



HoGent

Faculteit Bedrijf en Organisatie

Vulnerability scanners: een vergelijkende studie tussen openvas en nessus

Niels Van Driessche

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem

Instelling: —

Academiejaar: 2016-2017

Tweede examenperiode

Faculteit Bedrijf en Organisatie

Vulnerability scanners: een vergelijkende studie tussen openvas en nessus

Niels Van Driessche

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem

Instelling: —

Academiejaar: 2016-2017

Tweede examenperiode

Samenvatting

Voorwoord

Eerst en vooral wil ik de mensen bedanken die mij ondersteund hebben bij dit onderzoek:

- Van de Brempt Glenn: Voor het nalezen en aanpassen van dit onderzoek.
- Van Vreckem Bert: Voor de administratieve en technische hulp bij elke stap van dit onderzoek.

Ik heb dit onderwerp gekozen voor een paar redenen, maar de grootste reden is de licentie-verandering van nessus. Hierover vertel ik later meer, maar deze zat vroeger in de kali linux distributie. Deze is vervangen door openvas. Sinds ik graag test met verschillende security tools vroeg ik me af of deze 2 scanners van een gelijkwaardig kwaliteit zijn, en wat de belangrijkste verschilpunten hiervoor zijn. Een andere reden hiervoor is dat beveiliging steeds belangrijker wordt voor bedrijven. Dit heeft te maken met de verspreiding van een aantal grootschalige cryptolockers, en de impact die het op een bedrijf heeft.

Sinds security een constant veranderende wereld is, zijn papers hierover niet lang up to date. Ook zijn er weinig tot geen onderzoeken gevoerd vanuit een neutraal standpunt, hierdoor is dit volgens mij een perfect onderwerp voor mijn onderzoek. Ik kan mijn passies gebruiken terwijl ik een antwoord krijg op mijn persoonlijke vragen, en mensen laten inzien dat security een heel groot deel van elk bedrijf moet zijn.

Inhoudsopgave

1	Inleiding	9
1.1	Stand van zaken	11
1.1.1	Gelijkaardige onderzoeken	11
1.1.2	Hoe werkt een vulnerability scanner	12
1.1.3	Requirements	13
1.1.4	Nessus	14
1.1.5	Openvas	15
1.2	Probleemstelling en Onderzoeksvragen	17
1.3	Opzet van deze bachelorproef	18
2	Methodologie	19
2.1	Testen bepalen	19
2.2	Opzetten testomgeving	20

2.3	Installatie	20
2.3.1	Nessus	20
2.3.2	Openvas	21
2.3.3	Honeypots	22
2.3.4	Webservers	22
2.4	Instellingen scans	22
2.4.1	Nessus	22
2.4.2	Openvas	23
2.5	Scan resultaten	23
2.5.1	Nessus	24
2.5.2	Openvas	28
2.5.3	Resultaat	30
2.6	Scan resultaten valideren	30
2.7	Analyseren van de data	31
3	Conclusie	33
	Bibliografie	37

1. Inleiding

Security is de laatste jaren een echte trend geworden. Elk bedrijf is hiermee bezig, en ook in het nieuws komen er steeds meer berichten over beveiliging. Maar hoe zou een bedrijf hier zich best tegen kunnen verdedigen? Is een gewone firewall en antivirus niet genoeg meer?

De volgende stap is voor vele bedrijven een 'Vulnerability management' systeem. Dit is een continu proces dat bestaat uit 4 delen: 'discovery', 'reporting', 'prioritization' en 'response'. Tijdens het 'discovery' deel moet ELK systeem getest worden en in een databank geplaatst worden. Deze databank wordt gebruikt in de volgende stap 'reporting'. Hier moet er een rapport gemaakt worden zodat alle risico's in kaart gebracht kunnen worden. Eens dat de rapporten klaar zijn, moeten er 'priorities' opgesteld worden. Dit gebeurt door elk risico te bekijken en studeren wat de invloed is op het netwerk als deze misbruikt word. Uiteindelijk moeten deze risico's opgelost worden in het 'response' deel. „What is Vulnerability Management Anyway?” (2013)

Voor we beginnen uitleggen wat men hiervoor kan doen als verdediging, zullen er een paar termen uitgelegd worden dat essentieel zijn voor het verdere verloop van dit onderzoek. Een vulnerability is een zwakheid in een stuk code of een design fout (de meeste vulnerabilities komen van verouderde software en verkeerd geconfigureerde software) waardoor het mogelijk is om een actie uit te voeren wat niet de intentie van de software en/of het design was Rouse (2005b). Vulnerabilities die het mogelijk maken om informatie te krijgen van het systeem krijgen een CVE nummer. Dit is een uniek nummer voor een publieke vulnerability dat begint met het jaar waarin deze ontdekt werd. Deze nummers en namen zijn nodig omdat voor dit systeem bestond, elk bedrijf een andere naam had voor een vulnerability. Dit was onduidelijk en zorgde voor veel verwarring „About CVE” (g.d.). De acties dat gebruik maken van een vulnerability om bepaalde commandos uit te voeren op

een machine (Arbitrary code execution) of een server niet meer toegankelijk maken voor legitieme gebruikers (Denial of Service), noemen we exploits Rouse (2014).

Tools dat voor vulnerability management gebruikt kunnen worden noemen we scanners. In dit onderzoek zal ik alleen de 'Vulnerability scanners' bespreken. Dit is een beveiligingstechniek dat fouten in een systeem kan opsporen „Vulnerability Scanning” (g.d.) over een netwerk. Deze tools worden zowel gebruikt door 'black hats' als 'white hats'. Het verschil tussen deze 2 is dat de 'white hat' ethisch correct is. Deze zal op voorhand toegang vragen om een systeem te mogen testen, en als deze test lukt het bedrijf melden wat er mis is. De 'black hat' vraagt geen toestemming en gebruikt de testen om binnen te dringen op een systeem en de data hiervan te verkopen, of het systeem te laten crashen Hoffman (2013). Voor een volledig vulnerability management bij te houden, moet men ook nog andere scanners implementeren zoals een 'web application scanner' (bv. Arachni). Deze scant een website op mogelijke problemen als SQL injecties en cross-site scripting.

Een SQL injectie, is een vulnerability dat voorkomt als men data moet invullen op een website dat een actie uitvoert op de databank. Deze kunnen gaan van een wachtwoord van een gebruiker omzeilen, tot toevoegen of verwijderen van data **acunetix**

Cross-site scripting (vaak afgekort als XSS), is een techniek waarbij een hacker legitieme gebruikers kan aanvallen via een legitieme website. Deze techniek wordt vooral misbruikt bij javascript, omdat deze op bijna elke site aanwezig is. Dit gebeurt als er input van een gebruiker gebruikt wordt door bijvoorbeeld HTML of JS. Hier kan de aanvaller de gebruiker bijvoorbeeld doorverwijzen naar een valse website, en daar persoonlijke data van de gebruiker proberen te krijgen „Cross-site Scripting (XSS)” (g.d.).

Het resultaat van een vulnerability scan is meestal een lijst met 'cvss' scores van de gescande targets. Een cvss score is een framework om te bepalen hoe erg een bepaalde vulnerability is. Dit word bepaald door een complexe formule waarbij er rekening gehouden wordt met de impact op de target server, welke privileges je op de server reeds moet hebben, hoe complex de vulnerability is en van waar deze vulnerability misbruikt kan worden (lokaal netwerk, internet, fysische toegang, ...). Deze scores liggen tussen 0 en 10 en hebben aan de hand hiervan een bepaald label. Volgens versie 3 van cvss zijn de labels: low (0-3.9), medium (4-6.9), high (7-8.9) en critical (9-10) „NVD CVSS Support” (g.d.). Naast een cvss score, heeft het rapport ook een oplossing voor de vulnerability (indien deze bestaat). Deze kan bestaan uit het updaten van een bepaalde service, tot een configuratie file aanpassen.

Momenteel zijn er een aantal vulnerability scanners, maar voor dit onderzoek zal er alleen rekening gehouden worden met 'openvas' en 'nessus'. De reden hiervoor is dat nessus momenteel 1 van de populairste en bekendste tools hiervoor is dankzij zijn kwaliteit/kost verhouding „Top 125 Network Security Tools” (2015), maar deze heeft voor bedrijven wel een betalende licentie nodig „Nessus Professional” (2017). Openvas daarentegen is open-source en is gebaseerd op de laatste code die van nessus is vrijgegeven. Hierdoor kunnen we ook vergelijken of het mogelijk is om een open-source security producten te gebruiken in plaats van betalende software, en of deze dezelfde kwaliteit kan leveren.

Hoewel dit niet de primaire bedoeling is van dit onderzoek, is dit wel een mooie toevoeging. Een andere scanner die in aanmerking kwam voor dit onderzoek was CORE impact, deze wordt gezien als een van de beste en meest volledige scanners. De reden waarom deze niet onderzocht geweest is, is omdat deze een zeer hoge licentiekost heeft, en geen gratis versie heeft dat we konden gebruiken voor dit onderzoek.

1.1 Stand van zaken

In het volgende deel zal ik meer uitleg geven over hoe vulnerability scanners werken, ook refereer ik naar gelijkaardige onderzoeken en wat de verschillpunten zijn met mijn onderzoek. Hierna geef ik meer uitleg over nessus en openvas zelf, en wat de theoretische verschillen hiertussen zijn (zoals licenties, hardware,...). De praktische verschillen zullen we in het volgende hoofdstuk bespreken.

Elk onderzoek dat verband heeft met security is nooit lang up-to-date. Dit komt omdat deze constant veranderd en er steeds nieuwe technieken worden ontdekt waar men zich moet tegen beschermen. Een interessant voorbeeld van deze innovatieve technieken is over „Linkedin” (g.d.). Hierin slaagde een persoon om met css (Cascading Style Sheet), een webpagina van linkedin over te nemen. Dit werd mogelijk gemaakt door een bestaand css regel dat het hele scherm in gebruik nam. De aanvaller gebruikte via injecties deze regel om het hele scherm aanklikbaar te maken, en naar een willekeurige site te gaan.

Dit onderzoek is hier dus geen uitzondering. Na een paar maanden zal sommige informatie irrelevant zijn. Dit wil zeggen dat ook vorige onderzoeken al oude informatie kunnen gebruiken, waardoor men zich hierop niet kan baseren.

1.1.1 Gelijkaardige onderzoeken

Volgens de meeste onderzoeken die gevoerd zijn, is core impact de 'beste' scanner. Hiermee wordt bedoeld dat deze zeer snelle updates heeft met NVTs en een aantal tools ingebouwd heeft voor 'false positives' te testen, maar de licentiekosten bedragen rond de 30.000\$ / jaar. De meest populaire is zoals eerder vermeld nessus. „Vulnerability Scanners” (2016) && „Top 125 Network Security Tools” (2015)

Voor een vergelijking tussen openvas en nessus, is het aantal papers beperkt. Men geeft steeds dezelfde links naar het onderzoek „Nessus, OpenVAS and Nexpose VS Metasploitable” (2012). Dit onderzoek is gebeurd rond Juni 2012, voor security standards is dit dus heel outdated. Hoewel de vergelijking goed beschrijft hoe de scans gebeurd zijn en in welke environment dit gebeurd is, zijn er een aantal zaken die ontbreken. Zoals bijvoorbeeld een windows target. In het onderzoek heeft men alleen metasploitable gebruikt. Ook waren de 'extra' tools voor openvas niet geïnstalleerd en werd voor de nessus scan niet een volledige diepe scan uitgevoerd. In de comments werd het nessus probleem aangesproken door Paul Asadoorian. Deze werkt voor tenable en heeft een uitleg gegeven hoe je een volledige scan doet met nessus Asadoorian (2012). Merk op dat deze referentie partijdig is, sinds deze

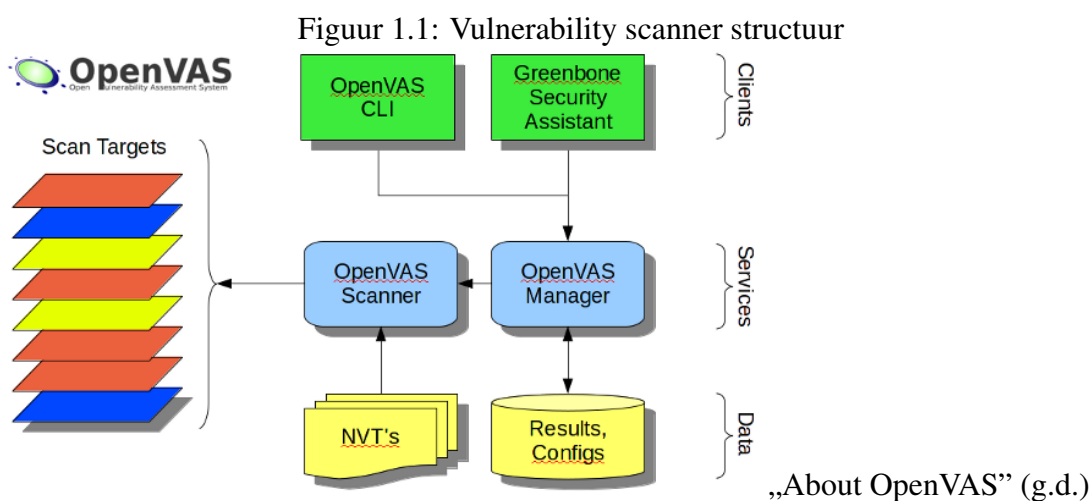
gegeven werd door een medewerker van nessus.

Een ander onderzoek „OpenVAS vs Tenable Nessus” (2011) gebruikt wel windows en linux targets, maar geeft geen informatie over hoe alle scans uitgevoerd zijn. Dit is een groot probleem als men niet voor elke scanner een gelijkaardig profiel gekozen heeft. Wel is dit onderzoek interessant omdat er een methode gebruikt werd om 'false positives' eruit te filteren. Deze is helaas niet meer beschikbaar in de huidige versie van metasploitable, en is dus ook niet meer toepasbaar op dit onderzoek.

Andere gevoerde onderzoeken werden uitgevoerd als nessus nog open source was en zijn dus ook niet toepasbaar op dit onderzoek, of waren van een te lage kwaliteit om deze te analyseren.

1.1.2 Hoe werkt een vulnerability scanner

Een vulnerability scanner bestaat traditioneel uit 3 delen. Een user interface, een manager en een scanner. Eerst zal er een kleine uitleg gegeven worden over hoe alle delen met elkaar werken, en daarna een gedetailleerde beschrijving hoe een target gescanned word.



De user interface (hierboven vermeld als 'Greenbone Security Assistant') is in veel gevallen een webinterface dat acties doorstuurt naar de manager, zoals een scan starten, of rapporten maken. Het is ook mogelijk om acties door te sturen naar de manager via een command line interface (hierboven vermeld als 'OpenVAS CLI'), dit is handig voor als men scans wil automatiseren.

De manager (hierboven vermeld als 'OpenVAS Manager') staat in voor de communicatie tussen de user interface en de scanner engine. Deze onderhoud ook een aantal databanken waarin alle configuraties, resultaten, SCAP data, CERT data en NVT's zitten. SCAP, CERT en NVTs zijn hierbij de belangrijkste, deze staan in voor het testen van een target of deze vulnerable is. SCAP data is een feed waarbij de community helpt deze te ontwikkelen „The Security Content Automation Protocol (SCAP)” (2017). Het verschil met CERT data is dat deze eerder voor interne scans bedoeld is, door een serie van testen uit te voeren

met een aangemelde gebruiker „Insider Threat” (g.d.). Uiteindelijk zijn er nog NVTs, deze zijn geen vaste feeds waar iedereen dezelfde data krijgt. Deze hangt af per vulnerability scanner. Later in dit onderzoek zullen we deze verschillen duidelijk zien met openvas en nessus. Het is de job van de manager om deze databanken up-to-date te houden zodat deze steeds een volledige vulnerability management oplossing kan bieden.

De scanner (hierboven vermeld als 'OpenVAS Scanner') staat in voor de effectieve scan van een target. Deze heeft een cache waar alle NVT's aanwezig zijn die de scanner nodig heeft om een target te kunnen scannen. Deze geeft zijn resultaten terug aan de manager.

Indien men een scan start voor een range van machines, zal de scanner engine eerst een 'alive check' uitvoeren op alle hosts in de range. Dit kan bestaan uit een simpele ICMP ping naar de host, tot een range van poorten scannen en kijken of de host reageert op 1 van de requests.

Indien de host reageert, zal er een port scan gestart worden, de tool die men hiervoor gebruikt is in veel gevallen 'nmap'. De range van poorten wordt op voorhand gedefinieerd door de gebruiker en indien er een firewall aanwezig is, zal hier ook rekening mee gehouden worden.

De scanner zal hierna een aantal dingen analyseren, zoals het besturingssysteem dat op de host draait, en welke services overeen komen met de open poorten. Als al deze feiten uiteindelijk bekend zijn, zal de scanner op basis van deze analyse NVTs naar de hosts sturen en kijken of deze vatbaar is voor bepaalde exploits „How does vulnerability scanning work?” (2015).

Uiteindelijk zal de scanner alle gevonden informatie doorsturen naar de manager, en op basis hiervan een rapport maken die leesbaar is voor de gebruiker.

1.1.3 Requirements

In de volgende delen, zullen we nessus en openvas vergelijken op een aantal vlakken dat belangrijk zijn in een scanner. Een aantal belangrijke zaken (snelheid van de scanner, accuraatheid, etc...) zullen pas besproken worden in hoofdstuk 2, omdat deze tot het onderzoek behoren. Eerst zal er kort een aantal zaken besproken worden zoals de geschiedenis van de scanner, de licenties, de support dat de scanner krijgt en de hardware vereisten. Hierna bespreken we de belangrijkste zaken.

Een volwaardige vulnerability scanner moet aan een paar voorwaarden voldoen, het belangrijkste onderdeel zijn de NVTs. Als deze niet tijdig worden geüpdate, is de scanner niet meer betrouwbaar voor de nieuwste vulnerabilities, en geeft dit een vals veilig gevoel. Hiervoor zullen we 2 grote vulnerabilities (1 voor linux, en 1 voor windows) opzoeken en kijken wanneer beide scanners deze NVT toegevoegd hebben. Voor linux heb ik gekozen voor de 'shellshock' vulnerability (CVE-2014-6271), voor windows is dit 'Server Service Vulnerability' (CVE-2008-4250)

Uiteindelijk bespreken we ook de functionaliteiten van beide scanners. Eerst zullen we

bepaalde functionaliteiten bespreken die een betere performantie geven, of een accurater beeld geven van het systeem. Daarna bespreken we bepaalde functies die niet direct bijdragen tot het eindrapport, maar de scanner gemakkelijker laten implementeren met de bestaande infrastructuur of het grootste deel van het onderhoud automatiseren.

Voor het eerste deel gaan we na of beide scanners een 'credential scan' ondersteunen, deze test gebeurt door het inloggen op een bepaalde dienst (meestal ssh) en heeft dus ook de mogelijkheid om configuratie files te controleren op fouten Peterson (2010). Een basisvereiste is ook vaak dat scanner meerdere scans tegelijkertijd kunnen uitvoeren, Hoewel dit bij moderne scanners zo goed als steeds het geval is gaan we voor volledigheid dit ook nakijken. Uiteindelijk kijken we ook of de scanner 'agent based' scanning toestaat, dit is een klein programma dat op een host geïnstalleerd wordt, en commandos van een centrale machine kan ontvangen. Deze zal een scan uitvoeren op de lokale machine, en zijn resultaten doorsturen naar de centrale machine waar deze beschikbaar is via alle interfaces.

Uiteindelijk kijken we of de server zichzelf onderhoud (zoals NVTs updaten), en wat er nog manueel moet gebeuren. Ook kijken we of er een REST interface, of een API aanwezig is. Hierdoor is het mogelijk om acties uit te voeren op de scanner server, zonder enige extra software te moeten installeren op het systeem.

1.1.4 Nessus

Geschiedenis

Nessus is in 1998 opgericht door Renaud Deraison, met de bedoeling om een gratis vulnerability scanner te maken dat over het internet machines kon scannen. Rond deze software werd het bedrijf tenable opgericht en werd al snel 1 van de populairste oplossingen voor vulnerability management. Vanaf nessus versie 3 (2005) werd de source code niet meer publiek gemaakt omdat er misbruik van gemaakt werd door de concurrentie. De nessus scanner werd doorverkocht door bedrijven met hun eigen logo en vermelden dat het hun scanner was, ook was er weinig tussenkomst van de community (vooral op gebied van de scanner engine) LeMay (2005).

Licenties

Tenable heeft een aantal nessus producten, maar voor dit onderzoek kijken we naar 'Nessus Professional'. De reden hiervoor is dat dit product zo goed als volledig overeenkomt met een openvas installatie (met een paar limitaties). De licentiekosten hiervoor zijn 2190 \$ / jaar (ongeveer 2010 euro) en is proprietary software.

Support

Als men een licentie van 'Nessus Professional' heeft, is het mogelijk om support te krijgen via live chat, email en een support portal. Deze zaken zijn 24/7 te benaderen „Nessus

Support” (g.d.).

Nvt development

Nessus heeft op moment van schrijven 87.071 plugins „Nessus plugins” (g.d.). Deze NVTs worden geschreven in .nasl (Nessus attack script language).

Voor de shellshock vulnerability is er een NVT verschenen op 24-09-2014 „Nessus shellshock” (g.d.). Dit is dezelfde dag als shellshock publiek gemaakt werd. De windows vulnerability NVT werd geüpload op 23-10-2008, dit is ook op de dag dat windows de vulnerability publiek maakte.

Functionaliteiten

Credential scan: Deze functionaliteit is beschikbaar, met ondersteuning voor een aantal services. De belangrijkste services worden hier opgelijst: database, mongoDB, ssh, windows en SNMPv1/v2c „Nessus products” (g.d.).

Meerdere scans: Dit is mogelijk op elke versie.

Agent based: Dit is niet mogelijk met de 'professional' licentie, maar wel met de 'manager' licentie „Nessus products” (g.d.). **Onderhoud:** Op de webinterface is het mogelijk om bepaalde of alle elementen te updaten. Ook is het mogelijk om een interval in te stellen wanneer deze moet geüpdate worden (dagelijks, wekelijks of maandelijks).

Hardware

De minimum systeemvereisten voor een scanner zijn:

- Dual core 2GHz CPU
- 2-4 GB RAM
- 30GB HDD

„Nessus Hardware Requirements” (g.d.)

1.1.5 Openvas

Geschiedenis

Openvas is een fork van de laatste source code die nessus publiek heeft gemaakt in 2005, en begon onder de naam „GNessus - GPL based Nessus” (g.d.) en werd opgestart door Tim Brown. Dit project is nog steeds volledig opensource „Source Code Documentation” (2017) en gratis voor zowel personen, als bedrijven met uitzondering op een (niet verplichte) betalende licentie van greenbone voor NVT's „About NVT Feed” (g.d.). De grootste bijdrager voor openvas is greenbone, deze zorgt voor een groot deel van de NVTs, de webinterface en bugfixes „Contributors” (g.d.).

Licenties

Openvas valt onder de GPL license, is open-source en is gratis te gebruiken voor zowel personen, als bedrijven.

Support

Een nadeel van open-source projecten is vaak dat je afhankelijk bent van de community voor support. Dit is bij openvas niet anders, buiten een mailing list en een IRC is de support zo goed als niet bestaande. Een 'Internet relay chat (IRC)' is een online chatroom waar een reeks gebruikers zich aanmelden om over een bepaald onderwerp te Discussiëren „IRC” (g.d.). In dit geval is het onderwerp openvas, en heeft de IRC gemiddeld 2000-3000 lijnen per uur „OpenVAS Online Chat” (g.d.). Een mailing list is een lijst van mensen die zich met een email geabonneerd heeft. Hier verstuurt een andere gebruiker of bedrijf een email naar over een bepaald onderwerp zodat deze mensen up-to-date blijven met alle nieuwigheden. Een voorbeeld hiervan is de openvas mailing list, deze verstuurt een mail telkens dat er een nieuwe versie beschikbaar is. Deze mail bevat alle informatie met wat er veranderd is ten opzichte van de vorige versie Rouse (2005a).

De openvas website bevat veel verouderde links zoals een wiki en een aantal screenshots. Op het moment van schrijven is de laatste entry in de wiki voor openvas 7 „OpenVAS Documentation” (2014), ook op de screenshots pagina is de laatste entry voor openvas 7 „Screenshots” (g.d.).

Nvt development

Openvas heeft op het moment van schrijven 53.082 plugins „About NVT Feed” (g.d.) waarvan er 214 geüpload zijn in de laatste maand. Wat hier ook opvallend is, is dat alle plugins in openvas geschreven worden in nasl (dit staat voor 'nessus attack script language'). Openvas gebruikt dus nog steeds dezelfde code voor NVTs te schrijven als nessus.

Voor de shellshock vulnerability heeft openvas de NVT geupload op 25-09-2014 „Openvas shellshock” (g.d.). Dit is 1 dag later als de announcement van shellshock en is gemaakt door greenbone. Dit lijkt onschuldig, maar shellshock heeft 24 uur na de ontdekking al voor het ontstaan van botnets gezorgd Greenberg (2014).

De windows vulnerability werd beschikbaar gesteld op 24-10-2008, Dit is 1 dag na de bekendmaking door windows. Hoewel de gevolgen niet zo erg waren als shellshock, gebruikte de worm „Conficker” (g.d.) deze vulnerability om zich te verspreiden. Een worm is een stuk software dat zichzelf automatisch kan verspreiden, waarbij het meestal leidt tot het installeren van backdoors of data kan stelen van een gebruiker „What is a Computer Worm?” (g.d.). Een backdoor is een techniek waardoor het mogelijk wordt om een prompt te krijgen op een systeem zonder dat de gebruiker hiervan weet „Backdoor” (g.d.).

Functionaliteiten

Credential scan: Deze functionaliteit is beschikbaar, maar dit is niet op scan niveau. Dit betekent dat men 2 verschillende targets (systeem dat gescanned zal worden) moet aanmaken als men een credential en een non-credential wil starten (ookal is dit hetzelfde IP). De ondersteunde services zijn hier: SSH, SMB, ESXi en SNMP.

Meerdere scans: Dit is mogelijk.

Agent based: Dit is mogelijk.

Onderhoud: Het is niet mogelijk om in de webinterface updates te starten. De updates (voor alles) worden wel automatisch elke dag geüpdate om 1 uur snachts door middel van cronjobs. Crond is een daemon dat draait op de achtergrond en dat toestaat op een bepaald interval een actie uit te voeren indien de gebruiker hier permissies voor heeft „cronjob” (g.d.). Deze kan gebruikt worden voor bijvoorbeeld gegevens op te halen, of in het geval van puppet op een bepaald interval een puppet run te starten.

Hardware

De minimum systeemvereisten voor openvas staan niet op de site, er is wel een virtuele machine aanwezig voor openvas te testen. De requirements hieronder zijn van deze virtuele machine, in realiteit zullen deze requirements hoger liggen. De reden hiervoor is dat dit een minimale versie is dat bepaalde features niet heeft, ook zijn er nog meer beperkingen zoals dat het besturingssysteem kan niet updaten. Dit is dus niet gepast voor een productie omgeving.

- Dual core CPU
- 2 RAM
- 9GB HDD

„Openvas Hardware Requirements” (g.d.)

1.2 Probleemstelling en Onderzoeksvragen

Idealistisch zou elk bedrijf hoe groot en klein dan ook, beveiliging als een prioriteit moeten beschouwen. In de realiteit is dit helaas niet zo, veel (vooral kleine) bedrijven bezien dit als een bonus (of erger: een extra kost) hoewel dit een grote impact kan hebben op de verdere toekomst van het bedrijf. Stel bedrijf X, een IT bedrijf dat klantgegevens bijhoudt en een kleine database heeft voor een website. Bedrijf x heeft een oude versie van de apache webserver, en deze wordt niet geüpdate omdat de server blijft werken. Op een dag is de server gecrashed en in de log files ziet men dat een aanvaller toegang gekregen heeft tot het systeem (door een vulnerability in apache). Hierdoor stuurt bedrijf x naar alle klanten een bericht dat men hun wachtwoord moet veranderen omdat de wachtwoorden niet gesalt waren, en dus mogelijk achterhaald kunnen worden. Veel mensen zullen misnoegd zijn en

bedrijf X niet meer vertrouwen, hierdoor zullen hun inkomsten dalen.

Dit is natuurlijk een overdreven voorbeeld, maar het laat zien waarom beveiliging zo belangrijk is. Dit onderzoek zal aantonen dat er gratis oplossingen zijn voor bedrijven om te beginnen met vulnerability management, en of deze vergelijkbaar zijn met een 'professionele' en betaalbare software. Concreet betekent dit dat mijn onderzoeksvragen als volgt zijn:

-

1.3 Opzet van deze bachelorproef

De rest van dit onderzoek is als volgt opgebouwd:

In hoofdstuk 2 wordt het effectieve onderzoek gevoerd. Deze is opgebouwt in een aantal grote stukken. Het eerste stuk is bepalen hoe we deze scanners zullen vergelijken. Daarnaast zetten we ook een testomgeving op die de scans op een veilige manier laten gebeuren.

Hierna volgt de installatie van beide scanners, waar we vergelijken hoeveel manuele stappen elke scanner nodig heeft. Als we eenmaal een volledige omgeving hebben, bespreken we het belang van de verschillende soorten scans. Uiteindelijk voeren we de testen uit en valideren / analyseren we de gevonden data.

In Hoofdstuk 3, tenslotte, wordt de conclusie gegeven en een antwoord geformuleerd op de onderzoeksvragen. Hierbij zal ook het belang ugelegd worden van deze studie, en waarom het belangrijker is dat dit onderwerp steeds meer onderzoek krijgt.

2. Methodologie

2.1 Testen bepalen

In dit onderzoek zal er met elke scanner 4 scans uitgevoerd worden op elk target. De reden hierachter is dat er in een professionele omgeving 2 punten zijn die belangrijk zijn. Zowel nessus als openvas hebben hiervoor aparte profielen die focussen op een ander deel. Naast deze 2 scans is een credential scan ook een noodzaak om een inzicht te krijgen hoe goed een systeem lokaal beveiligd is (indien men toegang en permissie heeft om deze te scannen met bv. ssh). Voor elke scanner zal er dus een snelle credential, accurate credential, snelle uncredential en een accurate uncredential scan uitgevoerd worden. Dit zal gebeuren in een paar omgevingen die we hieronder bespreken.

Eerst testen we dit op honeypots, dit zijn servers die intentioneel een zwakke configuratie hebben die veel vulnerabilities toont op de scanner. Het doel hiervan is dat deze heel goed opgevolgd moeten worden, en van zodra men activiteit op de server opmerkt dit onderzoeken en kijken of er onbevoegde persoon / malware aanwezig is. Deze test zal bepalen hoeveel vulnerabilities op een zwak systeem gevonden worden voor zowel een linux, als een windows machine.

Hierna zullen de scanners gebruikt worden om een webserver te scannen over het internet. Deze test is een goede weergave voor een deel van een penetration test. We verwachten op de server zelf weinig tot geen fouten, maar de extra informatie die de scanner ons geeft zoals het besturingssysteem of welke diensten er draaien op het systeem. Deze informatie kan gebruikt worden voor andere aanvallen zoals social engineering. Dit is een techniek waarbij een aanvaller zich bijvoorbeeld voordoeft als een werknemer van een bedrijf, en op deze manier toegang probeert te krijgen tot het interne systeem. Een ander voorbeeld hiervan is dat een aanvaller een mail kan versturen en zich voordoeft als een admin, om zo

credentials te krijgen.

2.2 Opzetten testomgeving

Voor dit onderzoek werd er gebruik gemaakt van 4 virtuele machines op Virtualbox versie 5.1.10. Voor de virtuele machines waar er een vulnerability scanner op moest draaien, hebben we gebruik gemaakt van een CentOS minimal image. Deze kregen ook dezelfde instellingen die hieronder vermeld worden:

- 8000 MB ram
- 4 CPUs
- NAT & host-only interfaces

Voor de honeypots gebruikten we een iets minder zware virtuele machine, de instellingen hiervoor worden hieronder ook vermeld.

- 1000 MB ram
- 2 CPUs
- Host-only interface

De fysieke computer waar deze testen op zijn uitgevoerd heeft 16GB ram en een cpu met 4 cores en 8 threads. Tijdens de testen waren er geen andere programmas actief die een impact zouden kunnen hebben op de resultaten. Ook werden alleen de machines die getest moeten worden aangezet zodat beide scanners geen impact op elkaar konden hebben.

2.3 Installatie

2.3.1 Nessus

Voor nessus te downloaden op de virtuele machine (met enkel CLI), moeten we naar de productpagina gaan. Sinds we geen extra software op de virtuele machine wouden (zoals FTP of SMB), zullen we wget gebruiken voor de .rpm file te downloaden. Dit is niet zo simpel, sinds de download link een pop up weergeeft voor de juiste versie te kiezen. Na dit probleem op te zoeken, is het duidelijk dat we bepaalde waarden met wget moeten meegeven *COMMAND*.

Na de installatie van de RPM file (38 MB), poort 8834/tcp te openen in firewall-cmd en de service nessusd te starten is de webinterface bereikbaar. Hier moeten we de licentiecode ingeven die we verkregen hebben bij het aanvragen van een nessus home licentie. Hierna volgt een download scherm dat +- 7 minuten duurt. Hierna kunnen we beginnen met scannen.

2.3.2 Openvas

Voor openvas te downloaden op de virtuele machine (met enkel CLI), hebben we 2 opties. We kunnen deze van source compilen, of de „OpenVAS Binary Packages (3rd party)” (g.d.). Wij gebruiken de packages, omdat dit minder tijd in beslag neemt. Voor deze packages te downloaden gebruiken we de atomicorp repository, omdat hiernaar verwezen word op de openvas site.

We installeren de packages voor openvas 9 via yum. De totale grootte van alle packages is 70 MB (185 dependencies). Na deze te downloaden moeten we alles installeren via command line, hiervoor gebruiken we 'openvas-setup'. Deze download eerst alle NVTs (28 MB, verplicht), maar na de download crashed het script omdat we de package 'bzip2' niet geïnstalleerd hebben. We starten het script terug op en merken dat we alle NVTs opnieuw moeten downloaden. Hierna geeft het script een niet fatale error 'certool not found', maar het script gaat verder en blijft vasthangen na 'verify admin password'. Op de achtergrond is het script een databank aan het maken met alle NVTs in, na deze stap is het script gedaan. Het script heeft 23 minuten in beslag genomen.

Na poort 9392/tcp open te zetten in firewall-cmd is het *niet* mogelijk om te connecteren op de webinterface. We gebruiken een ander script 'openvas-check-setup -v9' om te kijken wat er mis is. Deze geeft weer dat de redis server niet gestart is. Deze zit in een failing state, en wil niet herstarten. Na de log files te raadplegen hebben we selinux op 'permissive' gezet, en werkt de redis server nu wel.

Na de check-config nog eens te overlopen, blijkt dat de services van openvas momenteel niet aanstaan. Bij deze opmerkingen staan er tussen haakjes openvassd en openvasmd. Na deze commandos uit te voeren, blijkt de services nog steeds niet te draaien. Na dit verder te onderzoeken, blijkt dat de service openvas-scanner, openvas-manager en gsad noemen. we starten deze op en runnen de check-config opnieuw.

De config geeft nogmaals een probleem, deze keer omdat de database geen tot weinig gegevens bevat. Hiervoor moeten we het commando 'openvasmd -rebuild' starten. Na 4-5 minuten is deze gedaan en draaien we de check-config nogmaals. Nu geeft deze weer dat er geen SCAP en CERT data aanwezig zijn. Hoewel dit *niet* verplicht is, installeren we deze gegevens voor de volledigheid van de scanner. De sync moet gebeuren met rsync, maar sinds deze sync veel files bevatten die niet groot zijn gaat dit heel traag. Deze duurden samen ongeveer 22 minuten en nemen +-800MB in beslag.

Als al deze gegevens aanwezig zijn, draaien we het commando check-config nog eens. Deze geeft weer dat selinux moet gedisable worden. We proberen deze eerst op 'permissive' te zetten, maar het script geeft nog steeds een foutmelding. Na een reboot van de server voor selinux uit te zetten, geeft het script weer dat er een paar packages die **optioneel** zijn niet aanwezig zijn (zoals alien en net-tools). We installeren deze ook zodat deze geen invloed kunnen hebben op de eindresultaten van de scans.

Uiteindelijk geeft het script weer dat de openvas installatie klaar is voor gebruik. Na het bezoeken van de webinterface blijkt dat dit nog steeds niet mogelijk is. Na te troubleshooten

blijkt dat de error in het begin van openvas-setup (certool not found) hier schuldig voor is. Na de package 'gnutls-utils' te installeren en nieuwe certificaten te genereren, is het mogelijk om de webinterface te raadplegen.

Als we de CLI bekijken, blijkt dat deze niet werkt omdat we niet werken met socket files. We lossen dit op door openvas op localhost te laten luisteren in plaats van een .sock file te gebruiken (openvasmd -a 127.0.0.1), hierna werkt de CLI zoals het hoort.

2.3.3 Honeypots

Voor de linux honeypot hebben we 'metasploitable' genomen. Dit is een linux machine die gebaseerd is op ubuntu 8.04 en een reeks services heeft draaien. Deze services zijn opzettelijk zo zwak mogelijk geconfigureerd zodat men op deze virtuele machines vulnerability scanners kunnen testen en zo leren een penetration test uit te voeren.

Voor de windows honeypot hebben we een windows xp machine zonder service packs geïnstalleerd, maar met een aantal servers (SMB,SMTP,SNMP, FTP en IIS). Vooral door deze missende service packs hebben deze veel problemen + er zijn geen opzettelijk slecht geconfigureerde machines zoals metasploitable omdat windows xp onder een licentie valt.

2.3.4 Webservers

2.4 Instellingen scans

2.4.1 Nessus

Nessus heeft in de gratis versie geen profielen voor zowel een snelle scan als een diepe scan. Voor dit onderzoek heb ik voor beide scans zelf een profiel opgesteld. Voor de snelle scan heb ik onderstaande settings gebruikt, deze zorgt dat er een snelle scan geleverd wordt, maar met nog steeds accurate resultaten.

- alive test = ICMP
- network type = private LAN
- port list = default
- Local port enumerators = SSH,WMI,SNMP
- port scan technique = SYN
- Scan web applications (enable generic tests)
- Scan for malware

Voor de diepe scan heb ik een aantal testen aangezet waardoor er een mogelijkheid is dat een bepaalde service stopt met werken. Ook gebruik ik hier een brute-force aanval, waar zwakke gegevens zullen weergegeven worden. De volledige settings ziet u hieronder:

- alive test = ICMP,TCP, ARP
- network type = private LAN

- port list = 0-65535
- Local port enumerators = SSH,WMI,SNMP, only network port scan when local failed
- Port scan technique = SYN
- Search for SSL = all ports
- Perform thorough tests = yes
- Test default accounts = yes
- Scan web applications (enable generic tests)
 - Try all http methods
 - Attempt http parameter pollution
 - Test embedded web servers
- Scan for malware
- Enable safe checks = no

Deze settings zijn zowel voor de windows machine, als de linux machine. Hier is er slechts 1 verschil, namelijk credentials. Voor de linux machine is er een SSH authenticatie, terwijl er voor de windows machine SMB gebruikten. Beide accounts hadden admin / root toegang op het systeem.

2.4.2 Openvas

In tegenstelling tot nessus heeft openvas een reeds voorgemaakte profielen. Voor de snelle scan gebruiken we de 'full and fast' scan methode. Hier worden bijna alle NVTs gebruikt, alleen niet diegene dat ervoor kunnen zorgen dat het systeem onstabiel wordt. Samen met de 'All IANA assigned TCP 2012-02-10' port list (5625 poorten). Dit is een lijst van poorten dat geassocieerd zijn met een 'bekende' service. Dit betekent niet dat dit steeds de service is dat er op draait, maar dient meer als een richtlijn voor een unieke poort te kiezen voor een eigen service. Als alive test hebben we hier voor 'Consider alive' geopteerd, omdat we zeker zijn dat dit systeem zal reageren.

Voor de diepe scan hebben we het profiel 'full and very deep ultimate' genomen. Deze bevat *alle* NVTs (ook diegene dat het systeem onstabiel kunnen maken). Samen met alle TCP poorten (0-65535) en de 'consider alive' optie. Ook hier gebruiken we voor de credential scan SSH voor linux, en SMB voor windows.

Wat ons hier opmerkt is dat openvas standaard voor al zijn voorgemaakte scans een TCP connect scan uitvoert, in tegenstelling tot nessus die standaard SYN gebruikt. SYN is een heel stuk sneller, maar kan onbetrouwbare resultaten geven als het systeem dit als een aanval herkent.

2.5 Scan resultaten

In dit deel worden alle resultaten besproken tegenover elk target met elke scanner. Hierbij wordt alleen rekening gehouden met vulnerabilities dat een cvss score hebben van 2

of hoger (dus geen logs). De gevonden vulnerabilities met als rating 'critical' zullen ook kort uitgelegd worden (indien er gelijkaardige resultaten of duplicaten zijn, zal dit vermeld worden). Op het einde zal ik de beide scanners vergelijken op basis van de gevonden vulnerabilities. Deze resultaten worden in het volgende hoofdstuk gebruikt om te controleren of er geen 'false positives' aanwezig zijn. Alle resultaten die hieronder besproken worden zijn getest op 14 Maart, met de nieuwste NVTs die publiek beschikbaar zijn.

2.5.1 Nessus

Voor de credential scans zijn er een aantal opties. Voor de metasploitable credential scan hebben we gekozen voor SSH, omdat dit simpel te gebruiken is, en op bijna alle machines reeds aanwezig is. Voor de windows credential scan hebben we gekozen voor een 'Windows login', hier hebben we een administrator account gegeven waardoor nessus de nodige files kon opvragen.

Metasploitable: snel zonder credentials

- 6 critical
- 8 high
- 36 medium
- 10 low

Deze scan heeft 17 minuten in beslag genomen.

'Debian OpenSSH/OpenSSL Package Random Number Generator Weakness' (CVE-2008-0166): Deze vulnerability komt 2 keer voor en kan ervoor zorgen dat een aanvaller een prompt krijgt via het netwerk, of een man in the middle aanval kan uitvoeren (poort 25 & 22). Dit zorgt ervoor dat een aanvaller alle data kan lezen die tussen de client en server verstuurd word.

'rexecd Service Detection' (CVE-1999-0618): Rexecd heeft weinig tot geen authenticatie, met geen encryptie. Dit kan gebruikt worden om vanuit het netwerk commandos te versturen naar de host waarop deze draait (poort 512).

'Rogue Shell Backdoor Detection' (**geen** CVE): Nessus heeft een poort ontdekt waarop het mogelijk is commandos te versturen met root rechten, zonder enige vorm van authenticatie (poort 1524).

'Unix Operating System Unsupported Version Detection' (**geen** CVE): Deze host maakt gebruik van een end of life besturingsysteem (ubuntu 8.04), en krijgt dus geen belangrijke updates meer.

'VNC Server "password>Password' (**geen** CVE): Men kan inloggen in de VNC server met het wachtwoord 'password'. Hierdoor kan een aanvaller een prompt krijgen (port 5900).

Metasploitable: snel met credentials

- 8 critical
- 7 high
- 29 medium
- 10 low

Deze scan heeft 11 minuten in beslag genomen. Dit is 6 minuten minder als de scan zonder credentials. Dit is te verklaren door het feit dat nessus om netwerk gebruik en tijd te besparen, het commando 'netstat' uitvoert. Hierdoor kan nessus alle poorten zien die openstaan, en op basis hiervan de scan verder uitvoeren. Hieronder zijn de extra vulnerabilities die nessus ontdekt heeft met de credential scan.

'Bash Remote Code Execution (Shellshock)' (CVE-2014-6271 & CVE-2014-7169): Een zeer ernstige bug (vergelijkbaar met de meer recentere *heartbleed* bug) dat ervoor zorgt dat een aanvaller toegang krijgt tot een machine, dit kwam door de manier waarop bash functies afhandelt. De gevolgen van deze bug zijn reeds besproken in het deel *inleiding - requirements*.

'Weak Debian OpenSSH Keys in ~/.ssh/authorized_keys' (CVE-2008-0166): Er is een zwakke ssh key aanwezig in de authorized_keys file. Hierdoor kan een aanvaller een brute-force aanval gebruiken om toegang te krijgen tot het systeem. Een brute-force aanval is (in de meeste gevallen) een aanvaller die met behulp van een programma een lijst van wachtwoorden afgaat in de hoop dat deze toegang krijgt tot een account, maar in dit geval gaat het over een 'exhaustive key search'. Hier zal een aanvaller in plaats van zich op het wachtwoord te richten proberen de 'key' te raden, waardoor hij het gewone wachtwoord van de gebruiken kan achterhalen.

Zoals u hierboven kunt zien zijn er 3 unieke vulnerabilities toegevoegd, terwijl er maar 2 resultaten meer zijn. De vulnerability van ubuntu end of life wordt niet meer weergegeven.

Metasploitable: diep zonder credentials

- 6 critical
- 10 high
- 40 medium
- 10 low

Deze scan heeft 42 minuten geduurt, en heeft geen extra critical vulnerabilities opgeleverd. De extra 2 high vulnerabilities zijn een SQL injectie, en een command execution. Deze zijn dus beiden gerelateerd aan de extra web server scans.

Metasploitable: diep met credentials

- 9 critical
- 10 high
- 38 medium

- 8 low

Deze scan heeft 44 minuten geduurt, en heeft slechts 1 extra vulnerability extra gegeven. Deze is een extra verzameling van CVE's over shellshock (CVE-2014-6277 & CVE-2014-6278).

Windows: snel zonder credentials

- 17 critical
- 5 high
- 10 medium
- 3 low

Deze scan heeft 3 (!) minuten geduurt.

'ASN.1 Multiple Integer Overflows (SMTP check)' (CVE-2003-0818): Hierdoor kan een aanvaller code uitvoeren op een systeem. Hiervoor moet men een speciaal SMTP authenticate packet versturen.

'Microsoft IIS 6.0 & Windows XP Unsupported Version Detection' (**geen** CVE): End of life voor de webserver en het besturingssysteem, op zich is dit geen vulnerability. Deze informatie kan wel gebruikt worden om andere vulnerabilities te vinden.

'Cumulative Update for Microsoft RPC/DCOM' (CVE-2003-0352 & CVE-2003-0715 & CVE-2003-0528 & CVE-2003-0605 & CVE-2003-0813 & CVE-2004-0116 & CVE-2003-0807 & CVE-2004-0124 & CVE-2008-4250): Dit is een reeks van vulnerabilities die een reeks patches nodig hadden om volledig opgelost te kunnen geraken, hierdoor zijn er een reeks verschillende CVE nummers. Deze exploit maakt het mogelijk voor een aanvaller om code uitvoeren met systeem rechten. Deze vulnerability reeks werd gebruikt om een aantal worms te verspreiden zoals blaster en lovesan.

'ASN.1 Vulnerability Could Allow Code Execution' (CVE-2003-0818): Deze vulnerability zorgt ervoor dat een aanvaller code kan uitvoeren op het systeem, door een speciaal NTLM (een reeks protocollen dat instaat voor de veiligheid van ingelogde gebruikers) packet te versturen.

'Security Update for Microsoft Windows' (CVE-2003-0533): Een bug in LSASS (het deel van windows dat zorgt voor de beveiliging, en het aanmaken van accounts) zorgt ervoor dat een aanvaller commandos kan uitvoeren met systeem rechten.

'Microsoft Windows Task Scheduler Remote Overflow' (CVE-2004-0212): Nogmaals een exploit om op afstand code te kunnen uitvoeren op een machine. Deze keer door een bug in de 'Task scheduler' van windows.

'Vulnerability in SMB Could Allow Remote Code Execution (CVE-2005-1206 & CVE-2008-4834 & CVE-2008-4835 & CVE-2008-4114): Deze vulnerability komt 2 keer voor. Een aanvaller kan (zonder the authenticeren) code uitvoeren door een bug in het SMB protocol. Hieronder volgt nog een 'speciale' SMB vulnerability waar meer uitleg nodig is.

'ETERNALBLUE, ETERNALCHAMPION, ETERNALROMANCE, and ETERNAL-SYNERGY SMB vulnerabilities' (CVE-2017-0143 & CVE-2017-0144 & CVE-2017-0145 & CVE-2017-0146 & CVE-2017-0147 & CVE-2017-0148): Deze reeks vulnerabilities zijn reeds onlangs aan het licht gekomen. Ze worden actief misbruikt op het internet door de 'wannacry' worm en de verschillende varianten. Deze bug (die gevonden werd door NSA, maar niet gedeeld werd met microsoft) heeft voor een ongeziene verspreiding van de cryptolocker gezorgd. Hiermee is het mogelijk om code op een systeem uit te voeren, en verschillende malware programma's op de host te downloaden.

'Vulnerability in Printer Spooler Service Could Allow Remote Code Execution' (CVE-2005-1984): Door deze bug kan een aanvaller de spooler service laten crashen, of in het ergste geval code uitvoeren op het systeem.

'Vulnerabilities in MSDTC Could Allow Remote Code Execution' (CVE-2005-2119 & CVE-2005-1978 & CVE-2005-1979 & CVE-2005-1980 & CVE-2006-0034 & CVE-2006-1184): Deze exploit komt 2 keer voor. Door een vulnerability in de MSDTC kan een aanvaller verschillende exploits toepassen (bv. local privilege escalation waarbij een reeds ingelogde gebruiker toegang krijgt tot meer permissies).

Windows: snel met credentials

- 18 critical
- 5 high
- 8 medium
- 3 low

Deze scan heeft 2 (!) minuten geduurt.

Het enige verschil (voor de critical vulnerabilities) is 'MS03-043: Buffer Overrun in Messenger Service (CVE-2003-0717)'. Hoewel deze in dezelfde patch wordt opgelost als de printer spooler service, heeft de uncredential scan deze niet gedetecteerd. De andere vulnerabilities komen overeen.

Windows: diep zonder credentials

- 18 critical
- 5 high
- 10 medium
- 3 low

Deze scan heeft 8 minuten geduurt, en bevat geen veranderingen ten opzichte van de vorige scan.

Windows: diep met credentials

- 18 critical

- 5 high
- 10 medium
- 3 low

Deze scan heeft 10 minuten geduurt, en bevat geen veranderingen ten opzichte van de vorige scan.

2.5.2 Openvas

Ook voor openvas hebben we SSH gekozen voor de metasploitable credential scan, maar voor windows is dit ingewikkelder. Hier moeten we een SMB user geven, maar na het testen hiervan waren er geen verschillen tussen de credential en non-credential scans.

Openvas gaf geen error terug dat de credentials niet juist waren, maar na onderzoek blijkt dat men eerst een uitvoerbaar bestand (.exe) moet downloaden via openvas, deze op de machine die je wilt scannen te openen zodat de gebruiker aangemaakt wordt. Na deze stappen te ondernemen was er nog steeds geen verschil tussen de 2 scans. Voor openvas moet ook de optie 'easy file sharing' niet aanstaan (enkel op windows XP machines). Hierna gaf de scanner een opmerkelijk aantal resultaten meer ten opzichte van de non-credential.

Metasploitable: snel zonder credentials

- 24 high
- 42 medium
- 4 low

Deze scan heeft 18 minuten geduurt. Wat hier opvallend is, is dat er geen rekening met de 'critical' categorie gehouden werd. Dit duidt op het gebruik van cvss versie 2 dat nog geen critical categorie had. Deze versie heeft een compleet andere formule voor het bekomen van het resultaat, en zorgt ervoor dat deze moeilijker te vergelijken zijn met de 3de versie die in 2015 in gebruik genomen werd.

Metasploitable: snel met credentials

- 121 high
- 161 medium
- 14 low

Deze scan duurde 19 minuten.

Metasploitable: diep zonder credentials

- 27 high
- 44 medium

- 4 low

Deze scan duurde 37 minuten.

Metasploitable: diep met credentials

- 124 high
- 165 medium
- 14 low

Deze scan duurde 39 minuten.

Windows: snel zonder credentials

- 7 high
- 11 medium
- 0 low

Deze scan duurde 11 minuten.

Windows: snel met credentials

- 252 high
- 56 medium
- 2 low

Deze scan duurde 14 minuten.

Windows: diep zonder credentials

- 9 high
- 12 medium
- 0 low

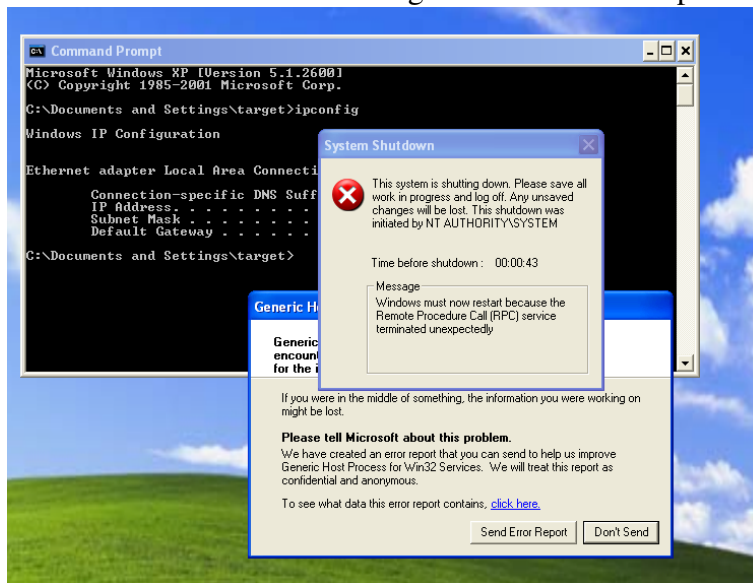
Deze scan duurde 21 minuten. Terwijl de scanner bezig was, is er op de windows xp machiene de volgende boodschap verschenen:

Hierdoor moest het systeem herstarten. Op het einde van deze scan als de machine herstart was, was het niet mogelijk om een actie uit te voeren hierop. Het besturingssysteem reageerde compleet niet meer, en moest dus opnieuw herstart worden.

Windows: diep met credentials

- 254 high
- 57 medium
- 2 low

Figuur 2.1: Windows xp crash



Deze scan duurde 25 minuten. Ook hier deed het scenario van de vorige test zich voor (2 keer herstarten).

2.5.3 Resultaat

2.6 Scan resultaten valideren

Een scanner kan veel vulnerabilities tonen, maar als meer dan de helft van de resultaten false positives zijn, heeft het geen nut om een scan uit te voeren. Hoewel 'false positives' zeker een probleem zijn, is er een groter probleem dat niet direct opvalt. Dit wordt mooi samengevat met de quote: "The real problem is not false positives, it's false negatives", hiermee bedoelt men dat het niet erg is als een scanner soms eens een false positive geeft, zolang hij geen echt vulnerabilities negeert.

Helaas is er geen complete lijst van vulnerabilities voor zowel windows xp als metasploitable. Een reden hiervoor is dat men constant nieuwe vulnerabilities vindt, hierdoor zou de lijst na een aantal maanden of jaren niet meer relevant zijn. Ook de methode die gebruikt werd in voorgaande onderzoeken (met behulp van autopwn in metasploit), bestaat momenteel niet meer. Hierbij is er nog een extra uitdaging omdat nessus met cvss verise 3 werkt, terwijl openvas nog steeds versie 2 gebruikt. Dit kan ervoor zorgen dat de score die een vulnerability krijgt zeer drastisch veranderd.

Een voorbeeld hiervan is de windows xp machine. Nessus geeft een 'low' resultaat omdat er een X server edetecteerd werd, terwijl dit bij openvas een 'high' vulnerability is. Om toch enige methode van filter te gebruiken, zullen we alle 'critical' vulnerabilities van nessus en de vulnerabilities met een score van 10 tot 9 in openvas onderzoeken.

Hiervoor zullen we vergelijken of een vulnerability van 1 scanner te vinden is in de andere

scanner (zonder rekening te houden met de cvss score). Op deze manier kunnen we nakijken dat de scanners dezelfde vulnerabilities ontdekken, ondanks de verschillende cvss versies. Indien er een vulnerability is dat slechts door 1 scanner wordt ontdekt, zullen we onderzoeken of dit niet over een 'false positive' gaat.

2.7 Analyseren van de data

3. Conclusie

Bibliografie

- Asadoorian, P. (2012). The Right Way To Configure Nessus For Comparison. Verkregen van <http://securityweekly.com/2012/08/24/the-right-way-to-configure-nessus/>
- Greenberg, A. (2014). Hackers Are Already Using the Shellshock Bug to Launch Botnet Attacks. Verkregen van <https://www.wired.com/2014/09/hackers-already-using-shellshock-bug-create-botnets-ddos-attacks/>
- Hoffman, C. (2013). Hacker Hat Colors Explained: Black Hats, White Hats, and Gray Hats. Verkregen van <https://www.howtogeek.com/157460/hacker-hat-colors-explained-black-hats-white-hats-and-gray-hats/>
- LeMay, R. (2005). Nessus security tool closes its source. Verkregen van <https://www.cnet.com/news/nessus-security-tool-closes-its-source/>
- About CVE. (g.d.). Verkregen van <https://cve.mitre.org/about/>
- About NVT Feed. (g.d.). Verkregen van <http://www.openvas.org/openvas-nvt-feed.html>
- About OpenVAS. (g.d.). Verkregen van <http://www.openvas.org/about.html>
- Backdoor. (g.d.). Verkregen van <https://www.techopedia.com/definition/3743/backdoor>
- Conficker. (g.d.). Verkregen van <https://www.microsoft.com/security/portal/threat/encyclopedia/entry.aspx?Name=Win32%2fConficker>
- Contributors. (g.d.). Verkregen van <http://www.openvas.org/contributors.html>
- cronjob. (g.d.). Verkregen van <http://www.adminschoice.com/crontab-quick-reference>
- Cross-site Scripting (XSS). (g.d.). Verkregen van <https://www.acunetix.com/websecurity/cross-site-scripting/>
- GNessus - GPL based Nessus. (g.d.). Verkregen van <http://www.securiteam.com/tools/6V00B0AEAW.html>
- Insider Threat. (g.d.). Verkregen van <http://www.cert.org/insider-threat/>
- IRC. (g.d.). Verkregen van <http://www.webopedia.com/TERM/I/IRC.html>
- Linkedin. (g.d.). Verkregen van <https://security.linkedin.com/blog-archive#11232015>

- Nessus Hardware Requirements. (g.d.). Verkregen van <https://docs.tenable.com/nessus/Content/HardwareRequirements.htm>
- Nessus plugins. (g.d.). Verkregen van <http://www.tenable.com/plugins/index.php?view=all>
- Nessus products. (g.d.). Verkregen van <http://www.tenable.com/products/nessus-vulnerability-scanner>
- Nessus shellshock. (g.d.). Verkregen van https://vulners.com/nessus/BASH_REMOTE_CODE_EXECUTION.NASL
- Nessus Support. (g.d.). Verkregen van http://www.tenable.com/products/nessus/nessus-faq#Nessus_Support
- NVD CVSS Support. (g.d.). Verkregen van <https://nvd.nist.gov/vuln-metrics/cvss>
- OpenVAS Binary Packages (3rd party). (g.d.). Verkregen van <http://www.openvas.org/install-packages.html>
- Openvas Hardware Requirements. (g.d.). Verkregen van <http://www.openvas.org/vm.html>
- OpenVAS Online Chat. (g.d.). Verkregen van <http://www.openvas.org/online-chat.html>
- Openvas shellshock. (g.d.). Verkregen van <https://vulners.com/openvas/OPENVAS:804489>
- Screenshots. (g.d.). Verkregen van <http://www.openvas.org/screenshots.html>
- Vulnerability Scanning. (g.d.). Verkregen van <https://www.techopedia.com/definition/4160/vulnerability-scanning>
- What is a Computer Worm? (g.d.). Verkregen van <http://www.pctools.com/security-news/what-is-a-computer-worm/>
- OpenVAS vs Tenable Nessus. (2011). Verkregen van <http://rageweb.info/2011/04/13/openvas-vs-tenable-nessus/>
- Nessus, OpenVAS and Nexpose VS Metasploitable. (2012). Verkregen van <https://hackertarget.com/nessus-openvas-nexpose-vs-metasploitable/>
- What is Vulnerability Management Anyway? (2013). Verkregen van <https://www.tripwire.com/state-of-security/vulnerability-management/what-is-vulnerability-management-anyway/>
- OpenVAS Documentation. (2014). Verkregen van https://wiki.openvas.org/index.php/Main_Page
- How does vulnerability scanning work? (2015). Verkregen van <https://community.qualys.com/docs/DOC-1068>
- Top 125 Network Security Tools. (2015). Verkregen van <http://sectools.org/tag/vuln-scanners/>
- Vulnerability Scanners. (2016). Verkregen van <https://www.concise-courses.com/hacking-tools/vulnerability-scanners/>
- Nessus Professional. (2017). Verkregen van https://store.tenable.com/index.php?main_page=product_info&cPath=1&products_id=94
- Source Code Documentation. (2017). Verkregen van <http://www.openvas.org/src-doc.html>
- The Security Content Automation Protocol (SCAP). (2017). Verkregen van <https://scap.nist.gov/index.html>
- Peterson, D. (2010). credentialed-scanning. Verkregen van <http://www.digitalbond.com/blog/2010/01/25/3-reasons-you-should-be-using-credentialed-scanning/>
- Rouse, M. (2005a). mailing list. Verkregen van <http://whatis.techtarget.com/definition/ mailing-list>
- Rouse, M. (2005b). vulnerability. Verkregen van <http://whatis.techtarget.com/definition/ vulnerability>

Rouse, M. (2014). exploit. Verkregen van <http://searchsecurity.techtarget.com/definition/exploit>

Lijst van figuren

1.1	Vulnerability scanner structuur	12
2.1	Windows xp crash	30

Lijst van tabellen