les besoins du TP. La compréhension du PHP n'a au final pas posé tant de soucis que ça et a été assez rapide.

La tâche de compréhension et de configuration des environnements a duré 20 heures environ.

La découverte d'un framework de pentest

Au cours de la création de ce TP, j'ai pu me re-découvrir le puissant framework de pentest Metasploit et ainsi bien comprendre le principe d'exploit, de payload, de post-exploitation … J'ai donc pu me rendre compte de la facilité d'utilisation de cet outil et donc du risque qu'il présente pour une utilisation mal-intentionnée.

Néanmoins, j'ai été un peu déçu de ne pas avoir pris le temps de comprendre plus en détail l'outil et notamment le fonctionnement détaillé de *msfvenom*, le générateur de payload utilisé dans le TP. Je me doute que le code dérrière ne doit pas être plus compliqué que ce que nous pouvons trouver ici: https://www.asafety.fr/reverse-shell-one-liner-cheat-sheet/ mais je n'ai pas eu l'occasion de vérifier.

La découverte de l'élévation de privilège

Cette étape a été implémenté dans le TP la veille de la présentation. En effet, nous avons réussi à obtenir un shell sur le serveur web vulnérable. Comment réussir un élévation de privilèges

sans installer de services vulnérables exécutés par *root*? J'ai donc pu me documenté un peu sur les différente manière d'aborder ce problème notamment sur le lien suivant: https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/.

Je me suis donc renseigné sur une potentielle faille du noyaux linux installé sur le serveur web vulnérable et BINGO!, il y en avait plusieurs. Il nous a fallu un peu de temps pour tester les différents exploits, en choisir un, et l'adapter aux besoins du TP.

Je vous avoue que je suis extrêmement satisfait de cette partie du TP. En effet, c'est la première fois que j'exploite un vulnérabilité non patchée sur un système professionnel (la box utilisé pour le serveur web propose une offre professionnelle https://box.scotch.io/ payante).

Encore une fois, je suis un peu déçu de ne pas avoir pris le temps de mieux comprendre l'exploit utilisé. En effet le code c est trop complexe à comprendre. J'ai juste réussi à comprendre qu'il se basait sur un *buffer overflow*.

La répartition du travail

Le nombre de personne au seins du groupe projet était trop important pour réussir à se répartir intelligemment le travail. Je suis vraiment passionné par le vaste domaine de la sécurité informatique, j'ai donc naturellement réaliser un travail supérieur au reste du groupe. J'avoue être un peu déçu du manque d'implication de certaine personne.

Conclusion

Ce TP fût un très bon prétexte pour me replonger dans la sécurité informatique et me confirme qu'il s'agit d'un domaine qui me passionne énormément. J'ai pu découvrir énormément d'outil que je ne connaissais pas ou peu. Enfin, même si j'ai essayé de vraiment comprendre le fonctionnement détaillé des attaques menées, je suis un peu déçu de ne pas avoir pris le temps d'approfondir certain point. Ce n'est que parti remise :)