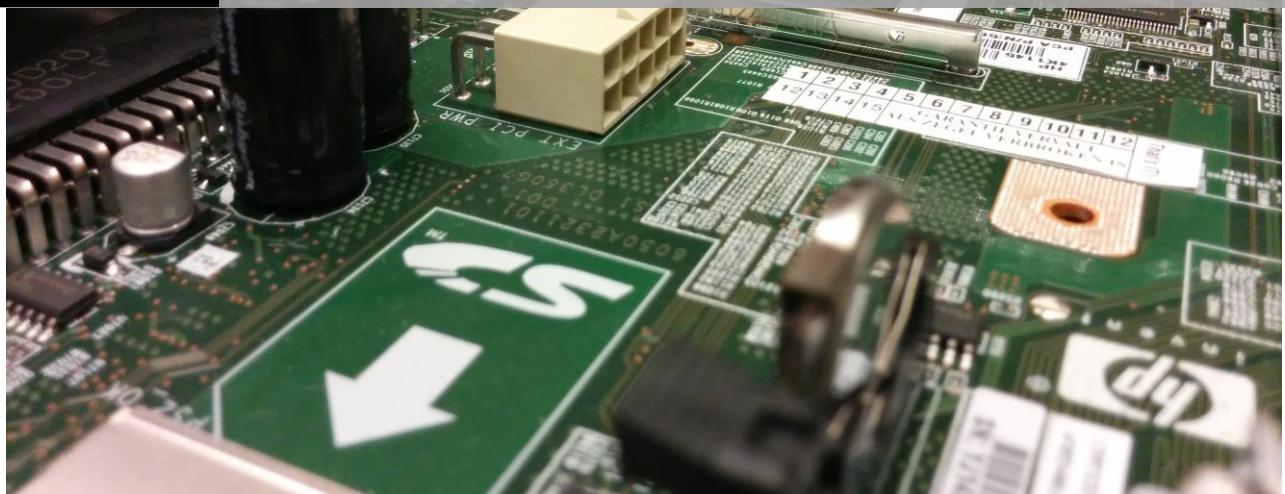


12-6-2014



SURFNET

DNS(SEC) ENERGY CONSUMPTION



Hogeschool van Amsterdam

Afstudeerrapport | Remy Bien

Titel	Afstudeerrapport
Ondertitel	DNS(SEC) Energy consumption
Auteurs naam en voorletter(s)	Bien, R
Studentnummer	500618045
Telefoonnummer	+31 (0)6 40 93 36 19
Plaats	Utrecht
Datum	12-6-2014
Naam onderwijsinstelling	Hogeschool van Amsterdam
Naam studierichting	System and Network engineering
Begeleidende docent	Arnim Eijkhoudt
Bedrijf	SURFnet BV
Adres	Radboudkwartier 273 3511CK Utrecht
Telefoonnummer	+31 (0)3 02 30 53 05
Bedrijfsbegeleider(s) SURFnet	Albert Hankel Roland van Rijswijk
Bedrijfsbegeleider(s) SEFlab	Eric Hoekstra
Stageperiode	03-02-2014 - 27-06-2014 Studiejaar 4

Inhoud

Samenvatting	5
Management Summary.....	6
1 Inleiding	7
2 Achtergrondinformatie	8
2.1 Bedrijfsinformatie	8
2.1.1 SURFnet.....	8
2.1.2 Structuur.....	8
3 Projectdefinitie	9
3.1 Analyse en aanleiding opdracht	9
3.2 Doelstelling.....	9
3.3 Probleemstelling.....	9
3.3.1 Hoofdvraag	9
3.3.2 Deelvragen	9
3.4 Randvoorwaarden.....	9
4 Methodebeschrijving en -verantwoording	10
4.1 Projectmethode	10
4.2 Projectfasering.....	10
4.2.1 Fase 1: Definitiefase	10
4.2.2 Fase 2: Onderzoeksfase	10
4.2.3 Fase 3: Meetfase.....	11
4.2.4 Fase 4: Documentatiefase.....	11
4.2.5 Planning	11
5 Ontwerp en realisatie van het onderzoek.....	13
5.1 Vooronderzoek	13
5.1.1 Unbound	13
5.1.2 BIND.....	13
5.1.3 Microsoft DNS.....	14
5.1.4 Besturingssystemen.....	14
5.1.5 DNS dataset.....	14
5.2 Methodologie.....	15
5.2.1 Pcap replay	15
5.2.2 SEFlab	15
5.2.3 Meetopstelling.....	16
5.2.4 Kalibratie	16
5.2.5 LabVIEW	16

5.2.6	MATLAB	17
5.2.7	Samplefrequentie	17
5.2.8	Downsample tool	17
5.2.9	MATLAB code	17
5.3	Onderzoeksplan.....	18
5.3.1	Centrale onderzoeksvergadering	18
5.3.2	Deelvragen	19
5.4	Meetplan.....	22
5.4.1	Doel metingen.....	22
5.4.2	Aannames	22
5.4.3	Randvoorwaarden	23
5.4.4	Initiële meetprotocollen	23
5.4.5	OS Baseline meting	23
5.4.6	DNS-pakket meting	24
5.4.7	Aanpassing meetprotocollen.....	25
6	Resultaten.....	27
6.1	Baseline verbruik.....	27
6.2	Hoofdvraag	28
6.3	Deelvragen	31
7	Conclusies	36
7.1	Hoofdvraag	36
7.2	Deelvragen	36
8	Aanbevelingen	38
8.1	Toekomstig werk	38
8.1.1	Configuratie	38
8.1.2	IPv6	38
8.1.3	Besturingssystemen	38
8.1.4	Hardware	38
8.1.5	Verplaatsen resolver	38
8.1.6	Unbound op Windows	38
9	Persoonlijke reflectie en ontwikkeling	39
9.1	Competenties	39
9.1.1	S.An3(Software, analyseren)	39
9.1.2	I.Ad3(Infrastructuur, Adviseren)	40
9.1.3	I.Re3 (Infrastructuur, Realiseren)	40
9.1.4	Taakgericht, relatiegericht en zelfgericht	40

9.2	Bedrijfsfeedback.....	42
9.2.1	SURFnet.....	42
9.2.2	SEFlab	42
9.3	Reflectie	42
10	Bronnenlijst	44
11	Bijlagen	45
	Bijlage I: Overzicht PCB's SEFlab	45
	Bijlage II: Kalibratietabel	61
	Bijlage III: MATLAB code	62
	Bijlage IV: Ongekalibreerde vs. gekalibreerde data.....	67
	Bijlage V: Replay script	68
	Bijlage VI: Voorbeeld output replayscript	71
	Bijlage VII: Woordenlijst	74
	Bijlage VIII: Resultaten.....	75
	Bijlage IX: Downsample tool verschil	89
	Bijlage X: MATLAB profiler	91
	Bijlage XI: Analyse gedrag Unbound	92
	Bijlage XII: Formulier eindbeoordeling.....	94

Samenvatting

Green IT is een hot topic. Er worden overal energiebesparende oplossingen ingevoerd en uitgedacht, maar er wordt nog maar weinig gekeken naar efficiënt programmeren. Het energieverbruik van software is minstens net zo belangrijk als dat van de hardware. Daarom is de volgende opdracht bij SURFnet tot stand gekomen: Meet het energieverbruik van drie verschillende DNS resolver pakketten en vergelijk de resultaten met elkaar. De pakketten binnen de scope van dit onderzoek zijn BIND, Unbound en Microsoft DNS.

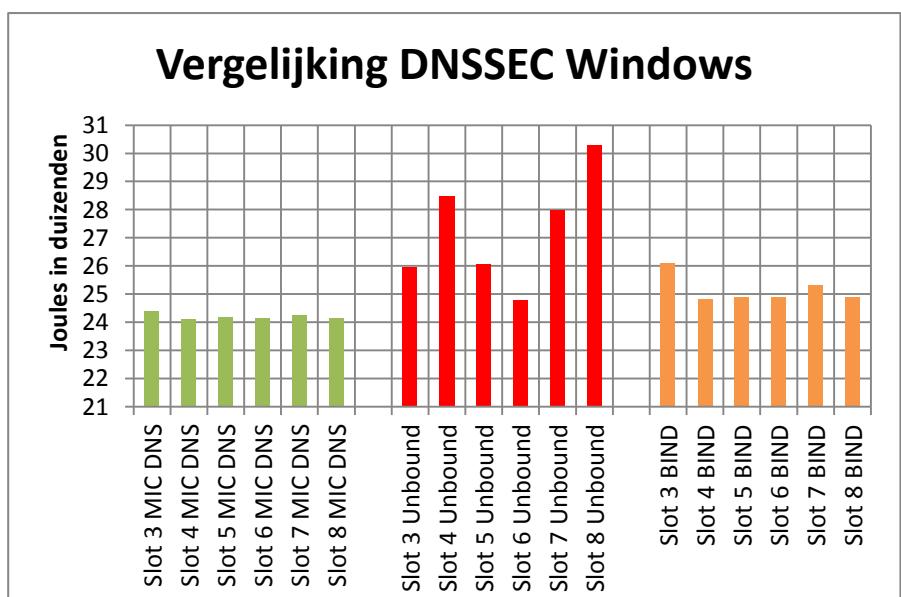
Om te kunnen meten wat software precies verbruikt moet een server uitgerust worden met meetapparatuur. Dat gebeurt bij het Software Energy Footprint lab (SEFlab), gespecialiseerd in het meten van het energieverbruik van software. Bij het SEFlab zijn een aantal servers uitgerust met meetapparatuur die het verbruik op componentniveau kunnen meten. Een van die servers is gebruikt tijdens dit onderzoek. Om een goede representatie van realistische omgevingen na te bootsen is de server geïnstalleerd met Windows Server 2012 R2 en CentOS. Op CentOS zijn BIND en Unbound gemeten en op Windows zijn Microsoft DNS, BIND en Unbound gemeten. Microsoft DNS werkt helaas niet op CentOS. Er is bij SURFnet tien minuten aan DNS queries opgenomen op een van SURFnet's DNS resolvers. Deze dataset aan queries wordt gebruikt om opnieuw af te vuren naar de te meten server.

Er is een meetprotocol opgesteld dat een meting opdeelt in acht slots. In de eerste twee slots gebeurt niks, daar wordt het idle verbruik van het operating system gemeten. In de volgende zes slots wordt er DNS data naar de resolver gestuurd en wordt het energieverbruik tijdens het verwerken van data gemeten. De verwachting is dat in het derde slot het verbruik hoger ligt dan in het vierde t/m het achtste slot omdat dan de cache gevuld wordt. Deze meting wordt op elk pakket twee maal uitgevoerd.

Nadat de metingen uitgevoerd zijn op alle pakketten, is DNSSEC ingeschakeld. Met DNSSEC ingeschakeld zijn alle metingen herhaald. Dit zorgt er voor dat het verbruik met en zonder DNSSEC vergeleken kan worden.

Een van de uitkomsten van de metingen is dat het baselineverbruik van CentOS hoger ligt dan het baselineverbruik van Windows. Het DNS-pakket dat het minste energie verbruikt op Windows is Microsoft DNS. Op CentOS is het pakket dat het minste energie verbruikt BIND. Wordt DNSSEC ingeschakeld dan is op Windows Microsoft DNS nog steeds het efficiëntste programma. Op CentOS is het verbruik van BIND en Unbound met DNSSEC nagenoeg gelijk.

Er is ook een kostenanalyse gemaakt om te kijken hoe veel er bespaard kan worden op jaarbasis door te kiezen voor het efficiëntste programma. De grootste besparing die gehaald kan worden is €2,78 op jaarbasis.



Figuur 1 Voorbeeldgrafiek resultaten

Management Summary

Green IT is a hot topic. Energy saving solutions are being thought out and implemented everywhere. But not a lot of effort is being put into efficient programming. The energy consumption of software is as good as directly responsible for the energy consumption of the hardware. To measure the energy consumption of software, SURFnet came up with an assignment. The assignment is to measure the energy consumption of three different DNS resolvers. The three resolvers that were within the scope of this project are BIND, Unbound and Microsoft DNS.

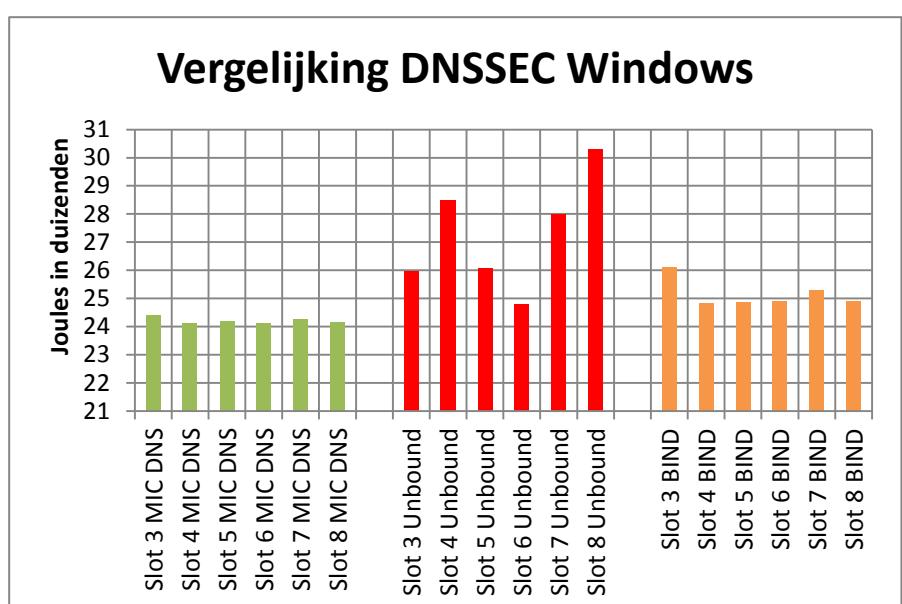
To measure the exact energy consumption of software, a server needs to be equipped with sensors. Exactly that happens at the Software Energy Footprint lab(SEFlab). SEFlab is specialized in measuring energy consumption of software. The lab has several servers that are equipped with measurement sensors which can measure the energy consumption on a component level. One of those servers was used during this assignment. To make sure the measurement environment was a realistic reflection of the real world, multiple operating systems were tested. Windows Server 2012 R2 was tested in combination with Microsoft DNS, BIND and unbound. CentOS was tested in combination with BIND and Unbound. Microsoft DNS unfortunately only works on Windows. A dataset of DNS queries were captured from a live DNS resolver hosted by SURFnet. This dataset was used to replay queries in the test environment.

A measurement protocol was constructed dividing a measurement in 8 slots. In the first two slots nothing happens, that is where idle usage is being measured. In the next 6 slot the DNS dataset is replaying queries to the resolver. The energy consumption during the processing of queries is measured in those slots. The expectation is that the usage of the third slot is higher than slots 4 through 8. The reason for this is the empty cache at the start of the third slot. This measurement was repeated at least once for every resolver.

After measuring each resolver, DNSSEC was enabled. With DNSSEC enabled every measurement was repeated. Results from these measurements made it possible to compare the difference between the resolvers with DNSSEC enabled and disabled.

One of the results was that the baseline usage of CentOS is higher than that of Windows. The most efficient resolver on Windows is Microsoft DNS. The most efficient resolver on CentOS is BIND. When DNSSEC is enabled Microsoft DNS is still the most efficient resolver on Windows. The usage of BIND and Unbound on CentOS is almost equal with DNSSEC enabled.

An analysis of costs was also made to determine how much could be saved by picking the most efficient resolver compared to the least efficient one. De biggest difference in cost is €2.78 on a yearly basis.



Figuur 2 example of results

1 Inleiding

Dit document is onderdeel van de afstudeerstage van Remy Bien. De afstudeerstage is onderdeel van de opleiding informatica aan de Hogeschool van Amsterdam (HvA). De Business Unit die bij de opleiding hoort is System and Network Engineering (SNE). De afstudeerstage loopt van 3 februari 2014 tot en met 27 juni 2014.

In dit document wordt beschreven hoe de eindresultaten zijn verkregen. Er wordt stilgestaan bij de toegepaste methoden en welke beslissingen er zijn genomen tijdens de stageperiode. Tevens worden het meetplan en het onderzoeksplan uitgediept. Het vooronderzoek was de eerste fase van de afstudeeropdracht. In het vooronderzoek wordt de probleemstelling onder de loep genomen, met als resultaat het plan van aanpak, onderzoeksplan en meetplan.

De afstudeeropdracht gaat over het meten van het energieverbruik van DNS Resolver software. Voordat er gemeten kon worden moest er eerst bekendheid opgebouwd worden met de DNS software en het meetsysteem bij het Software Energy Footprint lab (SEFlab).

Green IT is tegenwoordig een hot topic. Datacenters worden ‘groen’, er worden efficiënte servers ingezet en er worden zonnepanelen geplaatst. Er wordt dus een hoop gedaan om het energieverbruik te verminderen. Er wordt echter nog niet veel gekeken naar het daadwerkelijke verbruik van software zelf. Een van de onderdelen van het Internet, het Domain Name System (DNS), wordt door iedereen gebruikt. Het is echter niet bekend hoeveel energie het eigenlijk kost. Het doel van dit onderzoek is om uit te zoeken hoeveel energie de verschillende DNS resolver pakketten verbruiken. En daarnaast wat de security uitbreidings van DNS (DNSSEC) voor invloed heeft op het energieverbruik.

De hoofdstukindeling is logisch opgebouwd, eerst komt de projectdefinitie gevolgd door de methodebeschrijving en -verantwoording. Daarna volgt het ontwerp en de realisatie van het onderzoek gevolgd door de resultaten. Na de resultaten komen nog de conclusie en aanbeveling aan bod, en uiteindelijk ook de persoonlijke reflectie.

2 Achtergrondinformatie

2.1 Bedrijfsinformatie

2.1.1 SURFnet

SURFnet is de NREN (National Research and Education Network) provider van Nederland. De missie van SURFnet is volgens de website¹: “*Het verhogen van de kwaliteit van het hoger onderwijs en onderzoek door het stimuleren, innoveren, ontwikkelen en exploiteren van een geavanceerde, vertrouwde en verbindende ICT-infrastructuur.*”



Figuur 3 Logo SURFnet

De netwerkinfrastructuur van SURFnet geeft de mogelijkheid voor het hoger onderwijs en onderzoek om met behulp van ICT optimaal samen te werken. Het SURFnet netwerk verwerkt per jaar 48.000 terabyte aan inkomend verkeer, en 36.000 terabyte aan uitgaand verkeer. De aangesloten instellingen beschikken over netwerkaansluitingen van 1, 10 of 100 Gbit/s.

Het netwerk van SURFnet maakt gebruik van moderne technologieën als IPv6 en DNSSEC en het verkeer loopt grotendeels over lichtpaden. In 2013 en 2014 wordt het nieuwe netwerk, SURFnet7 uitgerold. SURFnet7 is de nieuwste generatie van het SURFnet-netwerk en gebruikt tienduizend kilometer aan glasvezel om de verbindingen te leggen.

2.1.2 Structuur

Binnen SURFnet werd ik begeleid door Roland van Rijswijk ,Team Lead - Security, Privacy & Trust bij SURFnet en Albert Hankel, Project Manager & Sustainability Ambassador bij SURFnet. Mijn functie binnen SURFnet was junior onderzoeker/stagiair en ik rapporteerde direct aan Roland en Albert. Een deel van mijn stage bracht ik door in het SEFlab in de kelder van de Leeuwenburg in Amsterdam. De technische begeleiding in het SEFlab werd verzorgd door Eric Hoekstra.

Op de momenten dat ik bij SURFnet zat probeerde Roland, Albert en ik wekelijks voortgangsgesprekken te voeren. Tijdens mijn tijd in het SEFlab had ik regelmatig contact met Roland via Google Hangouts. Daarnaast was er regelmatig mailcontact tussen mij, Roland en Albert om elkaar op de hoogte te houden.

¹ <http://www.surfnet.nl/>

3 Projectdefinitie

In dit hoofdstuk komt de analyse en aanleiding van de opdracht aan bod. Ook worden de doel- en probleemstelling behandeld, en wordt er naar de hoofd- en deelvragen gekeken.

3.1 Analyse en aanleiding opdracht

Het Domain Name System (DNS) is een vitaal onderdeel van het internet. DNS maakt de vertaalslag van domeinnamen naar IP-adressen, en zorgt er dus voor dat computers er achter kunnen komen welke domeinnaam bij welk IP-adres hoort. Om dit mogelijk te maken zijn er DNS servers nodig, ook wel resolvers genoemd. Die servers draaien een gewoon besturingssysteem met daarop een DNS softwarepakket geïnstalleerd. Het is op dit moment nog onbekend hoeveel energie de software pakketten verbruiken. Het is de bedoeling dat daar na dit onderzoek meer over bekend is. De drie pakketten waarvan het verbruik gemeten wordt zijn BIND, Unbound en Microsoft DNS. Volgens Roland van Rijswijk beslaan deze drie DNS resolver pakketten het grootste gedeelte van de markt.

Om tot dat doel te komen worden de drie DNS pakketten vergeleken bij het Software Energy Footprint lab (SEFlab). In het SEFlab zijn servers uitgerust met meetapparatuur die het energieverbruik van verschillende componenten meten. Een van die servers is ter beschikking gesteld voor uitvoeren van de metingen.

3.2 Doelstelling

De doelstelling van dit onderzoek is om het energieverbruik van drie verschillende DNS pakketten te meten en vergelijken.

3.3 Probleemstelling

In deze paragraaf wordt de probleemstelling beschreven, de probleemstelling bestaat uit de hoofdvraag, de deelvragen en een definitie van gebruikte termen.

3.3.1 Hoofdvraag

Welk van de drie gemeten DNS software pakketten gaat het efficiëntste om met energie?

3.3.2 Deelvragen

De onderstaande deelvragen zullen beantwoord worden om een volledig antwoord op de bovenstaande onderzoeksfrage te geven. De context van de vragen is te vinden in hoofdstuk 5.3.

1. Wat zou er bespaard kunnen worden door gebruik van één van de te onderzoeken pakketten in vergelijking met de andere pakketten?
2. Wat is de invloed van het gebruiken van DNSSEC op het energieverbruik van de verschillende pakketten?
3. Kan het aanpassen van de configuratie van een of meer van de pakketten leiden tot een efficiënter gebruik van energie?
4. Is het efficiënter om de resolver functionaliteit te verplaatsen naar een hoger niveau in het netwerk?

3.4 Randvoorwaarden

De metingen vinden plaats bij het SEFlab, daar is een server ter beschikking gesteld die is uitgerust met meetapparatuur die het energieverbruik van verschillende hardwarecomponenten kan meten.

4 Methodebeschrijving en -verantwoording

Dit hoofdstuk beschrijft de algehele aanpak van de afstudeeropdracht. In dit hoofdstuk zullen onder andere de projectmethode en projectfasering beschreven worden. Daarnaast zullen ook de producten per fase gedefinieerd worden.

4.1 Projectmethode

Het project is verdeeld in verschillende fases. Deze fases werden van te voren eerst gepland, daarna werden de geplande activiteiten uitgevoerd. Na het uitvoeren van de activiteiten volgde een kwaliteitscheck, dit werd door zowel de stagiair als de bedrijfsbegeleider gedaan. Daarna werd indien nodig de feedback verwerkt. Dit proces volgt in grote lijnen de 'Deming cycle'², oftewel plan-do-check-act. Door dit proces te doorlopen wordt de kwaliteit van het eindresultaat gewaarborgd doormiddel van de checks door de bedrijfsbegeleider en de stagiair.

Er is voor deze projectmethode gekozen omdat het een 'lichtgewicht' methode is. Het gaat hier om een project uitgevoerd door 1 persoon. Een projectmethode als bijvoorbeeld PRINCE2 is geschikt voor complexe projecten door bijvoorbeeld het project initiation document. Ook is er niet gekozen voor de kleine elementen uit PRINCE2, omdat de fasering en documentatie uit de huidige methode die punten ook dekt. De fasering zorgt er voor dat het project in logische stukken wordt opgedeeld. Doormiddel van het plan-do-check-acht traject wordt de kwaliteit gewaarborgd.

4.2 Projectfasering

4.2.1 Fase 1: Definitiefase

De definitiefase was de eerste fase van het project. Tijdens deze fase is er een beknopt plan van aanpak geschreven met de opdrachtbeschrijving, initiële deelproducten en de initiële planning. Het plan van aanpak gold als 'kapstok' voor de rest van het project, en is in grote lijnen gevuld als leidraad.

Tijdens deze fase vond ook al een deel van het vooronderzoek plaats. Er werd ervaring opgedaan met de verschillende DNS pakketten en hun mogelijkheden. Er is gekeken naar verschillende tools om te gebruiken tijdens de metingen, waarvan de beschrijvingen zijn opgenomen in de documentatie. Ook het initiële meetplan en onderzoeksplan zijn opgesteld tijdens deze fase, waarvan de afronding plaatsvond tijdens de onderzoeksfase.

Deelproducten definitiefase:

- Beknopt plan van aanpak;
- Initiële planning;
- Initieel meetplan;
- Initieel onderzoeksplan;
- Concept scriptie.

4.2.2 Fase 2: Onderzoeksfase

De onderzoeksfase was de tweede fase van het project. Tijdens deze fase zijn de initiele documenten uit de eerste fase uitgewerkt aan de hand van het plan van aanpak.

² Bron [1] Bulsuk, Karn (2 Februari, 2009). Taking the First Step with the PDCA (Plan-Do-Check-Act) Cycle. <http://www.bulsuk.com/2009/02/taking-first-step-with-pdca.html>

Gedurende de onderzoeksfase is de testomgeving opgebouwd, bestaande uit een HP server, vijf harde schijven met verschillende DNS configuraties, een replayscript om DNS data af te spelen en MATLAB-code om de ruwe meetdata te verwerken. De MATLAB-code is in samenwerking met het SEFlab geschreven. De code is geschikt gemaakt voor de eisen van dit project door bijvoorbeeld support voor slots toe te voegen.

Er zijn veel testmetingen uitgevoerd tijdens de onderzoeksfase om ervaring met het meetsysteem op te bouwen. Tijdens deze testmetingen zijn er een aantal verbeteringen voorgesteld en doorgevoerd in het meetsysteem.

Deelproducten onderzoeksfase:

- Definitieve planning;
- Afgerond meetplan;
- Afgerond onderzoeksplan;
- Initieel onderzoeksrapport;
- Concept scriptie;
- Werkende testomgeving, bestaande uit de server, verschillende harde schijven met de verschillende DNS configuraties, een replayscript om DNS data af te spelen en MATLAB code om meetdata te verwerken.

4.2.3 Fase 3: Meetfase

De onderzoeksfase was de derde fase van het project. Tijdens deze fase zijn de daadwerkelijke metingen uitgevoerd en verwerkt in MATLAB.

Deelproducten meetfase:

- Ruwe meetdata;
- Meetdata verwerkt door MATLAB-code.

4.2.4 Fase 4: Documentatiefase

De documentatiefase was de vierde en laatste fase van het project. Tijdens deze fase is de meetdata verwerkt en de vergelijkingen tussen de verschillende DNS pakketten gemaakt. De resultaten daarvan zijn verwerkt in het onderzoeksrapport en scriptie. Daarnaast zijn er conclusies gemaakt en beargumenteerd in de documentatie.

Deelproducten documentatiefase:

- Afgerond onderzoeksrapport;
- Scriptie.

4.2.5 Planning

De planning heeft een totale doorlooptijd van 106 dagen, waarvan 6 dagen uit mogen vallen. In de tabel 1 op de volgende pagina is de planning per fase te zien.

Taaknaam	Effort	Begindatum	Einddatum
DNS(SEC) Energy consumption	106 dagen	maa 3-2-14	maa 30-6-14
Fase 1	30 dagen	maa 3-2-14	vri 14-3-14
Plan van aanpak	10 dagen	maa 3-2-14	vri 14-2-14
Vooronderzoek	30 dagen	maa 3-2-14	vri 14-3-14
Onderzoeksplan	25 dagen	maa 10-2-14	vri 14-3-14
Presentatie onderzoeksplan	0 dagen	din 18-2-14	din 18-2-14
Meetplan	25 dagen	maa 10-2-14	vri 14-3-14
Metingen en onderzoek	5 dagen	maa 10-3-14	vri 14-3-14
Fase 2	30 dagen	maa 17-3-14	vri 25-4-14
Eerste concept afstudeerrapport	0 dagen	maa 17-3-14	maa 17-3-14
Proefmetingen en onderzoek	35 dagen	maa 10-3-14	vri 18-4-14
Fase 3	10 dagen	maa 21-4-14	Vri 2-5-14
Metingen	10 dagen	maa 21-4-14	Vri 2-5-14
Go/No-go moment	0 dagen	maa 28-4-14	maa 28-4-14
Fase 4	46 dagen	maa 5-5-14	maa 30-6-14
Onderzoeksrapport	50 dagen	maa 17-3-14	vri 23-5-14
Afstudeerrapport	86 dagen	maa 3-3-14	maa 30-6-14
Afstudeersessie	0 dagen	maa 16-6-14	maa 16-6-14

Tabel 1 Planning

5 Ontwerp en realisatie van het onderzoek

5.1 Vooronderzoek

Dit hoofdstuk geeft een beknopte beschrijving van een aantal onderdelen die aan bod zijn gekomen tijdens het vooronderzoek.

5.1.1 Unbound

Unbound is een validerende, recursive en caching DNS resolver. Unbound wordt onwikkeld en onderhouden door NLnet Labs. Unbound is vanaf mei 2008 beschikbaar onder BSD licentie(open source). Unbound is beschikbaar voor zowel Linux als Windows.

Configuratie

De configuratie van Unbound is relatief simpel, onderstaande configuratie is gebruikt tijdens de metingen:

```
server:  
    interface: 192.168.10.11  
    access-control: 192.168.10.0/24 allow  
    do-ip6: no  
    root-hints: /etc/unbound/root.hints
```

Interface definieert de inkomende interface voor queries. *Access-control* definieert vanuit welk subnet queries toegestaan worden, alle andere queries worden geweigerd. *Do-ip6* geeft aan of IPv6 queries beantwoord worden, omdat er in het SEFlab geen IPv6 beschikbaar is schakelen we IPv6 support uit.³ *Root-hints* verwijst naar de locatie van de root hints file.

5.1.2 BIND

BIND is open-source DNS software geschreven als referentie implementatie van het DNS protocol. BIND is te gebruiken als high-volume applicatie, en is het meest gebruikte DNS pakket op het internet.⁴ De afkorting BIND staat voor ‘Berkeley Internet Name Domain’, BIND is rond 1980 ontwikkeld aan de universiteit van Californië in Berkeley. BIND is beschikbaar voor zowel Linux als Windows.

Configuratie

Onderstaande configuratie is gebruikt tijdens de metingen:

```
options {  
    // tell named where to find files mentioned below  
    directory "C:\named\etc";  
    // on a multi-homed host, you might want to tell named  
    // to listen for queries only on certain interfaces  
    listen-on { 127.0.0.1; 192.168.10.0/24; };  
    recursion yes;  
    allow-recursion { 192.168.10.0/24; };
```

³ Zie ‘Toekomstig werk’

⁴ Bron [2] Internet Systems Consortium (datum onbekend). BIND, The most widely used Name Server Software. <https://www.isc.org/downloads/bind/>
Bron [3] Moore, Don (23 Mei, 2004). DNS Server Survey. <http://myDNS.bboy.net./survey/>

```

};

// The single dot (.) is the root of all DNS namespace, so
// this zone tells named where to start looking for any
// name on the Internet
zone "." IN {
    // a hint type means that we've got to look elsewhere
    // for authoritative information
    type hint;
    file "named.root";

```

5.1.3 Microsoft DNS

Microsoft DNS is de DNS service implementatie van Windows. Microsoft DNS bestaat uit client en server software, alleen de server software is interessant voor dit onderzoek. Microsoft DNS werd voor het eerst meegeleverd met Windows NT server 3.51 (1995). Microsoft DNS is alleen beschikbaar op Windows.

5.1.4 Besturingssystemen

BIND en Unbound werken beide zowel op Linux als op Windows, maar Microsoft DNS niet. Om te zorgen voor een constante baseline, is er besloten om BIND, Unbound en Microsoft DNS alle drie op Windows server 2012 R2 te meten. Het energieverbruik van BIND en Unbound op Windows kan echter ook verschillen van het verbruik op Linux, daarom is het ook interessant om deze pakketten op Linux te meten. De gekozen Linux distributie is CentOS. CentOS is een Red Hat kloon, en Red Hat wordt gebruikt op de systemen van SURFnet.

5.1.5 DNS dataset

De DNS dataset die gebruikt wordt om queries naar het test systeem te sturen, is een set aan queries die verzameld is van een live DNS systeem in beheer van SURFnet. Het wordt opgeslagen als een PCAP-file, die de ruwe queries en antwoorden bevat. Voordat deze dataset bruikbaar is, worden eerst de queries van de antwoorden gescheiden in Wireshark en daarna het source/destination MAC-adres en IP-adres gewijzigd. Het gebruik van echte DNS-data zorgt er voor dat het gedrag van de DNS software zo realistisch mogelijk is. Tijdens testmetingen is een dataset van ongeveer 9,5 minuut gebruikt en werden er geen vreemde afwijkingen of stroombijtjes waargenomen. Het stroomverbruik was constant gedurende de hele meting. Tijdens de echte metingen is gebruik gemaakt van 10 minuten en 20 seconden. Het is geen dataset van exact 10 minuten omdat er wat handwerk vereist is bij het opnemen van de data, wat er voor zorgt dat de dataset iets langer wordt. De dataset bestaat uit 1240548 queries.

5.2 Methodologie

5.2.1 Pcap replay

Om de DNS dataset af te spelen is gebruik gemaakt van de tool Tcpreplay. Tcpreplay⁵ is een verzameling open source tools voor UNIX met de mogelijkheid om vooraf opgenomen netwerkverkeer (in libpcap formaat) aan te passen en af te spelen. De toolset omvat de volgende tools: tcpprep, tcprewrite, tcpreplay, tcpliveplay, tcpreplay-edit, tcpbridge, tcpcapinfo. De exacte beschrijving voor al deze tools kan worden gevonden op de Tcpreplay website.

Tcpreplay zal gebruikt worden om de DNS datasets af te spelen tegen de DNS servers. Ook wordt Tcprewrite gebruikt om de source/destination IP- en MAC-adressen van de DNS datasets aan te passen. Op die manier is het mogelijk om live opgenomen verkeer opnieuw af te spelen in de gecontroleerde meetomgeving. Het afspeLEN van de DNS data moet consistent gebeuren bij de verschillende metingen. Om de consistentie te waarborgen wordt het afspeLEN uitgevoerd door een script. De stappen die het script doorloopt zijn te vinden in de bijlage.

Een testdataset is een aantal keer afgevuurd op een DNS server en de performance van Tcpreplay is daarbij gemeten. De resultaten daarvan zijn te vinden in tabel 2. De gemiddelde waardes zijn afgerond tot maximaal 2 punten achter de komma, en de maximale afwijking is berekend in verhouding met het gemiddelde. De afwijking tussen de tests is zo klein namelijk 0,012% dat het geen meetbare invloed heeft gehad op de testresultaten. De afwijking van 0.01 seconde zou gelijk staan aan een halve datasample. Een halve datasample kan niet gemeten worden met de huidige opstelling. Een gemeten dataset bestaat uit ongeveer 31.000 samples, afhankelijk van de exacte looptijd.

	Test run 1	Test run 2	Test run 3	Average	Max. deviation
Number of packets	1003135	1003135	1003135	1003135	0
Time in seconds	564.08	564.08	564.09	564.08	0.01
Bytes per second	152806.7	152800.8	152776.6	152794.7	18.1
Megabit per second	1.22	1.22	1.22	1.22	0
Packet per second	1775.92	1775.85	1775.57	1775.78	0.21

Tabel 2 tcpreplay onderzoek

5.2.2 SEFlab

Het Software Energy Footprint Lab is een onderzoeks lab dat is opgericht om de impact van software op het energieverbruik van hardware te testen. Het SEFlab is een samenwerking tussen de Software improvement Group (SiG) en het CleanTech onderzoeksprogramma van de HvA. Het SEFlab heeft een server ter beschikking gesteld voor gebruik tijdens het DNSSEC onderzoek. Deze server is uitgerust met meetbordjes die het energieverbruik van verschillende componenten kunnen meten. In bijlage 1 is de beschrijving van de meetbordjes te vinden.



Figuur 4 Meetopstelling

⁵ Bron [4] AppNeta, (December, 2013). Tcpreplay, Pcap editing & replay tools voor *NIX.
<http://tcpreplay.synfin.net/>

5.2.3 Meetopstelling

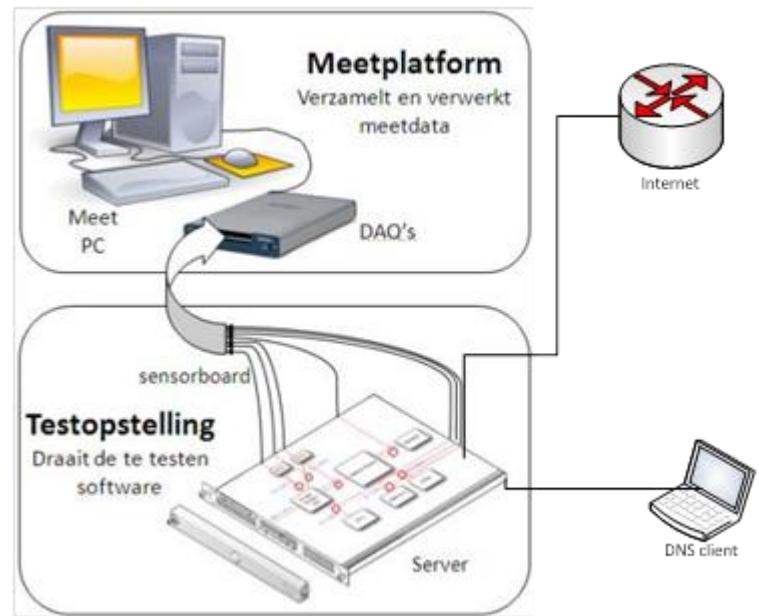
Op figuur 4 op de vorige pagina is de meetopstelling te zien. De schematische weergave is te zien in figuur 5. De meetopstelling bestaat uit de HP DL360 G7, het Data Aquisition Device (DAQ), de meetbordjes in de server en de computer waar de meetdata wordt opgeslagen. Tijdens het testen is de G7 afgedekt met de kap om een goede airflow te garanderen.

5.2.4 Kalibratie

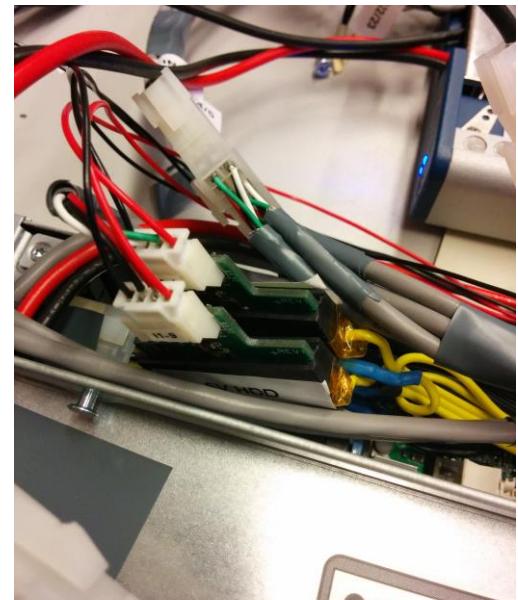
De meetbordjes, te zien in figuur 6, waren tijdens de proefmetingen nog niet gekalibreerd, dat hield in dat het stroomverbruik dat werd aangegeven kon afwijken van het daadwerkelijke verbruik. De toepassing van de kalibratie vind echter pas plaats nadat de metingen al zijn genomen, in de verwerking met de MATLAB-code. De MATLAB-code wordt behandeld in paragraaf 5.2.9. Dat houdt in dat de metingen die zijn genomen voordat er is gekalibreerd gewoon bruikbaar zijn. Ze moesten alleen opnieuw ingelezen worden met de gekalibreerde waardes in MATLAB. De kalibratie leverde maximaal 5 procent verschil op. Tijdens de kalibratie is een verkeerd ingestelde weerstand ook verbeterd, met als gevolg meer nauwkeurigheid. Zie bijlage IV voor de verschillen in gekalibreerde en ongekalibreerde data. De kalibratietabel is te vinden in bijlage II.

5.2.5 LabVIEW

LabVIEW is een afkorting voor Laboratory Virtual Instrument Engineering Workbench. Het is een system-ontwerp en -ontwikkel platform voor de visuele programmeertaal van National Instruments. De tool is tijdens dit project gebruikt om de DAQ uit te lezen. Om dit doel te bereiken heeft het SEFlab een LabVIEW applicatie gemaakt, specifiek voor G7 server. Deze applicatie was al in gebruik bij het SEFlab voordat het project was begonnen. In figuur 7 is de LabVIEW applicatie te zien tijdens het testen.



Figuur 5 Meetopstelling (schematisch)



Figuur 6 Meetbordjes



Figuur 7 Screenshot LABview

5.2.6 MATLAB

MATLAB is een zeer uitgebreid en interactief softwarepakket voor wiskundige berekeningen en visualisaties. MATLAB kan gebruikt worden voor het analyseren van data, het ontwikkelen van algoritmes, en het bouwen van modellen en applicaties. Tijdens dit project is MATLAB gebruikt als tool om de meet data in te lezen en te verwerken. Om de data te verwerken is er code geschreven, de beschrijving van deze code is te vinden in de paragraaf 5.2.9.

5.2.7 Samplefrequentie

LabVIEW leest de DAQ uit met een bepaalde samplefrequentie. Die frequentie is aan te passen tussen de 50.000 hertz en 1 hertz. Hoe hoger de samplefrequentie, hoe groter de datafile uiteindelijk wordt. Tijdens het vooronderzoek is gebleken dat datafiles met een grote samplefrequentie moeilijk te verwerken zijn, daarom is er in overleg met het SEFlab besloten de samplefrequentie in te stellen op 50 hertz. Bij een samplefrequentie van vijftig hertz duurt het verwerken van een dataset van 2 uur tussen de 4 en 6 uur. Dit is te zien in de MATLAB cpu profiler export, te zien in bijlage X. Als de samplefrequentie wordt verlaagd, is het niet zo dat er data mist in de uiteindelijke set. De data wordt gemiddeld doorgegeven. Als er gekozen wordt voor een sampelrate van 1Hz, is die ene hertz het gemiddelde van 50.000 sampels.

5.2.8 Downsample tool

Tijdens het meten en verwerken van de data bleek het lastig te zijn de resultaten snel inzichtelijk te maken vanwege een minimale verwerkijd van 5 uur per dataset. Om dit probleem op te lossen is er een downsample tool geschreven door Roland van Rijswijk. Deze downsampletool veranderd de samplefrequentie van de datafile. Bij het uitvoeren van de tool wordt aangegeven door welk getal de huidige samples gedeeld moeten worden, de tool rekent dan het gemiddelde uit van die samples, en maakt een nieuw bestand aan met de nieuwe samplefrequentie.

Bijvoorbeeld: van 50 hertz naar 1 hertz. De tool rekent het gemiddelde uit van 50 samples, en schrijft dat weg als 1 sample. Bijkomend voordeel is dat de datafiles die door de downsampletool zijn verwerkt een ‘mooiere’ grafiek geven. Dit zorgt voor een duidelijkere weergave van bijvoorbeeld het gedrag tijdens het vullen van de cache. Het downsamplen van een dataset introduceert een verschil van maximaal 0,16% als gevolg van de beperkte nauwkeurigheid van floating point.⁶ Het effect van het downsamplen is te zien in bijlage IX.

5.2.9 MATLAB code

De MATLAB code is modular opgezet, dat houdt in dat er verschillende files zijn met allemaal hun eigen functie, en een main file die al die files aan elkaar knoopt. De code zelf is te vinden in bijlage III. In deze paragraaf worden de functies van de files kort doorgenomen.

Main.m

Main.m is de main file van de code. In de main file worden de andere files aangeroepen en een aantal berekeningen uitgevoerd. De belangrijkste functie van de main.m file is de slot_data_joules berekening. Die functie berekent het verbruik in joule tijdens de vooraf aangegeven slots. De slots worden aangegeven in de measurements.m file.

⁶ Bron [5] Goldberg, David (Maart 1991). What Every Computer Scientist Should Know About Floating-Point Arithmetic. http://docs.oracle.com/cd/E19422-01/819-3693/neg_goldberg.html

Measurement.m

De measurement file is het stuk van de MATLAB code dat veranderd voor elke meting. Hierin wordt aangegeven welke kalibratietabel er moet worden gebruikt, hoeveel files er moeten worden ingelezen, hoeveel slots de files bevatten, en met welke samplefrequentie er is gemeten. Daarnaast worden er in measurement.m ook de slots gedefinieerd. Dit is belangrijk, omdat het stroomverbruik uiteindelijk via de slots berekend wordt.

Plotten.m

De plotten.m file maakt een plot van de uiteindelijke resultaten in een grafiek in MATLAB. Plotten.m kan gebruikt worden om snel resultaten in te zien, maar is niet gebruikt tijdens dit project. Plotten.m genereert plots met het energieverbruik in joule per slot.

Read.m

Read.m leest de data die gegenereerd is door labview in. Na het inlezen van de data wordt de data omgezet naar een door MATLAB bruikbaar formaat. Ook worden er een aantal variabelen opgeslagen die later voor berekeningen nodig zijn, zoals het aantal samples in de file. Het uitvoeren van read.m op een grote meetfile kan zeer lang duren, daarom is de downsample tool geschreven om het verwerken van de data te versnellen.

Init_servers.m

De init_servers.m file bevat de kalibratietabellen van de servers. De twee gebruikte kalibratietabellen zijn G7_1, de ongekalibreerde data en G7_1_cal, de gekalibreerde data. In bijlage 4 is de zien dat de verschillen tussen deze twee tabellen klein zijn, maar groot genoeg om kalibratie te verantwoorden.

Get_slot.m

Get_slot.m gebruikt de slotgegevens die zijn opgegeven in measurement.m en zoekt daarbij de juiste samples op. Die worden vervolgens weer in een variabele opgeslagen. Dit zorgt er voor dat er in measurement.m gewoon een datum en tijd opgegeven kunnen worden, en dat MATLAB daar zelf de juiste samples bij zoekt.

5.3 Onderzoeksplan

5.3.1 Centrale onderzoeksraag

“Welk van de drie te onderzoeken softwarepakketten; BIND, Unbound of Microsoft DNS gaat het efficiëntste om met energie?”

Doelstelling: De doelstelling van de centrale onderzoeksraag is om er achter te komen welk van de pakketten het efficiëntst omgaat met energie.

Toelichting: Om er achter te komen welk pakket het efficiëntste is, is er een metric bepaald. Alle pakketten hebben dezelfde dataset te verwerken gekregen, en het pakket dat na afloop het minste energie verbruikt heeft is het efficiëntste pakket. Een kanttekening hierbij is dat de verwerkingsijd van de dataset gelijk moet zijn voor alle pakketten.

Dataverzameling:	De dataverzameling vindt plaats bij het SEFlab. De gebruikte testopstelling is te vinden in hoofdstuk 6. Er heeft een laptop DNS queries verstuurd naar de server, die uitgerust is met energie sensoren. Deze sensoren zitten aangesloten op het DAQ device. Het DAQ device wordt uitgelezen door een meetcomputer.
Data analyse:	De statistische gegevens die voortkomen uit de meting werden verwerkt en geanalyseerd. Tijdens verwerking is de data ‘vertaald’ naar de vooraf gedefinieerde metric, zodat een analyse mogelijk was. De uitkomst van de tests van de verschillende pakketten zijn met elkaar vergeleken.
Meting:	Om de efficiëntie tussen de drie pakketten te kunnen vergelijken zijn verbruiksgegevens nodig voor elk pakket. Ook is het belangrijk om te weten hoeveel van het totale verbruik aan het besturingssysteem toe te kennen valt. Dit betekent dat er een aantal metingen moeten worden uitgevoerd; ten eerste een baseline meting voor het besturingssysteem. Uit de resultaten van deze meting wordt vastgesteld hoeveel energie het besturingssysteem verbruikt. Deze meting is uitgevoerd voor zowel Windows als voor Linux. Ten tweede moeten de verschillende pakketten getest worden op elk besturingssysteem. Microsoft DNS is alleen te testen op Windows. Om de betrouwbaarheid van de resultaten van de metingen te garanderen zijn de metingen een aantal keer uitgevoerd.

5.3.2 Deelvragen

5.3.2.1 Deelvraag 1

“Wat zou er bespaard kunnen worden door gebruik van één van de pakketten in vergelijking met de andere pakketten?”

Doelstelling:	Het doel van green-IT is het efficiënt om gaan met energie. Aan de hand van deze deelvraag leert men hoeveel er bespaard zou kunnen door het gebruik van het efficiëntste pakket.
Toelichting:	Het is voor bedrijven natuurlijk interessant om te weten wat ze kunnen besparen door het gebruik van een bepaald DNS pakket. Om er achter te komen hoe veel er bespaard kan worden is het belangrijk om eerst te weten hoeveel energie er verbruikt wordt. De data voor deze deelvraag is al verzameld voor het beantwoorden van de hoofdvraag. Verder is het ook belangrijk om een kostenbepaling te maken voor energie. Want niet alleen de energiebesparing is belangrijk, maar ook de kostenbesparing die dat met zich meebrengt.
Dataverzameling:	De dataverzameling hier is gelijk aan de dataverzameling van de vorige deelvraag.
Data analyse:	De statistische gegevens die voortkomen uit de meting zijn verwerkt en geanalyseerd. Tijdens verwerking is de data ‘vertaald’ naar de vooraf gedefinieerde metric, zodat een analyse mogelijk is. De uitkomst van de tests van de verschillende pakketten zijn met elkaar vergeleken. Daarna is een grafische weergaven van de kosten van het gebruik van elk pakket gemaakt.

Meting: De benodigde energiemetingen voor het beantwoorden van deze deelvraag zijn al aanwezig door het beantwoorden van de vorige vraag. De resultaten kunnen geschaald worden naar bijvoorbeeld een productiesysteem, meerdere productiesystemen of zelfs landelijk. Daarna kan de besparing in energieverbruik worden omgerekend in een financiële besparing. Aangenomen is dat de voorgaande metingen betrouwbaar zijn, en de resultaten niet veranderd zijn. Er zijn geen nieuwe metingen gedaan voor het beantwoorden van deze deelvraag.

5.3.2.2 Deelvraag 2

"Wat is de invloed van het gebruiken van DNSSEC op het energieverbruik van de verschillende pakketten?"

Doelstelling: De doelstelling van deze onderzoeksvervraag is om er achter te komen wat de invloed van DNSSEC op het energieverbruik is.

Toelichting: DNSSEC is een security uitbreiding voor DNS met een cryptografie onderdeel. Voor cryptografie is in theorie veel CPU rekenkracht nodig. Daarom zou het een grote invloed kunnen hebben op het energieverbruik van een DNS-pakket. De efficiëntietests die zijn uitgevoerd voor het beantwoorden van de centrale onderzoeksvervraag worden hier nogmaals uitgevoerd, maar nu met DNSSEC ingeschakeld.

Dataverzameling: De dataverzameling hier is gelijk aan de dataverzameling van de vorige deelvragen.

Data analyse: De statistische gegevens die zijn voortgekomen uit de meting zijn verwerkt en geanalyseerd. Tijdens verwerking werd de data ‘vertaald’ naar de vooraf gedefinieerde metric, zodat een analyse mogelijk is. De uitkomst van de tests van de verschillende pakketten zijn met elkaar vergeleken. Ook zijn de uitkomsten van deze tests vergeleken met de uitkomsten van de tests voor de centrale onderzoeksvervraag. Zo ontstaat er een beeld van het verschil in energieverbruik bij het al dan niet gebruik maken van DNSSEC.

Meting: Ook voor het DNSSEC vraagstuk zijn verbruiksgegevens nodig. Zowel van de pakketten in combinatie met DNSSEC, als het verbruik van de pakketten zonder DNSSEC. Ook zijn de baseline metingen voor de operating systems nodig. Aangenomen is dat er aan de baseline van de operating systems niets is veranderd, ook is er aangenomen dat het verbruik van de pakketten zonder DNSSEC niet veranderd is ten opzichte van de eerste metingen. Die metingen zijn dus niet opnieuw uitgevoerd.

5.3.2.3 Deelvraag 3

"Kan het aanpassen van de configuratie van een of meer van de pakketten leiden tot een efficiënter gebruik van energie?"

Doelstelling: De doelstelling van deze deelvraag is uitzoeken of het mogelijk is om een pakket nog efficiënter te maken door het aanpassen van de configuratie.

Toelichting:	Voor de andere deelvragen is een standaard configuratie gebruikt. Bij deze deelvraag wordt bepaald of de configuratieverandering invloed heeft op het energieverbruik van het softwarepakket. Het aanpassen van de config heeft waarschijnlijk ook invloed op de performance die ook interessant is voor de resultaten maar niet meegenomen zal worden in de efficiëntie berekening. Het is voor elk pakket verschillend wat er aangepast kan worden in de config, daarom zijn de exacte aanpassingen tijdens de tests zelf gedefinieerd. Ook is de scope van deze deelvraag heel erg afhankelijk van de beschikbare tijd, dit zal leidend zijn bij de keuze om de keuze op welke pakketten de deelvraag wordt toegepast.
Dataverzameling:	De dataverzameling hier is gelijk aan de dataverzameling van de vorige deelvragen.
Data analyse:	De statistische gegevens die voortgekomen zijn uit de meting zijn verwerkt en geanalyseerd. Tijdens verwerking is de data ‘vertaald’ naar de vooraf gedefinieerde metric, zodat een analyse mogelijk is. De uitkomst van de tests van de verschillende configuraties voor de pakketten zijn met elkaar vergeleken.
Meting:	Om de deelvraag te kunnen beantwoorden hebben we de volgende gegevens nodig; Het baseline verbruik van de operating systems. Het verbruik van de DNS pakketten met de standaard config, en het verbruik van de DNS pakketten met de aangepaste config. Er zijn een aantal nieuwe metingen uitgevoerd, namelijk metingen van het verbruik van de DNS pakketten in combinatie met de nieuwe configuratie. Aangenomen is dat er aan de baseline van de besturingssystemen niets is veranderd, ook is er aangenomen dat het verbruik van de pakketten met standaard config niet veranderd is ten opzichte van de eerste meting. Die metingen zijn dus niet opnieuw uitgevoerd.

5.3.2.4 Deelvraag 4

“Is het efficiënter om de resolver functionaliteit te verplaatsen naar een hoger niveau in het netwerk?”

Doelstelling:	De doelstelling van deze deelvraag is om er achter te komen of het aggregeren van DNS functionaliteit op een hoger niveau in het netwerk efficiënter is.
Toelichting:	Elke grote organisatie of educatieve instelling heeft zijn eigen DNS resolvers met allemaal hun eigen energieverbruik. Als al deze instellingen gebruik zouden maken van dezelfde resolver van bijvoorbeeld hun internetprovider zouden ze ze hun eigen resolvers kunnen uitfaseren. De vraag is of dit ook echt efficiënter zal zijn. Er spelen veel factoren mee bij deze onderzoeksvraag die slecht tot niet meetbaar zijn. Denk bijvoorbeeld aan het volgende: als een organisatie zijn resolver uitfasert en er voor kiest gebruik te gaan maken van de resolver van zijn provider gaat de latency in theorie omhoog. Als de organisatie duizend apparaten heeft die gebruik maken van deze DNS resolver zullen zij allemaal een aantal milliseconde langer moeten wachten op hun antwoord. Uiteindelijk

gebruikt de resolver misschien minder energie, maar er wordt ook energie verbruikt door alle clients die die extra milliseconden moeten wachten. Het verbruik van deze clients is moeilijk meetbaar, maar er is met onderzoek wel een hypothese aan te koppelen.

Dataverzameling: De dataverzameling hier is grotendeels gelijk aan de dataverzameling van de vorige deelvragen. Er is mogelijk echter ook een literatuuronderzoek nodig. Tevens is het beantwoorden van deze deelvraag afhankelijk van de beschikbare tijd.

Data analyse: De statistische gegevens die voortkomen uit de meting zullen worden verwerkt en geanalyseerd. Tijdens verwerking wordt de data ‘vertaald’ naar de vooraf gedefinieerde metric, zodat een analyse mogelijk is. De uitkomst van de tests van de verschillende pakketten zullen met elkaar worden vergeleken. Ook wordt de uitkomst van de verschillende datasets met elkaar vergeleken.

Meting: Exacte metingen voor deze deelvraag zijn nog niet gedefinieerd. Dit omdat het nog niet zeker is of er wel genoeg tijd is om deze deelvraag goed genoeg te kunnen beantwoorden. Een mogelijke test is om uit te zoeken wat het effect is als een server zwaarder belast wordt. Tevens kan er gemeten worden hoe groot de extra opgelopen vertraging is bij het gebruik van DNS-systemen die niet op het interne netwerk beschikbaar zijn. Door gebruik van deze gegevens kan een beeld geschat worden wat voor effect zo'n uitrol zou hebben op een organisatie of zelfs landelijk. Tevens zullen er bij het behandelen van deze deelvraag veel aannames gedaan worden, zoals bijvoorbeeld het gemiddelde stroomverbruik van clients, de verhoging in latency et cetera. Al met al maakt dat dit een lastige vraag om goed te beantwoorden.

5.4 Meetplan

In dit hoofdstuk worden de metingen, het doel van de metingen, de aannames, de randvoorwaarden en de meetprotocollen beschreven.

5.4.1 Doel metingen

Het doel van de metingen is om het energieverbruik in verschillende situaties vast te stellen. Met de resultaten van deze metingen zijn de verschillende onderzoeks vragen beantwoord.

5.4.2 Aannames

De volgende aannames gelden bij de metingen:

- De meetopstelling is goed gekalibreerd
 - Verantwoordelijkheid SEFlab
 - Kalibratie is achteraf toegepast
- De meetopstelling moet opwarmen, 30 minuten is genoeg om op te warmen
 - Niet opwarmen beïnvloedt het verbruik, het verbruik is hoger door allerlei lopende systeemprocessen na het opstarten. Dit is waargenomen tijdens proefmetingen.
- De dataset wordt doormiddel van een script bij elke test op dezelfde manier afgespeeld.
 - Achteraf gecontroleerd in de logfile op de replay-laptop
- Het DNS-pakket functioneert naar behoren

5.4.3 Randvoorwaarden

De volgende randvoorwaarden gelden bij de metingen:

- De omgeving is correct geconfigureerd
 - Oftewel: Juiste besturingssysteem, juiste DNS software etc.
- De systeemtijd is overal hetzelfde
 - Doormiddel van een NTP synchronisatie, belangrijk om meetgegevens van verschillende systemen met elkaar te kunnen vergelijken.

5.4.4 Initiele meetprotocollen

Het was de bedoeling dat de metingen volgens de onderstaande protocollen worden uitgevoerd. Later zijn de meetprotocollen echter aangepast om verwerking en analyse van de data makkelijker te maken. Hier worden eerst de initiële meetprotocollen beschreven en daarna de aangepaste protocollen. Bij de opzet van deze protocollen is rekening gehouden met de aannames en randvoorwaarden. Beide meetprotocollen zijn toepasbaar op een aantal cases. De volgende cases moeten worden uitgezocht om de onderzoeks vragen te beantwoorden:

- Baseline metingen
 - Baseline verbruik Windows
 - Het baseline verbruik van het Windows besturingssysteem, met een idle systeem.
 - Baselineverbruik van Windows met idle DNS pakket geïnstalleerd
 - Het baseline verbruik van Windows in combinatie met een idle DNS software pakket.
 - Baseline verbruik Linux
 - Het baseline verbruik van het Linux besturingssysteem, met een idle systeem.
 - Baselineverbruik van Linux met idle DNS pakket geïnstalleerd
 - Het baseline verbruik van Linux in combinatie met een idle DNS software pakket.
- DNS Pakket Meting
 - Metingen van DNS softwarepakketten die een dataset te verwerken krijgen, met of zonder DNSSEC
 - Windows
 - Verbruik Microsoft DNS;
 - Verbruik BIND;
 - Verbruik Unbound;
 - Verbruik Microsoft DNS met DNSSEC;
 - Verbruik BIND met DNSSEC;
 - Verbruik Unbound met DNSSEC.
 - Linux
 - Verbruik BIND;
 - Verbruik Unbound;
 - Verbruik BIND met DNSSEC;
 - Verbruik Unbound met DNSSEC.

5.4.5 OS Baseline meting

Het doel van deze meting is om het energieverbruik van het besturingssysteem in idle toestand te meten, zodat dit in een later stadium makkelijk gescheiden kan worden van het

energieverbruik van het DNS pakket. Om die meting met succes te kunnen uitvoeren moet er voldaan zijn aan een aantal randvoorwaarden specifiek voor deze test. Er wordt aangenomen dat de schermbeveiliging of zelfs het in slaapstand gaan van het beeldscherm invloed kan hebben op het energieverbruik van de server, en dus op het resultaat van de meting. Om dit te voorkomen worden beide functies dus uitgeschakeld. Ook worden geplande taken/automatisch onderhoud voor het besturingssysteem uitgeschakeld.

De verwachting is dat het energieverbruik van het besturingssysteem consistent is. Dus relatief stabiel, of met een herkenbaar patroon. Dit zou het makkelijk maken om het energieverbruik van het besturingssysteem van het energieverbruik van het DNS pakket te scheiden.

Test procedure		
Stap nummer	Actie	Verwacht resultaat
1	Het opnemen van de meetgegevens wordt ingeschakeld op de meet-PC	De meet-PC begint met het produceren van een file met meetresultaten.
2	Energetest wordt gestart en 45 minuten lang uitgevoerd	Energetest start en produceert resultaten. Deze resultaten zijn live te volgen en na afloop van de test zichtbaar op de meetcomputer
3	Het opnemen van de meetgegevens wordt gestopt	De meet-PC heeft een file met meetresultaten geproduceerd
6	Resultaten worden geanalyseerd	Uitkomst van de energetest wordt geanalyseerd en aan de hand van vooraf vastgestelde foutmarges beoordeeld. De meting moet binnen deze foutmarge resultaten produceren. Zo niet, moet er uitgezocht worden waarom niet.

Tabel 3 Baseline meetprotocol

5.4.6 DNS-pakket meting

Het doel van deze meting is om het energieverbruik van het DNS pakket te meten. Om die meting met succes te kunnen uitvoeren moet er voldaan zijn aan een aantal randvoorwaarden specifiek voor deze test. Er wordt aangenomen dat de schermbeveiliging of zelfs het in slaapstand gaan van het beeldscherm invloed kan hebben op het energieverbruik van de server en het resultaat van de meting. Om dit te voorkomen worden beide functies dus uitgeschakeld. Ook worden geplande taken/automatisch onderhoud voor het besturingssysteem uitgeschakeld. Tevens is de besturingssysteem baselinemeting nodig om het resultaat van deze meting te krijgen. De meetresultaten uit deze meting zijn namelijk het energieverbruik van het DNS-pakket in combinatie met het besturingssysteem, en die twee moeten van elkaar gescheiden worden.

Om consistentie te kunnen garanderen voor de verschillende keren dat deze meting uitgevoerd wordt, wordt het afspelen van data afgehandeld door een script. De specifieke acties die het script uitvoert zijn te vinden in bijlage V.

De verwachting is dat het energieverbruik van het DNS-pakket te scheiden is van het energieverbruik van het besturingssysteem.

Test procedure		
Stap nummer	Actie	Verwacht resultaat
1	Het opnemen van de meetgegevens wordt ingeschakeld op de meet-pc	De meet-pc begint met het produceren van een file met meetresultaten.
2	Op de replay-laptop wordt het afspeelscript gestart	Het script noteert de starttijd, schermuitvoer en eindtijd van het afspelen van de PCAP file in een log bestand. De DNS-server krijgt queries te verwerken waardoor, het stroomverbruik omhoog gaat.
3	Het script zal de dataset drie keer afspelen. Elke keer wordt de begin en eindtijd per run genoteerd. Tussen elke run zal 5 minuten zitten.	Het script noteert de starttijd, schermuitvoer en eindtijd van het afspelen van de pcap file in een logfile. De DNS-server krijgt queries te verwerken waardoor het stroomverbruik waarschijnlijk omhoog gaat. Bij de eerste run zal het verbruik vermoedelijk hoger zijn dan bij de tweede en derde run. Het stroomverbruik van de tweede en derde run zal vermoedelijk relatief gelijk zijn.
4	Het opnemen van de meetgegevens wordt gestopt	De meet-pc heeft een file met meetresultaten van alle drie de runs geproduceerd

Tabel 4 DNS Replay meetprotocol

5.4.7 Aanpassing meetprotocollen

Tijdens de metingen gingen er een aantal dingen mis. Zo werden er kapotte meetbordjes vervangen die vervolgens andere waardes aangaven en ging het PSU-meetbord kapot. Door het vervangen van bordjes en het opmerken van de verschillen in resultaten was verder onderzoek nodig. Het was nodig de bordjes te kalibreren en uiteindelijk heeft dit een nauwkeuriger meetsysteem opgeleverd. Hieruit bleek wel dat de resultaten van voorgaande metingen ongeldig waren. Er moesten dus metingen opnieuw gedaan worden. En gezien de hoeveelheid metingen die er gedaan moesten worden is er besloten om de meetprotocollen aan te passen. In overleg met Eric en Roland is er een nieuw efficiënter meetprotocol opgesteld. De baseline en gebruiksmetingen zijn samengevoegd. Ook zijn hier dezelfde randvoorwaarden weer van toepassing:

- Schermbeveiliging/slaapstand beeldscherm uitgeschakeld;
- Geplande taken uitgeschakeld;
- Automatisch onderhoud uitgeschakeld.

Het nieuwe meetprotocol is hetzelfde geworden voor alle tests en is als volgt ingedeeld: elke meting bestaat uit acht slots, elk slot bestaat uit een meting van 10 minuten en 20 seconden. Tussen elk slot zit een afkoeltijd van cijf minuten. De eerste twee slots zijn idle slots, daar kan het baselinegebruik van het operating system gemeten worden. Tijdens de slots 3 tot en met 8 wordt er DNS-data naar de server gestuurd, elke keer hetzelfde pakket aan data. Dit zorgt ervoor dat de cache in slot 3 gevuld wordt, en de andere slots data uit de cache kunnen halen. De totale looptijd van een meting is 2 uur.

Dat levert de testprocedure op die te zien is in tabel 5

Test procedure		
Stap nummer	Actie	Verwacht resultaat
1	De tijd wordt doormiddel van NTP gesynchroniseerd op alle systemen met de server time.nist.gov	
2	Er wordt een handmatige proefquery gedaan vanaf de replay laptop om werking van de resolver te garanderen.	De DNS resolver beantwoordt de query.
3	De cache van de DNS resolver wordt gewist en het DNS pakket wordt opnieuw opgestart door middel van ingebouwde functie van DNS software.	Eventuele cache entries door de proefquery worden gewist.
4	Het opnemen van de meetgegevens wordt ingeschakeld op de meet-PC.	De meet-PC begint met het produceren van een file met meetresultaten.
5	Het script om DNS-data af te spelen wordt gestart op de replaylaptop. Het script is te zien in bijlage V.	De replaylaptop doorloopt het replay script en verwerkt de 8 slots. Het script genereert een logfile met de schermoutput en de tijden van alle slots. Een voorbeeld van de logfile is te zien in bijlage VI
6	Het opnemen van de meetgegevens wordt gestopt op de meet-PC	De meet-data van de afgelopen 2 uur wordt opgeslagen.
7	De DNS-dataset wordt nog één keer handmatig afgespeeld vanaf de replaylaptop, en de antwoorden worden opgeslagen in een PCAP-file	De PCAP-file bevat de antwoorden van de resolver, dit om latere verificatie van de resultaten mogelijk te maken en werking van de resolver te garanderen.

Tabel 5 Nieuwe meetprotocol

6 Resultaten

In dit hoofdstuk worden de resultaten van de metingen besproken die relevant zijn voor de genoemde deelvraag. Om deze resultaten goed te kunnen interpreteren is het belangrijk om te weten hoe de resultaten opgebouwd zijn en welke resultaten er precies gebruikt worden. Het verwerken van een dataset in MATLAB levert een grote hoeveelheid data op. De variabelen die goed bruikbaar zijn voor het presenteren van de data zijn *slot_data_joules* en *channel_power*. *Slot_data_joules* bevat een tabel van 12 kolommen en 8 rijen. De verschillende kolommen staan voor de verschillende meetkanalen, en de rijen staan voor de gemeten slots. In de resultaten worden de tweede, vierde en achtste kolom weggelaten. De tweede en vierde kolom bevatten CPU2 en MEM2, die niet aangesloten zitten in deze meetopstelling. En kolom 8 bevat het totale verbruik van de PSU. Deze data wordt weergegeven in staafgrafieken, en tabellen met de ruwe getallen. *Channel_power* bevat het verbruik in watt per sample. Deze data wordt weergegeven in lijngrafieken en geeft een mooi beeld van het verloop van de meting. Al deze grafieken zijn te vinden in bijlage VIII. Daarnaast zijn de resultaten van stap 7 uit de testprocedure nog relevant. Als die afwijken van het normale gedrag wordt dat ook gemeld, tevens zijn de resultaten in de bijlage te vinden.

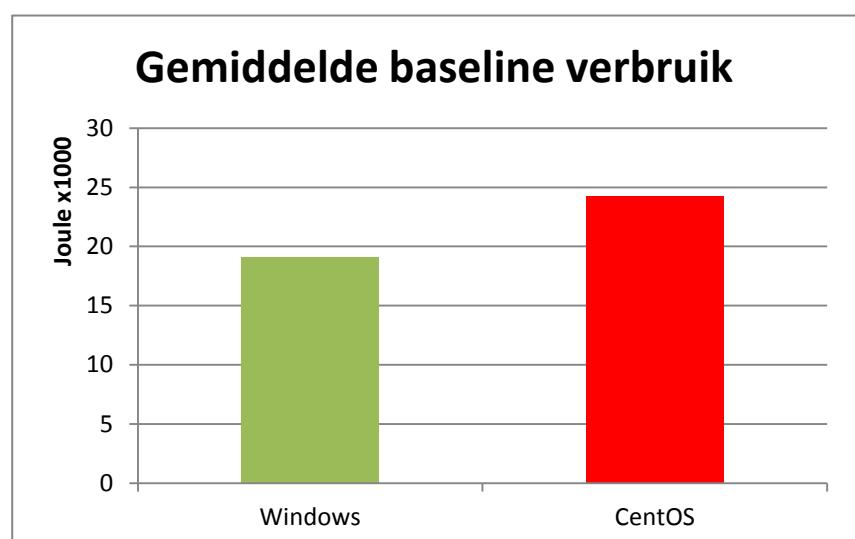
Tenzij anders vermeld, is elk DNS-pakket drie keer gemeten, een keer zonder DNSSEC, een keer met DNSSEC, en een controlemeting. In deze resultaten wordt de controlemeting niet meegenomen tenzij er vreemde resultaten uit de controlemeting zijn gekomen. Alle grafieken in dit hoofdstuk zijn gekleurd op efficiëntie. Waarbij het efficiëntste pakket groen gekleurd is, en het minst efficiënte pakket rood.

Unbound op Windows week af van het ‘normale’ gedrag. Dit gedrag komt terug in alle metingen van Unbound op Windows. Naast het afwijkende energieverbruik was Unbound op Windows ook niet in staat om alle queries te beantwoorden. Waar alle pakketten rond de 1,2 miljoen antwoorden terugstuurde slaagde Unbound er niet in om hoger te komen dan ongeveer 750.000 antwoorden. Daarom is besloten de resultaten van Unbound op Windows niet mee te nemen in de rest van het onderzoek. Het gedrag van Unbound wordt geanalyseerd in bijlage XI.

6.1 Baseline verbruik

In figuur 8 is het baseline verbruik te zien. Het is belangrijk om te noemen dat CentOS gemiddeld 5096 Joule meer verbruikt in tien minuten. Dit heeft als gevolg dat alle resultaten gemeten op CentOS hoger uitvallen dan de resultaten op Windows. In de vergelijkingsgrafieken die hier na volgen is het baseline verbruik per meting van het totaal af gehaald. Zo blijft alleen het verbruik van het

DNS pakket over.

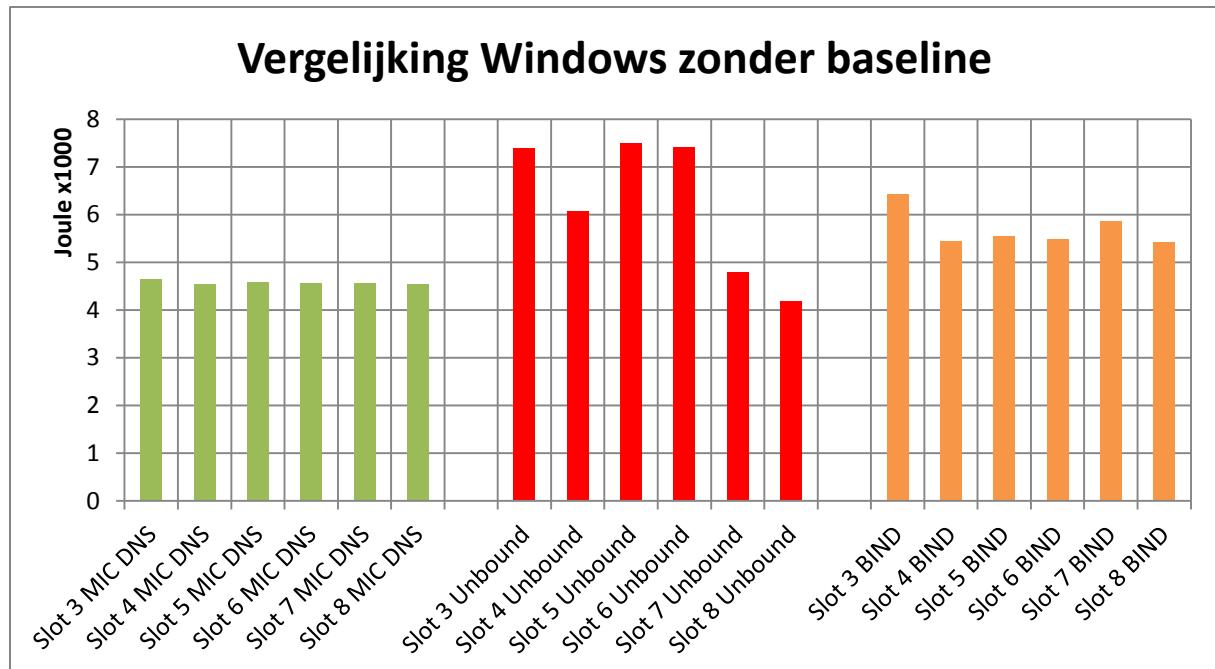


Figuur 8 Gemiddelde baseline verbruik

6.2 Hoofdvraag

'Welk van de drie gemeten DNS software pakketten gaat het efficiëntste om met energie?'

Er zijn een aantal situaties die kunnen worden vergeleken bij het beantwoorden van deze deelvraag. Zo zijn er twee verschillende besturingssystemen waar rekening mee gehouden moet worden. Daarom worden de pakketten op Windows vergeleken, en de pakketten op Linux vergeleken. Daarna worden de verschillen van BIND en Unbound op Windows en Linux bekeken. De volgende grafiek bevat de eerste vergelijking van DNS pakketten op Windows. Slots 1 en 2 bevatten het baseline verbruik en zijn uit deze grafiek geknipt.



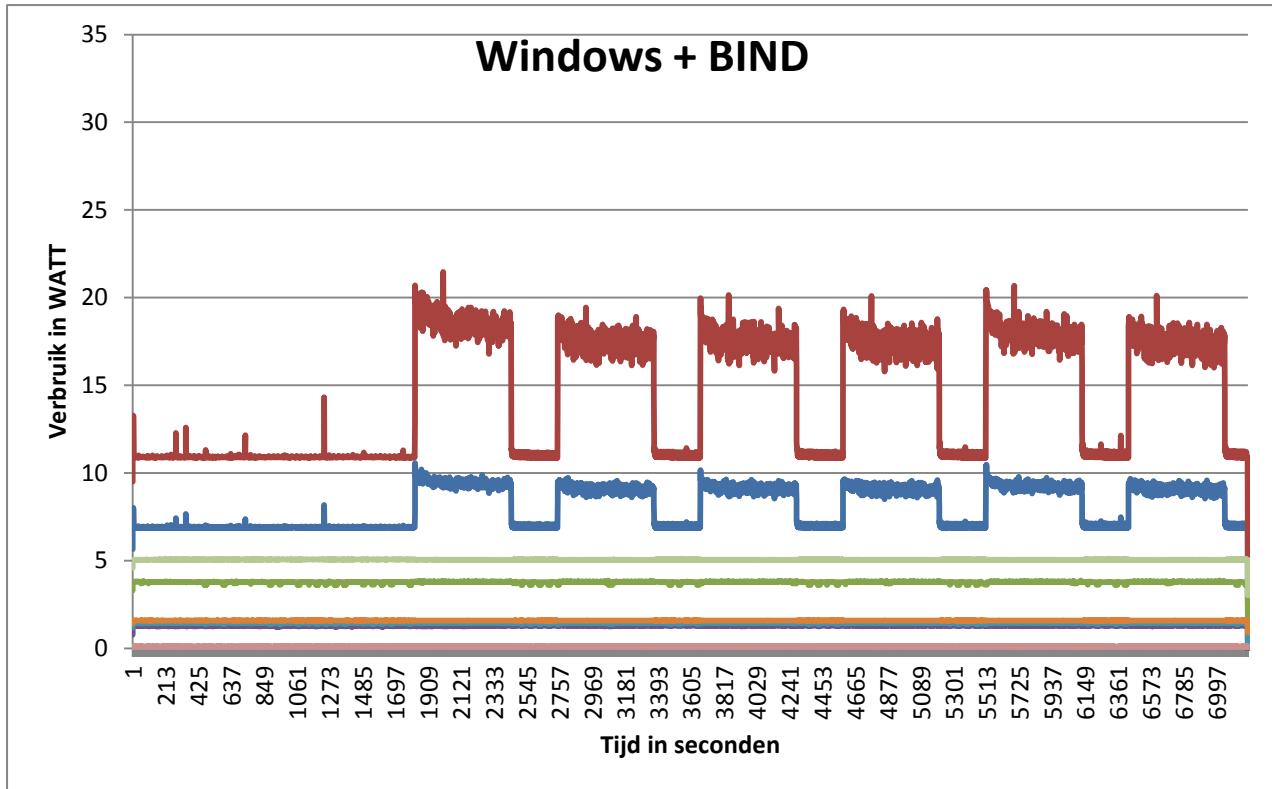
Figuur 9 Vergelijking Windows

Het gedrag van Windows met Microsoft DNS is heel constant en voorspelbaar. Het verbruik in het eerste verbruikslot ligt ongeveer 100 Joule hoger dan de andere 5 slots. 100 Joule in 10 minuten staat gelijk aan 0.16 Watt per seconde.⁷

Het gedrag van Unbound is onvoorspelbaar, zo zit er een verschil van ruim 3200 Joule tussen het eerste en het laatste verbruikslot. Dat staat gelijk aan ongeveer 5 Watt per seconde.

In tegenstelling tot Unbound is het gedrag van BIND weer mooi consistent en voorspelbaar. Het verbruik van het eerste en het vijfde verbruikslot ligt wat hoger dan dat van de andere slots. Dit komt overeen met de voorspellingen dat het verbruik wat hoger zou liggen tijdens het vullen van de cache. Het verschil tussen het eerste en het tweede slot is 990 Joule, dat staat gelijk aan 1,6 Watt per seconde. Dit is ook terug te zien in onderstaande grafiek. De legenda is verwijderd in verband met het formaat, de bovenste lijn is de CPU, de blauwe lijn het geheugen. Deze grafiek is beschikbaar voor alle metingen in bijlage VIII.

⁷ Bron [6] RapidTables, (datum onbekend). Joules to Watts calculator.
http://www.rapidtables.com/calc/electric/Joule_to_Watt_Calculator.htm

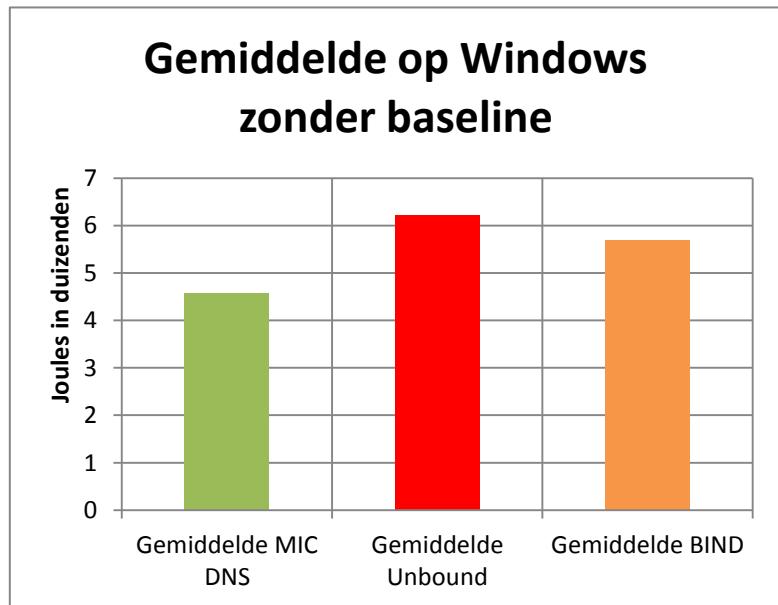


Figuur 11 Power verbruik Windows + BIND

Om de resultaten nog iets te verduidelijken wordt nu in de volgende grafiek het gemiddelde verbruik over alle verbruikslots zonder baseline meting weergegeven. Die grafiek is ook makkelijk te vergelijken met andere metingen zoals bijvoorbeeld de DNSSEC-meting en de Linux-meting.

Zoals hiernaast te zien is, is Microsoft DNS gemiddeld het efficiëntst op Windows. Gevolgd door BIND en vlak daarna Unbound. Bij dit resultaat moet er echter rekening mee gehouden worden dat Unbound vreemd gedrag vertoont dat de resultaten sterk beïnvloed. De exacte getallen van het gemiddelde verbruik in 10 minuten zijn als volgt:

- Microsoft DNS: 4567,85 Joule (7,37 Watt per seconde)
- Unbound: 6222,88 Joule (10,04 Watt per seconde)
- BIND: 5690,04 Joule (9,18 Watt per seconde)

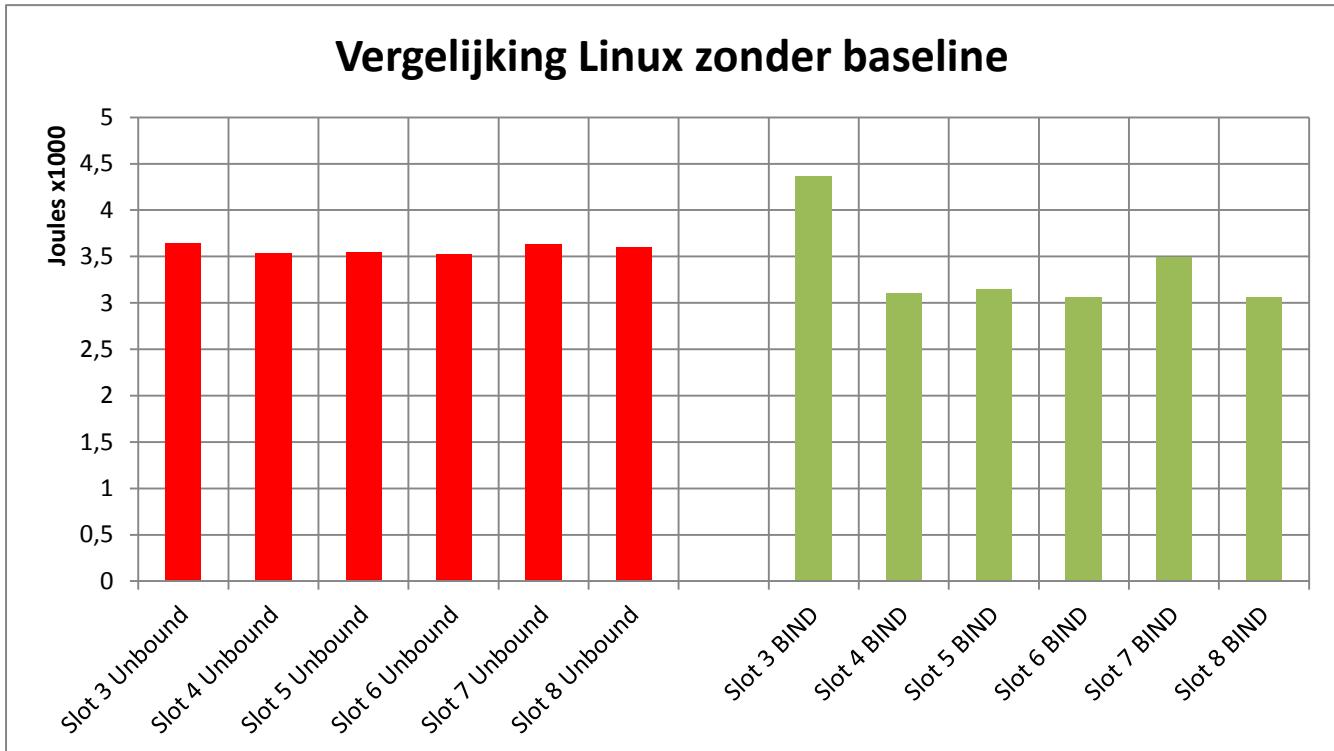


Figuur 10 Gemiddelde op Windows

Hieronder volgt de grafiek met de Linux vergelijking. Hier zijn alleen BIND en Unbound in opgenomen omdat Microsoft DNS, zoals al eerder vermeld, niet werkt op Linux.

Het gedrag van Unbound op Linux is in tegenstelling tot het gedrag op Windows zeer constant. Het eerste en vierde verbruikslot hebben een lichte verhoging in energieverbruik. Dit komt overeen met de voorspellingen dat het verbruik wat hoger zou liggen tijdens het vullen van de

cache. Het grootste verschil tussen de slots is 116 Joule, dat staat gelijk aan 0.19 Watt per seconde.



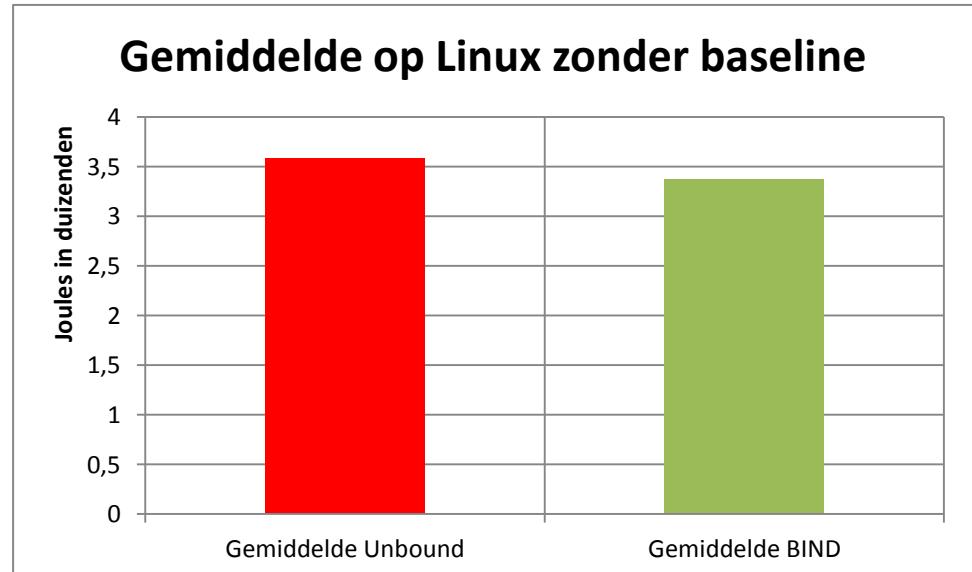
Figuur 12 Vergelijking CentOS

Het gedrag van BIND op Linux lijkt sterk op het gedrag dat BIND vertoont op Windows. Het eerste en vierde verbruiksslot hebben een verhoging in energieverbruik wat wederom overeenkomt met de voorspellingen. Het grootste verschil tussen de slots is 1303 Joule, dat staat gelijk aan 2,1 Watt per seconde.

Om de resultaten nog wat verder te verduidelijken wordt in de volgende grafiek het gemiddelde verbruik van alle slots weergegeven. Die grafiek is weer makkelijker te vergelijken met DNSSEC- of Windows-metingen.

Zoals hiernaast te zien is is BIND het efficiëntste op Linux. Met vlak daarachter Unbound. Het verschil is echter klein. De exacte getallen zijn als volgt:

- Unbound: 3579,99 Joule (5,77 Watt per seconde)
- BIND: 3369,34 Joule (5,43 Watt per seconde)



6.3 Deelvragen

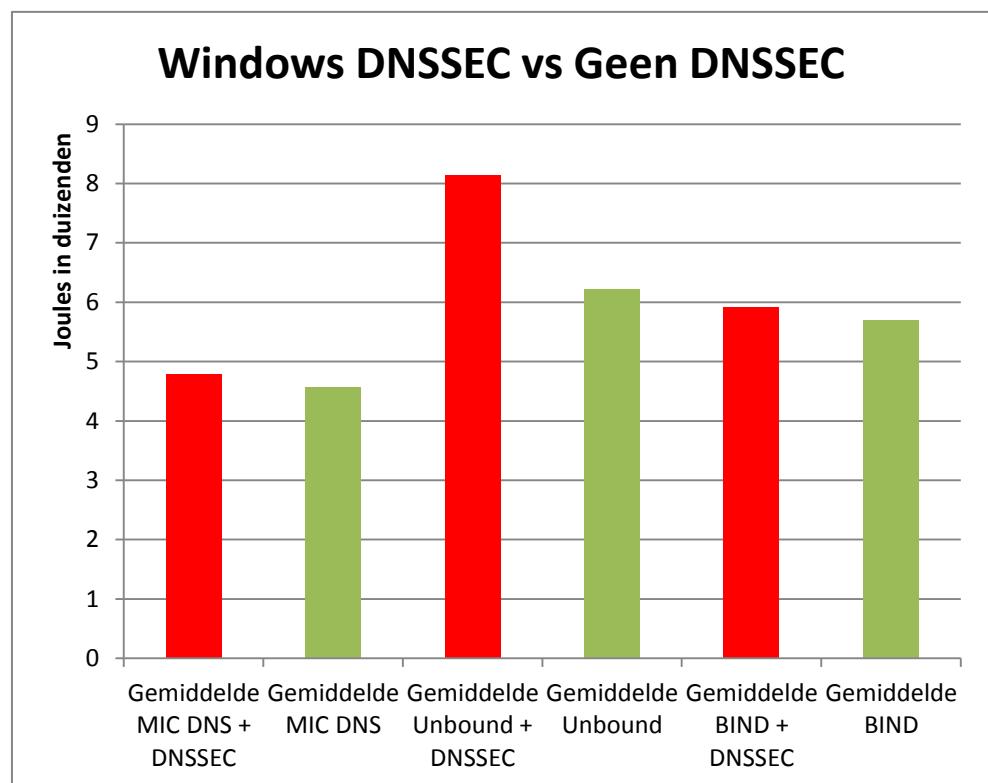
'Wat is de invloed van het gebruiken van DNSSEC op het energieverbruik van de verschillende pakketten?'

Het inschakelen van DNSSEC heeft tot gevolg dat antwoorden gevalideerd worden. Een deel van die validatie bestaat uit cryptografische berekeningen. In theorie kosten cryptografische berekeningen meer energie. Alle resultaten en grafieken die te zien zijn bij de resultaten van de hoofdvraag zijn ook beschikbaar voor deze deelvraag. Het is echter interessanter om meteen naar de vergelijking van het gemiddelde met DNSSEC, en het gemiddelde zonder DNSSEC te kijken. De overige grafieken zijn te vinden in bijlage VIII.

Hiernaast is de vergelijking te zien op Windows. Er is duidelijk te zien dat het verbruik met DNSSEC ingeschakeld hoger ligt dan het verbruik zonder DNSSEC. Kanttekening hierbij is dat de resultaten van Unbound op Windows ook met de DNSSEC meting onbetrouwbaar zijn. Uit de gegevens van de

Unbound meting kan dan ook geen goed onderbouwde conclusie getrokken worden. De exacte verschillen zijn als volgt:

- Microsoft DNS: extra verbruik DNSSEC: 221,29 Joule
- Unbound: extra verbruik DNSSEC: 1919,8 Joule⁸
- BIND: extra verbruik DNSSEC: 226,45 Joule



Figuur 13 Windows DNSSEC vs Geen DNSSEC

⁸ Onbetrouwbaar, zie bijlage XI

Ook met DNSSEC ingeschakeld is Microsoft DNS nog steeds het efficiëntste pakket op Windows, gevolgd door BIND. Het gedrag van Unbound staat vermeld omdat de gegevens beschikbaar zijn, maar de gegevens zijn onbetrouwbaar. In bijlage XI wordt het gedrag van Unbound op Windows geanalyseerd.

Ook op Linux is het verschil met of zonder DNSSEC goed zichtbaar. Waar er zonder DNSSEC nog een duidelijk verschil zichtbaar was tussen het verbruik van Unbound en BIND, lijkt het verschil met DNSSEC verdwenen. Als er ook naar de exacte cijfers gekeken wordt is te zien dat BIND met DNSSEC zelfs meer is gaan verbruiken dan Unbound met DNSSEC.

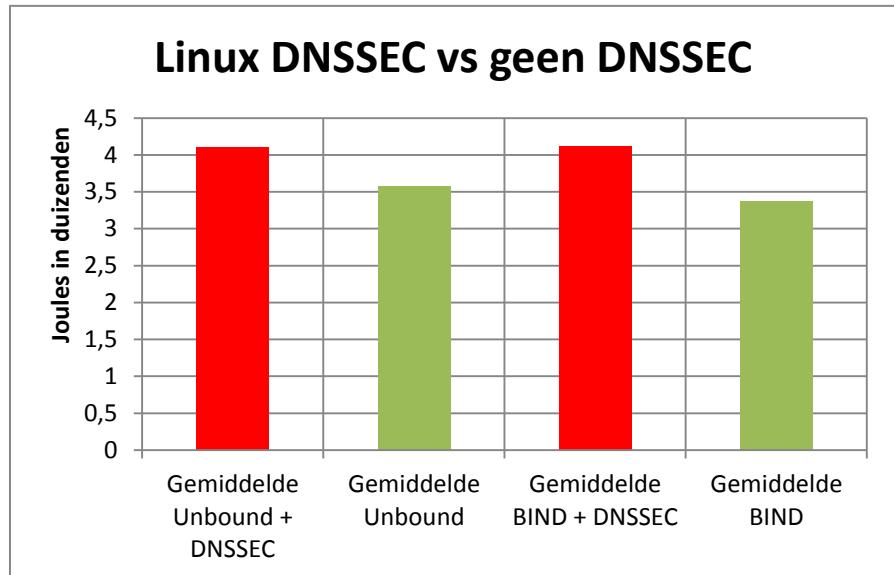
De exacte verschillen zijn als volgt:

- Unbound: Extra verbruik DNSSEC: 527,85 Joule
- BIND: extra verbruik DNSSEC: 753,54 Joule

'Wat zou er bespaard kunnen worden door gebruik van één van de te onderzoeken pakketten in vergelijking met de andere pakketten?'

Omdat het energieverbruik van zo veel verschillende factoren afhankelijk is worden hier twee vergelijkingen gemaakt. Een tussen BIND en Microsoft DNS op Windows, en een tussen BIND en Unbound op Linux. Dit heeft twee redenen, ten eerste is het gedrag van Unbound op Windows niet betrouwbaar. Het is niet mogelijk om een goed onderbouwde conclusie te trekken op basis van de verkregen resultaten. Ten tweede zijn de verschillen in het baselineverbruik van Windows tegenover Linux groot. Als er een onderlinge vergelijking gemaakt zou worden zou dat daar door een vergelijking tussen de verschillende operating systems worden.

Om de deelvraag goed te kunnen beantwoorden is het handig om te gaan kijken naar het verbruik per uur. De beschikbare meetdata geeft het verbruik weer per tien minuten. Het gemiddelde gebruik per tien minuten wordt daarom omgerekend naar een gemiddeld gebruik per uur. Daarnaast is het van belang om te weten wat energie nou eigenlijk kost. Een kort onderzoek wijst uit dat een kWh gemiddeld €0,17 kost bij een data center.⁹



Figuur 14 Linux DNSSEC vs Geen DNSSEC

⁹ Bron [8] Bhosted, (datum onbekend). Bhosted.nl rackspace. <http://www.bhosted.nl/colocatie/rackspace>
Bron [9] Base IP (datum onbekend). Base IP Network serviceprovider. <https://www.baseip.com/8-diensten/9-colocatie.html>

Bron [10] Bitency (datum onbekend). Bitency Colocatie. <https://bitency.nl/colocatie>

Het gemiddelde verbruik per tien minuten voor Microsoft DNS is 4567,85 Joule, dat geeft een gemiddelde van 27407,1 Joule per uur. 27407,1 Joule staat gelijk aan 0,007613 kWh.¹⁰ Dat levert een prijs op van €0,001294 per uur.

Het gemiddelde verbruik per tien minuten voor BIND is 5690,04 Joule, dat geeft een gemiddelde van 34140,2 Joule per uur. 34140,2 Joule staat gelijk aan 0,0094834 kWh. Dat levert een prijs op van €0,001612 per uur.

Deze prijzen zijn eigenlijk zo klein dat het weinig impact heeft. Voor het gemak wordt er nu van uit gegaan dat een DNS-server constant dezelfde load heeft voor een heel jaar. In de realiteit gebeurt dit natuurlijk niet, maar het maakt de resultaten wat inzichtelijker.

Kosten per:	Uur	Dag	Jaar
Microsoft DNS	€0,001294	€0,03106	€11,33
BIND	€0,001612	€0,03869	€14,12

Tabel 6 Kostenvergelijking DNS Windows

Mocht er voor gekozen worden om gebruik te maken van Microsoft DNS in plaats van BIND kan dat een besparing opleveren van €2,78 op jaarbasis.

Dezelfde vergelijking wordt gemaakt tussen Unbound en BIND op Linux:

Het gemiddelde verbruik per tien minuten voor Unbound is 3579,99 Joule, dat geeft een gemiddelde van 21479,9 Joule per uur. 21479,9 joule staat gelijk aan 0,005966 kWh. Dat levert een prijs op van €0,001014 per uur.

Het gemiddelde verbruik per tien minuten voor BIND is 3369,34 Joule, dat geeft een gemiddelde van 20216,04 Joule per uur. 20216,04 Joule staat gelijk aan 0,005615 kWh. Dat levert een prijs op van €0,00095464 per uur.

Deze prijzen zijn wederom zo klein dat het weinig impact heeft, daarom worden ze in dezelfde tabel ingevoerd

Kosten per:	Uur	Dag	Jaar
Unbound	€0,001014	€0,02434	€8,885
BIND	€0,0009546	€0,02291	€8,362

Tabel 7 Kostenvergelijking DNS Linux

Mocht er voor gekozen worden om gebruik te maken van BIND in plaats van Unbound kan dat een besparing opleveren van €0,52 op jaarbasis.

Dezelfde vergelijking kan gemaakt worden op het moment dat DNSSEC is ingeschakeld. De energievergelijking met en zonder DNSSEC wordt behandeld in de volgende deelvraag.

Het gemiddelde verbruik per tien minuten voor Microsoft DNS is 4789,15 Joule, dat geeft een gemiddelde van 28734,9 Joule per uur. 28734,9 Joule staat gelijk aan 0,007981 kWh. Dat levert een prijs op van € 0,001356 per uur.

Het gemiddelde verbruik per tien minuten voor BIND is 5916,5 Joule, dat geeft een gemiddelde van 35499 Joule per uur. 35499 Joule staat gelijk aan 0,009861 kWh. Dat levert een prijs op van €0,001676 per uur.

¹⁰ Bron [7] RapidTables, (datum onbekend). Joules to kWh conversion.
http://www.rapidtables.com/convert/energy/Joule_to_kWh.htm

Deze prijzen zijn eigenlijk zo klein dat ze niet veel zeggen. Voor het gemak gaan we er nu even van uit dat een DNS-server constant dezelfde load heeft voor een heel jaar. In de realiteit gebeurt dit natuurlijk niet, maar het maakt de resultaten wat inzichtelijker.

Kosten per:	Uur	Dag	Jaar
Microsoft DNS	€0,001356	€0,03256	€11,88
BIND	€0,001676	€0,04023	€14,68

Tabel 8 Kostenvergelijking DNSSEC Windows

Mocht er voor gekozen worden om gebruik te maken van Microsoft DNS in plaats van BIND kan dat een besparing opleveren van €2,80 op jaarrbasis.

Dezelfde vergelijking wordt gemaakt tussen Unbound en BIND op Linux.

Het gemiddelde verbruik per tien minuten voor Unbound is 4107,84 Joule, dat geeft een gemiddelde van 24647,0 Joule per uur. 24647,0 Joule staat gelijk aan 0.006846 kWh. Dat levert een prijs op van €0,001163 per uur.

Het gemiddelde verbruik per tien minuten voor BIND is 4122,88 Joule, dat geeft een gemiddelde van 24737,3 Joule per uur. 24737,3 Joule staat gelijk aan 0.006871 kWh. Dat levert een prijs op van €0,001168 per uur.

Kosten per:	Uur	Dag	Jaar
Unbound	€0,001163	€0,02793	€10,19
BIND	€0,001168	€0,02803	€10,23

Tabel 9 Kostenvergelijking DNSSEC Linux

Mocht er voor gekozen worden om gebruik te maken van Unbound in plaats van BIND kan dat een besparing opleveren van €0,04 op jaarrbasis.

'Kan het aanpassen van de configuratie van een of meer van de pakketten leiden tot een efficiënter gebruik van energie?'

Helaas is er tijdens het project niet genoeg tijd geweest om deze optionele vraag goed te kunnen beantwoorden. Deze vraag is dan ook terug te vinden onder toekomstig werk. Er kan bijvoorbeeld gekeken worden naar:

- Aanpassen van de cache size
 - Het vergroten/verkleinen van de cache zou een grote invloed kunnen hebben op het energieverbruik
- Aanpassen van de TTL
 - Gezien de resultaten van de metingen bestaat het vermoeden dat het verhogen van de TTL (langer geldig zijn van cache entries) voor een lager energieverbruik zorgt.
- Logging
- Threading

'Is het efficiënter om resolver functionaliteit te verplaatsen naar een hoger niveau in het netwerk?'

Helaas is er tijdens het project niet genoeg tijd geweest om deze optionele vraag goed te kunnen beantwoorden. Deze vraag is dan ook terug te vinden onder toekomstig werk. Er kan bijvoorbeeld gekeken worden naar:

- Extra latency
 - Door het verplaatsen van resolver-functionaliteit duurt het vermoedelijk een aantal milliseconde langer voordat een client antwoord krijgt op een query. Bij een instelling als de HvA zijn al gauw duizenden clients tegelijkertijd actief. Als al die clients een aantal milliseconde langer moeten wachten op hun antwoord kost dat in theorie meer energie.
- Machine load
 - Als meerdere instellingen gebruik maken van een resolver wordt de load op die machine hoger. Weegt het extra verbruik van die machine(s) op tegen het uitschakelen van resolvers bij de instellingen?
- Infrastructuur
 - Is er extra infrastructuur nodig om dit te bewerkstelligen? Zo ja, weegt het energieverbruik van die infrastructuur op tegen het uitschakelen van resolvers bij de instellingen?

7 Conclusies

7.1 Hoofdvraag

'Welk van de drie gemeten DNS software pakketten gaat het efficiëntste om met energie?'

Het antwoord op deze deelvraag is door de aard van het onderzoek in tweeën te delen. Namelijk: 'Welk DNS-pakket gaat het efficiëntst om met energie op Windows?' en 'Welk DNS-pakket gaat het efficiëntst om met energie op Linux?'. Onderling kan er per besturingssysteem ook nog onderscheid gemaakt worden tussen wel/niet gebruiken van DNSSEC.

Op Windows komt Microsoft DNS als efficiëntst uit de metingen. Zowel met, als zonder DNSSEC verbruikt Microsoft DNS het minste energie in vergelijking met BIND en Unbound.

Op Linux komt zonder DNSSEC BIND als efficiëntst uit de metingen. Wordt DNSSEC echter ingeschakeld dan wordt het verschil tussen BIND en Unbound heel klein. BIND gaat zelfs iets meer verbruiken dan Unbound. Het verschil is echter te klein om te kunnen zeggen welk pakket efficiënter is.

De efficiëntste combinatie binnen de scope van dit onderzoek is Windows met Microsoft DNS. Als vervolgonderzoek is het interessant om naast CentOS ook nog andere Linux-distributies te meten. Er kan dan gekeken worden of het verbruik van de DNS-pakketten ook nog verschilt per distributie.

7.2 Deelvragen

'Wat zou er bespaard kunnen worden door gebruik van één van de te onderzoeken pakketten in vergelijking met de andere pakketten?'

Ook deze deelvraag kan in tweeën gedeeld worden, namelijk: 'Wat zou er bespaard kunnen worden door gebruik van één van de pakketten op Windows in vergelijking met de andere pakketten?' en 'Wat zou er bespaard kunnen worden door gebruik van één van de pakketten op Linux in vergelijking met het andere pakket?'.

Windows:

Mocht er voor gekozen worden om gebruik te maken van Microsoft DNS in plaats van BIND kan dat een besparing opleveren van €2,78 op jaarrbasis. Wordt DNSSEC ingeschakeld op beide pakketten dan veranderd de besparing naar €2,80 op jaarrbasis.

Linux:

Mocht er voor gekozen worden om gebruik te maken van BIND in plaats van Unbound kan dat een besparing opleveren van €0,52 op jaarrbasis. Als DNSSEC wordt ingeschakeld wordt Unbound iets zuiniger dan BIND. Keuze voor het zuinigste pakket kan een besparing opleveren van €0,04 op jaarrbasis.

De besparing is dusdanig klein dat men zich kan afvragen of dit een rol zou moeten spelen bij de keuze voor een DNS pakket.

'Wat is de invloed van het gebruiken van DNSSEC op het energieverbruik van de verschillende pakketten?'

Het gebruiken van DNSSEC heeft één effect dat voor alle pakketten hetzelfde is, het energieverbruik gaat omhoog. Het verschil is niet even groot voor elk pakket. Opvallend is dat

het verbruik op Linux met DNSSEC meer verschilt van het normale gebruik, dan het gebruik op Windows met DNSSEC. Het hogere verbruik van Unbound met DNSSEC op Windows is helaas onbetrouwbaar, waardoor er geen conclusie uit getrokken kan worden. De exacte verschillen zijn als volgt:

- Microsoft DNS op Windows: extra verbruik DNSSEC: 221,29 Joule
- Unbound op Windows: extra verbruik DNSSEC: 1919,8 Joule¹¹
- BIND op Windows: extra verbruik DNSSEC: 226,45 Joule
- Unbound op Linux: Extra verbruik DNSSEC: 527,85 Joule
- BIND op Linux: extra verbruik DNSSEC: 753,54 Joule

'Kan het aanpassen van de configuratie van een of meer van de pakketten leiden tot een efficiënter gebruik van energie?'

Zoals eerder genoemd was deze deelvraag optioneel en zou alleen in behandeling genomen worden als daar tijd voor was. Helaas is door tijdgebrek deze deelvraag niet in behandeling genomen. De deelvraag is daarom terug te vinden in toekomstig werk.

'Is het efficiënter om resolver functionaliteit te verplaatsen naar een hoger niveau in het netwerk?'

Zoals eerder genoemd was deze deelvraag optioneel en zou alleen in behandeling genomen worden als daar tijd voor was. Helaas is door tijdgebrek deze deelvraag niet in behandeling genomen. De deelvraag is daarom terug te vinden in toekomstig werk.

¹¹ Onbetrouwbare data

8 Aanbevelingen

8.1 Toekomstig werk

Er zijn een aantal onderdelen waar niet aan is toegekomen vanwege tijdgebrek. In dit hoofdstuk worden een aantal punten genoemd die misschien in de toekomst nog gemeten kunnen worden.

8.1.1 Configuratie

Een van de deelvragen ging over het effect van het aanpassen van de configuratie. Denk hierbij aan bijvoorbeeld het aanpassen van de cache size, het aanpassen van de TTL etc. Dit zijn dingen die effect kunnen hebben op het energieverbruik. Om een interessant voorbeeld te noemen, stel dat www.google.nl een TTL instelt van 30 seconden, dan zou de cache entry voor www.google.nl na 30 seconden verlopen in elke resolver over de hele wereld. Dat zijn er zo'n 10 miljoen.¹² Wat voor invloed zou het hebben als al die 10 miljoen resolvers na 30 seconden opnieuw www.google.nl moeten resolven?

8.1.2 IPv6

Helaas was er geen IPv6 aanwezig in het SEFlab, de impact van IPv6 is dan ook niet gemeten op de huidige opstelling. In de toekomst zou er eventueel gekeken kunnen worden naar een dataset met alleen maar IPv6-queries tegenover een dataset met alleen maar IPv4-queries.

8.1.3 Besturingssystemen

De huidige keuze in besturingssystemen was beperkt. Er zijn bijvoorbeeld een aantal Linux distributies die nog gemeten kunnen worden. Het baseline verbruik van CentOS ligt hoger dan dat van Windows, maar misschien is er ook wel een besturingssysteem met een lager baseline verbruik dan dat van Windows. Het is interessant om te meten wat de verschillen kunnen zijn bij het gebruik van een ander besturingssysteem. Voorbeelden zijn WattOS (gebouwd op energiezuinigheid), Debian of Ubuntu server

8.1.4 Hardware

Een zeer belangrijk onderdeel bij het meten is de hardware. Het energieverbruik is onlosmakelijk verbonden aan de soort hardware die je gebruikt. Op Wikipedia is bijvoorbeeld een lijst samengesteld uit verschillende bronnen met het maximale stroomverbruik van allerlei CPU's.¹³ Het is interessant om te weten hoe de DNS-software zich gedraagt op verschillende hardware-architecturen. Nieuwe hardware wordt steeds kleiner en zuiniger, wat voor effect zou dat kunnen hebben op het energieverbruik van DNS-resolvers?

8.1.5 Verplaatsen resolver

De deelvraag '*Is het efficiënter om resolver functionaliteit te verplaatsen naar een hoger niveau in het netwerk?*' is niet beantwoord tijdens dit onderzoek. Dit zou een interessant onderwerp kunnen zijn voor een vervolgonderzoek.

8.1.6 Unbound op Windows

Het gedrag van Unbound op Windows is onverwacht en onbetrouwbaar. Het is interessant om dit gedrag te analyseren en uit te vinden waar het vandaan komt. Over de resultaten van de metingen van Unbound op Windows is al contact opgenomen met NLnet Labs. Het is daar in behandeling.

¹² Bron [11]: Ten Million DNS resolvers (22 Maart 2012). Icann. <http://blog.icann.org/2012/03/ten-million-dns-resolvers-on-the-internet/>

¹³ Bron [12]: List of CPU Power (darum onbekend). Wikipedia. http://en.wikipedia.org/wiki/List_of_CPU_power_dissipation_figures

9 Persoonlijke reflectie en ontwikkeling

Dit hoofdstuk gaat over mijn persoonlijke ontwikkeling tijdens dit project. De ontwikkeling van de competenties die genoemd zijn op het aanvraagformulier worden hier behandeld en daarna komt mijn persoonlijke reflectie aan bod.

9.1 Competenties

De competenties genoemd op het aanvraagformulier zijn als volgt:

1. S.An3 (Software, analyseren): Een requirement analyse uitvoeren voor een complex softwaresysteem in een context van diverse bestaande systemen met verschillende belanghebbenden.
2. I.Ad3 (Infrastructuur, Adviseren): Kunnen toepassen van een brede argumentatie vanuit techniek, bedrijfsprocessen, kosten/baten, risico's en wetgeving.
3. I.Re3 (Infrastructuur, Realiseren): Een systeem van customized oplossingen (niet-standaard oplossingen) voorbereiden, uitrollen en testen.
4. Alle taakgerichte (T1, T2, T3), relatiegerichte (R1, R2, R3, R4) en zelfgerichte (Z1, Z2) competenties
 - a. T1 Planmatig werken: De medewerker is in staat om op adequate wijze zijn eigen werkzaamheden planmatig te verrichten
 - b. T2 Projectmatig werken: De medewerker is in staat om op adequate wijze binnen een projectmatige structuur te werken
 - c. T3 Opdrachten toetsen: De medewerker is in staat om op adequate wijze om te gaan met zijn eigen verantwoordelijkheden en bevoegdheden
 - d. R1 Communiceren: De medewerker is in staat om op adequate wijze schriftelijk en mondelijk te communiceren
 - e. R2 Samenwerken: De medewerker is in staat om op adequate wijze met anderen samen te werken
 - f. R3 Omgaan met anderen: De medewerker is in staat om op adequate wijze om te gaan met de belevingswereld van anderen
 - g. R4 Relaties onderhouden: De medewerker is in staat om op adequate wijze relaties met anderen te onderhouden
 - h. Z1 Reflecteren: De medewerker is in staat om op adequate wijze de effectiviteit en de efficiëntie van zijn eigen handelen te bepalen en bij te stellen
 - i. Z2 Bijleren: De medewerker is in staat om op adequate wijze nieuwe ontwikkelingen te volgen en toe te passen

9.1.1 S.An3(Software, analyseren)

Aan de hand van de opdracht zijn er een onderzoeksplan en meetplan opgesteld. Na het uitvoeren van de metingen zijn de resultaten geanalyseerd om de deelvragen te kunnen beantwoorden. Voor het beantwoorden van de deelvragen moesten de metingen aan bepaalde requirements voldoen. Zo was het belangrijk dat de baseline van het besturingssysteem werd gemeten, maar ook het verbruik van het pakket gedurende verschillende stadia. Het vullen van de cache kon namelijk een ander verbruiksresultaat opleveren in vergelijking met cache hits. Daarna zijn de verschillende pakketten met elkaar vergeleken om tot een conclusie te komen.

De bovenstaande situatie sluit niet volledig aan op de competentiebeschrijving. Maar de competentiebeschrijving sluit ook niet volledig aan bij het project. Ik ben dan ook van mening dat ik deze competentie niet genoeg heb kunnen ontwikkelen. Daarom ben ik van plan in te toekomst nog te gaan werken aan de ontwikkeling van deze competentie.

9.1.2 I.Ad3(Infrastructuur, Adviseren)

Deze competentie wordt bewezen aan de hand van het onderzoeksrapport dat wordt opgeleverd aan SURFnet. Het onderzoeksrapport zal een advies bevatten dat beargumenteert is vanuit de techniek, kosten/baten en beveiligingsrisico's (DNSSEC). Punten uit dat verslag die belangrijk zijn voor deze competentie zijn:

- Kosten/baten van het kiezen voor een bepaald DNS pakket, de hoeveelheid energie en geld die bespaard kan worden door deze keuze en wat de keuze dus uiteindelijk op zal leveren
- De beveiligingsrisico's die het niet toepassen van DNSSEC met zich meebrengt en de kosten in stroomverbruik die het wel toepassen van DNSSEC met zich meebrengt. Een duidelijke afweging tussen de voor en nadelen is hier ook onderdeel van

9.1.3 I.Re3 (Infrastructuur, Realiseren)

Gedurende dit project is er gewerkt met een infrastructuur die voortdurend in ontwikkeling was en getest werd. De complete meetopstelling werd constant voorbereid en opnieuw getest, en soms aangepast. In samenwerking met het SEFlab zijn veel van die tests uitgevoerd om de juistheid en nauwkeurigheid van de testopstelling te kunnen garanderen. Daarnaast is de DNS resolver gebouwd met 2 besturingssystemen, en 3 DNS pakketten op 1 server om de combinaties van die pakketten en besturingssystemen te kunnen testen.

9.1.4 Taakgericht, relatiegericht en zelfgericht

- T1, Planmatig werken
 - In de eerste weken van het project is er een planning opgesteld te zien in tabel 1. Het schrijven van documentatie en het uitvoeren van de metingen is grotendeels volgens de planning verlopen. Enige uitzondering is het uitvoeren van de metingen. Dit ging een week langer door omdat de metingen langer duurde dan in eerste instantie ingeschat.
- T2, Projectmatig werken
 - Het projectmatig werken is in vorige projecten al meerdere malen bewezen. Tijdens dit project ging de ontwikkeling van de competentie vooral om het projectmatig werken zonder projectteam. Het zelf plannen/bewaken van de planning, en het zelf uitvoeren van projectmatig werk is verbeterd.
- T3, Opdrachten toetsen
 - In eerste instantie was het moeilijk om om te gaan met de eigen verantwoordelijkheid zonder veel sturing. Ik heb een grote ontwikkeling doorgemaakt in het zelfstandig opstellen van planningen en het zelfstandig uitvoeren van een project. Gezien de huidige staat van het project en de geproduceerde resultaten ben ik van mening dat ik deze competentie voldoende heb ontwikkeld. Er kan in de toekomst echter zeker nog een verbeterslag gemaakt worden. Feedback van SURFnet gaf aan dat ik soms te snel wilde gaan en een stapje terug moest doen. Daarnaast had ik wat meer open moeten staan voor suggesties vanuit SURFnet. Dit hoop ik in de toekomst te verbeteren.
- R1, Communiceren
 - Er vonden regelmatig voortgangsbesprekingen plaats met Albert en Roland gedurende het project. Tijdens deze gesprekken werd er op een relaxte en informele sfeer besproken hoe het project er voor stond en welke richting ik verder uit ging. Daarnaast is er veel contact via de mail geweest. Ook heb ik regelmatig met Roland gecommuniceerd via een chat-client, bijvoorbeeld als er

een probleem was met een van de DNS-systemen waar ik in mijn eentje niet snel uit kon komen.

- R2, Samenwerken
 - De ontwikkeling van de MATLAB-code vond plaats in samenwerking met Martin Droog, op dat moment was Martin in dienst voor het SEFlab. Samen met Martin en Eric hebben we een opzet gemaakt voor de MATLAB code. Waarna de code ook ontwikkeld is. Tevens heb ik veel samengewerkt met Eric gedurende het project om de meetopstelling te verbeteren. Door deze samenwerking is mijn basiskennis van elektrotechniek ook vooruit gegaan.
- R3, Omgaan met anderen
 - Tijdens de stage heeft het SEFlab mij gevraagd om tweedejaarsstudenten van de informatica opleiding een beetje te begeleiden tijdens hun gebruik van de meetopstelling. Dit omdat ik in korte tijd veel ervaring opgebouwd had met de meetopstelling. Het was voor mij even een omschakeling, omdat ik gewend was met mensen samen te werken die veel van de opstelling afwisten. De tweedejaarsstudenten hadden echter nog nooit met de opstelling gewerkt dus dingen die voor mij logisch waren wisten zij niet. Nadat ik dit door had was het een stuk makkelijker voor mij om de studenten te helpen.
- R4, Relaties onderhouden
 - Gedurende de stageperiode heb ik regelmatig contact gehad met mensen waar ik in het verleden mee heb samengewerkt. Door vorige samenwerkingsverbanden is het makkelijker om ingewikkelde concepten met elkaar te bespreken en ideeën met elkaar te delen. Daarnaast heb ik ook af en toe contact gehad met mijn vorige stagebegeleider (Anand Buddhdev, RIPE NCC) bij voorbeeld een DNS-probleem waar ik niet uit kwam. Op dat moment was mijn huidige stagebegeleider niet aanwezig.
- Z1, Reflecteren
 - Tijdens dit project zijn er een aantal momenten geweest waarbij er aanpassingen gedaan moesten worden aan de voortgang. Een concreet voorbeeld zijn de meetprotocollen. De vooraf opgestelde meetprotocollen bleken na enkele tests niet efficiënt genoeg om alle metingen binnen de vastgestelde tijd uit te voeren. De resultaten van de metingen voldeden wel aan de eisen, maar het uitvoeren van de beschreven metingen op alle pakketten zou te lang duren. Daarop is er tijdens een gesprek met de opdrachtgever besloten om de meetprotocollen aan te passen. In samenwerking met het SEFlab is er toen een nieuw meetprotocol opgesteld.
- Z2, Bijleren
 - DNSSEC was voor mij een nieuwe technologie, ik had er van gehoord maar nog nooit mee gewerkt. Tijdens de stageperiode heb ik DNSSEC configuraties gemaakt voor BIND, Unbound en Microsoft DNS en heb deze ook grondig getest. De meetsystemen bij het SEFlab waren ook nieuw voor me. Ik heb binnen korte tijd bekendheid met deze meetsystemen opgebouwd om er voor te zorgen dat ik begreep wat er gebeurde en wat de meetsystemen voor resultaten konden produceren. Ik had nog nooit gewerkt met MATLAB, dus het schrijven van code in MATLAB ging me wat moeilijk af. Gelukkig kreeg ik veel hulp van Martin bij het gebruik van MATLAB. Tevens is mijn algehele DNS kennis groter geworden en dan met name de kennis over BIND, Unbound en Microsoft DNS.

9.2 Bedrijfsfeedback

In deze paragraaf wordt de feedback van SURFnet en het SEFlab beschreven, ik ga verder in op de feedback in de Reflectie paragraaf.

9.2.1 SURFnet

Er vonden regelmatig voortgangsbesprekingen plaats. In eerste instantie was dat wekelijks, maar later werd dat minder omdat ik vooral aan het werk was in Amsterdam. Tijdens die voortgangsbesprekingen werd er besproken hoe ver ik was en welke richting ik verder op ging. Het was dan ook een informeel feedbackmoment. Daarnaast heeft er nog een functioneringsgesprek plaatsgevonden. Ook dit is op een informele manier gebeurd. Persoonlijk vond ik die informele werkomgeving prettig. De feedback die ik kreeg tijdens het functioneringsgesprek kwam neer op het volgende:

- Roland en Albert zijn blij hoe ik het oppak, met mijn enthousiasme, doorzettingsvermogen en ambitie om het goed te doen;
- Tegelijkertijd zijn bovenstaande punten ook een valkuil, Roland en Albert gaven aan dat ik soms te snel wil en adviseerde me af en toe een stapje terug te nemen;
- Roland en Albert gaven aan dat mijn eigenwijsheid goed is, maar dat ik soms meer moet open staan voor suggesties die ze doen;
- Roland en Albert gaven aan er alle vertrouwen in te hebben dat ik uiteindelijk een mooi resultaat oplever.

Daarnaast kreeg ik nog wat algemene feedbackpuntjes over mijn rapportage:

- Probeer de 7-word-rule te gebruiken in documenten. Dat houdt in dat binnen 7 woorden de hoofdzaak van een zin bereikt moet zijn;
- Kijk kritisch naar wat je opschrijft en vraag jezelf af, zijn deze details wel nodig hier? Treed niet meteen in teveel detail.

9.2.2 SEFlab

Tegen het einde van de stageperiode heb ik ook Eric van het SEFlab om feedback gevraagd, zijn feedback was als volgt.

Remy Bien is een gemotiveerde student zich serieus inzet voor zijn afstuderen. Het is prettig samenwerken met Remy. Als leerpunten wil ik Remy meegeven dat hij meer mag relativeren wanneer iets niet gaat zoals hij verwacht en dat hij meer buiten zijn comfort zone durft te opereren.

9.3 Reflectie

Een van de grootste uitdagingen voor mij was het uitvoeren van dit project in mijn eentje. Gedurende mijn studie heb ik altijd in projectteams gewerkt. En vooral de laatste twee jaar was de teamopbouw grotendeels hetzelfde. We wisten wat we van elkaar konden verwachten, en wie waar goed in is. Als je dan ineens een project in je eentje gaat doen heb je niet meer je teamleden om op terug te vallen. Dat was even een flinke omschakeling voor mij en kostte wat tijd om aan te wennen. Het is een hele nieuwe ervaring om in je eentje een compleet project te doen, maar ik zou liever projecten doen in teamverband.

Een andere uitdaging voor mij was de mate van vrijheid die ik kreeg bij deze opdracht. Dit hangt ook samen met het vorige punt, werken in teamverband was er niet bij. In het verleden bepaalde we altijd samen wat de grenzen waren van de opdrachten die we uitvoerden. Als iemand een onrealistisch beeld had om wat voor reden dan ook werd dat redelijk snel opgevangen door een van de andere teamleden. Ik heb het idee dat ik er wel redelijk mee om ben gegaan, maar ook dat

was iets waar ik flink aan moest wennen. Als er punten voor mij waren waarvan ik niet wist wat ik er mee aan moest kon ik wel altijd bij Roland of bij Eric terecht. Al met al heb ik het idee dat het project netjes gescoped en uitgevoerd is.

10 Bronnenlijst

[1]Bulsuk, Karn (2 Februari, 2009). Taking the First Step with the PDCA (Plan-Do-Check-Act) Cycle. <http://www.bulsuk.com/2009/02/taking-first-step-with-pdca.html>

[2]Internet Systems Consortium (datum onbekend). BIND, The most widely used Name Server Software. <https://www.isc.org/downloads/bind/>

[3]Moore, Don (23 Mei, 2004). DNS Server Survey. <http://myDNS.bboy.net./survey/>

[4]AppNeta, (December, 2013). Tcp replay, Pcap editing & replay tools voor *NIX. <http://tcpreplay.synfin.net/>

[5]Goldberg, David (Maart 1991). What Every Computer Scientist Should Know About Floating-Point Arithmetic. http://docs.oracle.com/cd/E19422-01/819-3693/ncg_goldberg.html

[6]RapidTables, (datum onbekend). Joules to Watts calculator. http://www.rapidtables.com/calc/electric/Joule_to_Watt_Calculator.htm

[7]RapidTables, (datum onbekend). Joules to kWh conversion. http://www.rapidtables.com/convert/energy/Joule_to_kWh.htm

[8]Bhosted, (datum onbekend). Bhosted.nl rackspace. <http://www.bhosted.nl/colocatie/rackspace>

[9]Base IP (datum onbekend). Base IP Network serviceprovider. <https://www.baseip.com/8-diensten/9-colocatie.html>

[10]Bitency (datum onbekend). Bitency Colocatie. <https://bitency.nl/colocatie>

[11] Ten Million DNS resolvers (22 Maart 2012). Icann. <http://blog.icann.org/2012/03/ten-million-dns-resolvers-on-the-internet/>

[12] List of CPU Power (darum onbekend). Wikipedia. http://en.wikipedia.org/wiki/List_of_CPU_power_dissipation_figures

[13]Nijssen, Emile (14 mei 2014). Overzicht PCB's SEFlab. Bijlage I

11 Bijlagen

Bijlage I: Overzicht PCB's SEFlab¹⁴

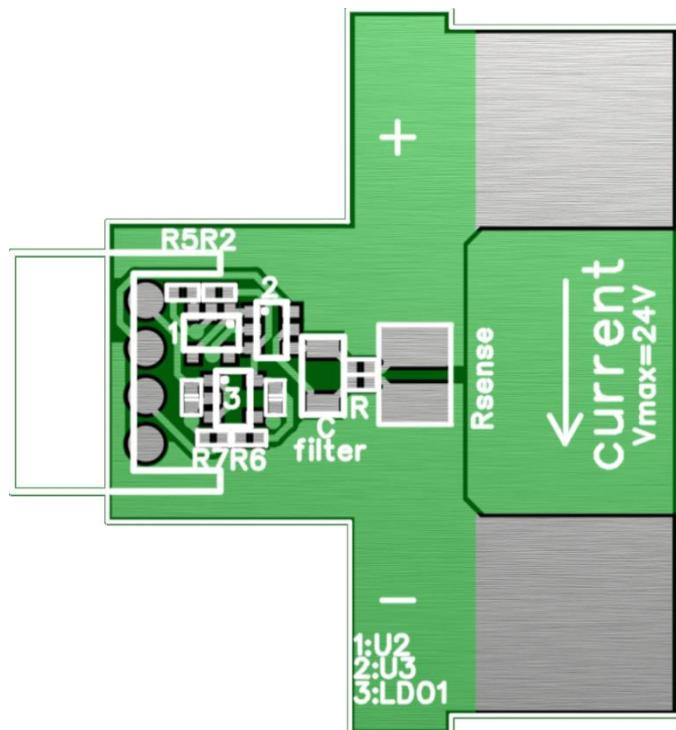
Deze bijlage is onderdeel van de documentatie van het SEFlab, en is toegevoegd aan dit document als naslagwerk. Niet alle bordjes genoemd in dit document zijn gebruikt tijdens dit project.

Overzicht PCBs SEFlab

In dit document bevinden zich alle printplaten die in het SEFlab worden gebruikt. Per product is er een korte uitleg voor revisie 0. Bij de daaropvolgende revisies zijn alleen de wijzigingen genoemd.

I0 - Stroommeet-PCB met versterker - revisie 0

Productie	10x
Serienummers	I0-1 tot I0-10



Tot de ontwikkeling van deze PCB gebruikte het SEFlab meetweerstanden die met een National Instruments DAQ uitgelezen werden. De weerstanden hadden weliswaar Kelvin-aansluitingen (om zo het effect van spanningsval over de meetaansluitingen uit te sluiten), maar er zat nog steeds een erg lange draad tussen de meetweerstand en DAQ. Het meten van zeer kleine spanningen door een lange draad heen is een recept voor interferentie. De oplossing is om zo dicht mogelijk bij de meetweerstand een buffer en/of versterker te plaatsen om zo een laagimpedant, voldoende groot in amplitude meetsignaal aan de DAQ aan te kunnen bieden.

Dit ontwerp doet precies dat. Een INA214 50V/V stroommeetversterker zit via een 1^e orde R-C filter aan de stroommeetweerstand vast, en de output van deze versterker gaat direct naar de DAQ. Het stroommeetcircuit wordt lokaal met een 3.3V LDO van stabiele voedingsspanning voorzien. Ook zit aan de REF-ingang van de INA214 een 50mV gebufferde referentiespanning om de INA214 in zijn lineaire gebied te krijgen (zie INA214 datasheet).

Als meetweerstand is de Welwyn ULR3 gekozen. Dit is een metaalstripweerstand met vrijwel geen zelfinductie, ook niet door fysieke grootte. Dit maakt het circuit ook bruikbaar to hoge frequenties (ca. 1MHz, hoewel de INA214 dit niet kan bijbenen). Het dissipierbaar vermogen is 3W, wat voldoende is om tot 60A te meten (met een 0,5 mohm variant).

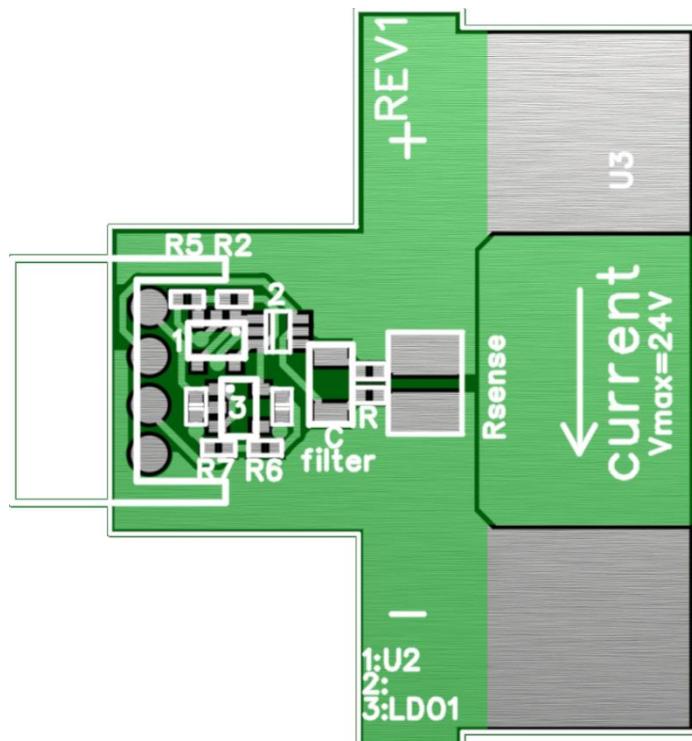
¹⁴ Bron [13]

Deze meetweerstand heeft wel een aanzienlijke temperatuurcoëfficient, maar deze is vanwege de opbouw van de weerstand zeer voorspelbaar en dus weg te kalibreren.

Als *design for testing* is de printplaat ontworpen om in een Molex Poweredge Xtreme 3-polige connector te passen. Dit maakt dat de printplaten gemakkelijk in en uit een meetsysteem te halen zijn, of bijvoorbeeld een kalibratie- of verificatieopstelling. Molex Poweredge is gekozen vanwege de hoge maximale stroomcapaciteit, geen andere eenvoudig toe te passen connector van een redelijk formaat was beschikbaar voor dit doel.

I1 - Stroommeet-PCB met versterker - revisie 1

Productie	10x
Serienummers	I1-1 tot I1-10



Verholpen in deze revisie:

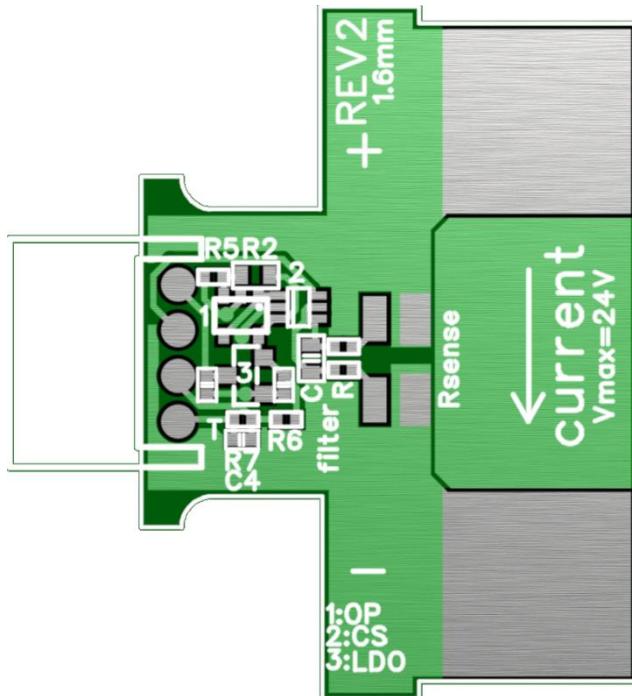
1. INA214 footprint van SOT23-6 naar SC-70-6
2. Filterweerstanden, andere weerstanden verder uit elkaar i.v.m. soldeerbaarheid
3. Revisie-tekst toegevoegd

Errata:

1. Silkscreen dient aan achterkant aangebracht te worden met stroomrichting, revisie
2. Geen ontwerpdatum op silkscreen
3. XC6223 slecht verkrijgbaar, dient vervangen te worden door XC6206
4. U3 silkscreen verkeerd

I2 - Stroommeet-PCB met versterker - revisie 2

Productie	20x
Serienummers	I2-1 tot I2-20



Verholpen in deze revisie:

Schema

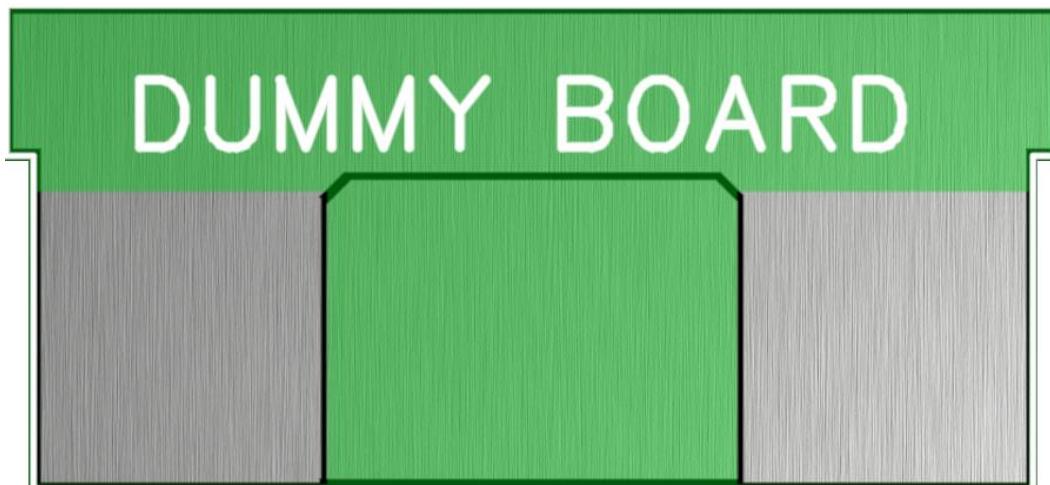
- INA213 ipv INA214 (vanaf nu hebben alle bordjes INA213)
- Nauwkeurigere 50mV offset, dit is nu by design 50mV +/- (0.1% + 250µV) (was origineel 50mV +/- (2% + 5mV))
- vanwege bovenstaande: MCP606 buffer-versterker voor de offset ipv MCP6L01
- Weerstand-footprint heeft nu kelvin-contacten en juiste footprint (niet meer te klein)
- Filter is nu identiek aan alle andere printjes
- LDO heeft een update naar het nieuwe model gekregen dat ook op alle andere bordjes wordt gebruikt
- Filtercondensator op de thermistor

PCB

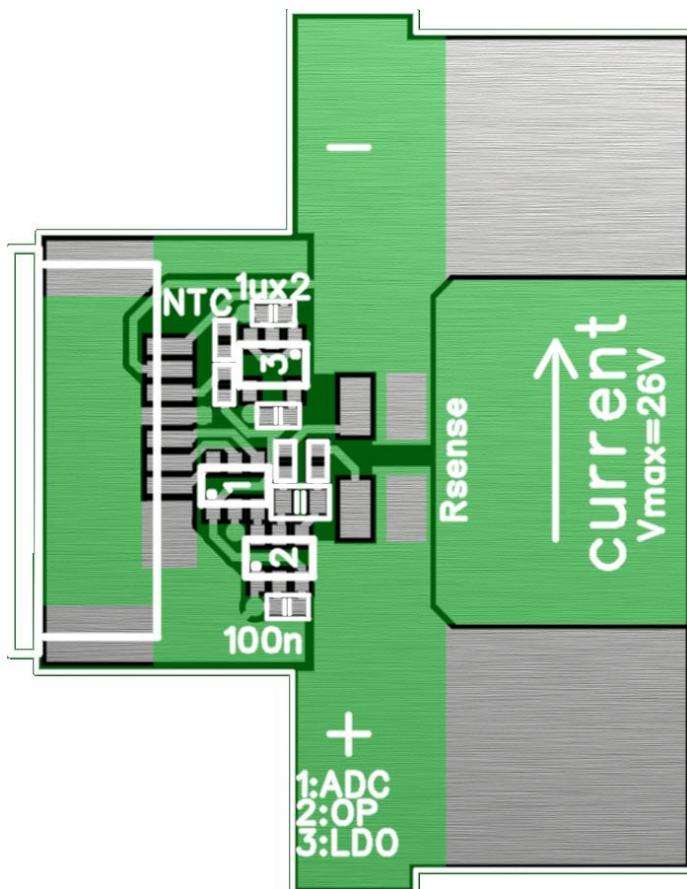
- Update van een aantal footprints zodat het makkelijker te solderen is
- PCB iets kleiner gemaakt
- 'lipjes' op de PCB gemaakt zodat hij gemakkelijker uit de Poweredge-connector te trekken is (suggestie Martin)
- Silkscreen overal geüpdate volgens protocol (revisie aan beide kanten, PCB-dikte in silkscreen, stroomrichting aan beide kanten, pinout van de connector op silkscreen)

Productie	10x
Serienummers	D_0-1 tot D_0-10

Deze PCB is niets anders dan een harde doorverbinding tussen de gebruikte polen van de Molex Poweredge voor plekken waar meting niet gewenst is of wanneer een computer zonder meetinstrumenten moet draaien.



Productie	10x
Serienummers	AD_0-1 tot AD_0-10



Als doorontwikkeling op de stroommeet-PCB met versterker is nu ook de ADC zo dicht mogelijk bij de meetweerstand geplaatst, zodat interferentie in het geheel geen vat meer krijgt op het signaal. Ook is de connector vervangen door een flatcable. Dit omdat meer polen nodig waren voor de extra signalen, en andere types connectoren te groot zouden worden. Deze printplaat is al vrij groot om gemakkelijk weg te kunnen werken in computers, we willen het liever nog kleiner maken dan groter.

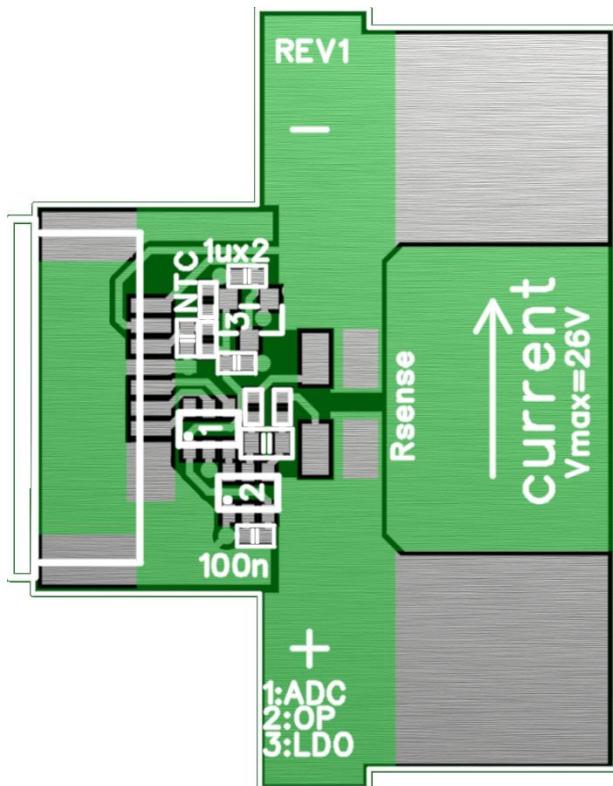
Naast het digitale stroomsignaal wordt ook de spanning en temperatuur analoog doorgegeven op de FFC (dit zijn trage signalen die maar enkele keren per seconde hoeft te worden gemeten).

Door het compactere ontwerp en het vervallen van de noodzaak voor een NI DAQ kan nu ook een veel grotere resolutie, zowel temporaal als in bit-diepte, worden gehaald. Momenteel is dit ontwerp voorzien van een ADS7886 12-bit, 1MSPS ADC. Echter, elke ADC uit de TI ADS7xxx, TI ADCS7xxx of ADI AD7xxx-serie kan op dit bord worden gesoldeerd. Dit maakt performance van 8-bit 1MSPS tot 16-bit 3MSPS mogelijk, gemakkelijk genoeg om 1-1.5MHz analoge bandbreedte te behalen met een 1^e orde R-C filter.

Om deze frequenties ook daadwerkelijk te kunnen meten is de stroommeetversterker vervangen door een ADM4073F (in plaats van de INA214). In het huidige ontwerp is het R-C filter echter de beperkende factor, deze is op ongeveer 10kHz -3dB getuned.

AD_1 - Stroommeet-PCB met ADC - revisie 1

Productie	20x
Serienummers	AD_1-1 tot AD_1-20

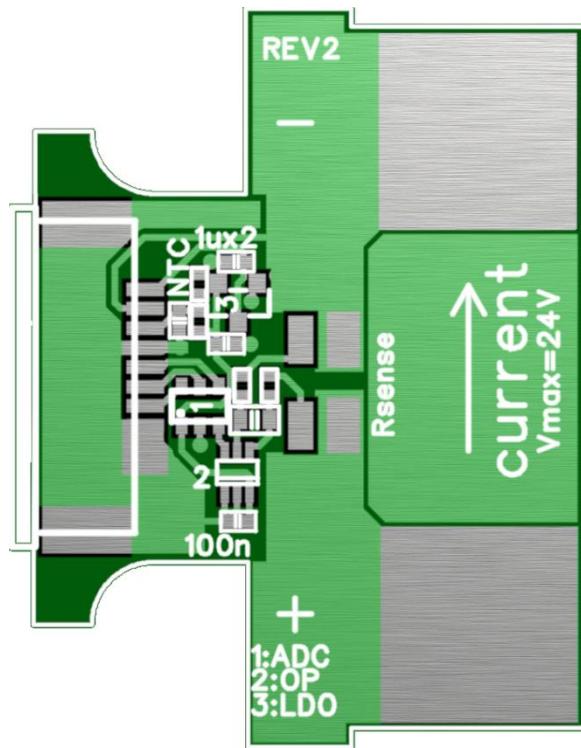


Verholpen in deze revisie:

1. XC6223 vervangen door XC6206
2. 1 μ F condensator parallel aan NTC geplaatst
3. Silkscreen revisie-tekst en stroomrichting aan beide kanten van PCB ipv enkelzijdig gemaakt
4. Ontwerpdatum toegevoegd

AD_2 Stroommeet-PCB met ADC - revisie 2

Productie	10x
Serienummers	AD_2-1 tot AD_2-10

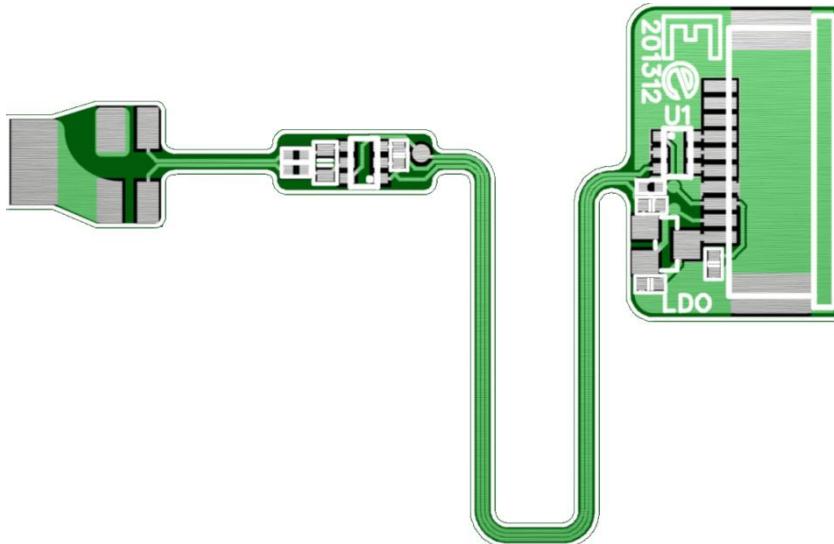


Verholpen in deze revisie:

- INA213 ipv ADM4073F
- Nokjes op de PCB om hem makkelijker uit de connector te kunnen trekken
- Silkscreen update
- Filter geüpdatedet

FL_0 - Flexibele stroommeet-PCB met ADC - revisie 0

Productie	10x
Serienummers	FL_0-1 tot FL_0-10
Let op:	0.15mm flex-PCB



Deze printplaat is in feite zoals de andere digitale meetbordjes, maar dan gemaakt van een 0.15mm dik flexibel mylar-substraat. Hierdoor is het flexibel en kan het in hoekjes en gaatjes worden gestopt waar de vorige niet-flexibele versies niet pasten. Helemaal links zit zichtbaar aan de bovenkant één pool van de meetweerstand, aan de andere zijde van deze pad zit de andere kant. De installatie van deze printplaat is dus als volgt:

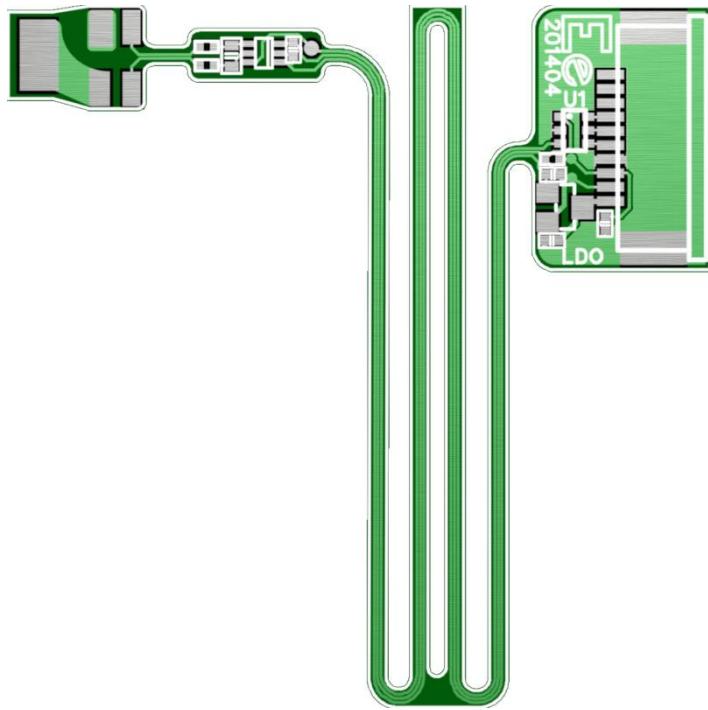
- Desoldeer een component waar de stroom doorheen moet worden gemeten (bijv. Inductor, MOSFET)
- Soldeer de flex-PCB op één van de pads van het onderdeel
- Soldeer het component weer terug op zijn plek (maar nu zit de meet-PCB ertussen)

Op deze manier kan ook stroom door actieve spoelen heen worden gemeten zonder de eigenschappen van de spoel aanzienlijk te veranderen. Er is genoeg flexibiliteit tussen de soldeerpads en de meetweerstand om de PCB op die plek om te buigen om over andere componenten heen te gaan. Ook zit de NTC op deze printplaat achterop de meetweerstand. De stroommeetversterker zit ongeveer 1cm vanaf de meetweerstand – ver genoeg om voldoende flexibel te zijn om bijv. 90 graden te kunnen buiten, maar dichtbij genoeg om geen last te hebben van interferentie. Achter nog eens 5cm flexibel PCB zit de ADC, stroomvoorziening en connector. Op de achterzijde van dit deel kan dubbelzijdig tape worden geplakt om de connector ergens aan vast te maken waar het geen kwaad kan.

Dit ontwerp laat een bijna ideale stroomloop toe, veel beter dan eerdere ontwerpen, maar het is moeilijk om een effectieve kalibratie-opstelling hiervoor te ontwerpen. Dit is dan ook nog een *work in progress*. Nieuwere revisies lossen dit mogelijk op.

FL_1 - Flexibele stroommeet-PCB met ADC – Revisie 1

Productie	40x
Serienummers	FL_1-1 tot FL_1-40
Let op:	0.15mm flex-PCB



Verholpen in deze versie:

Schema

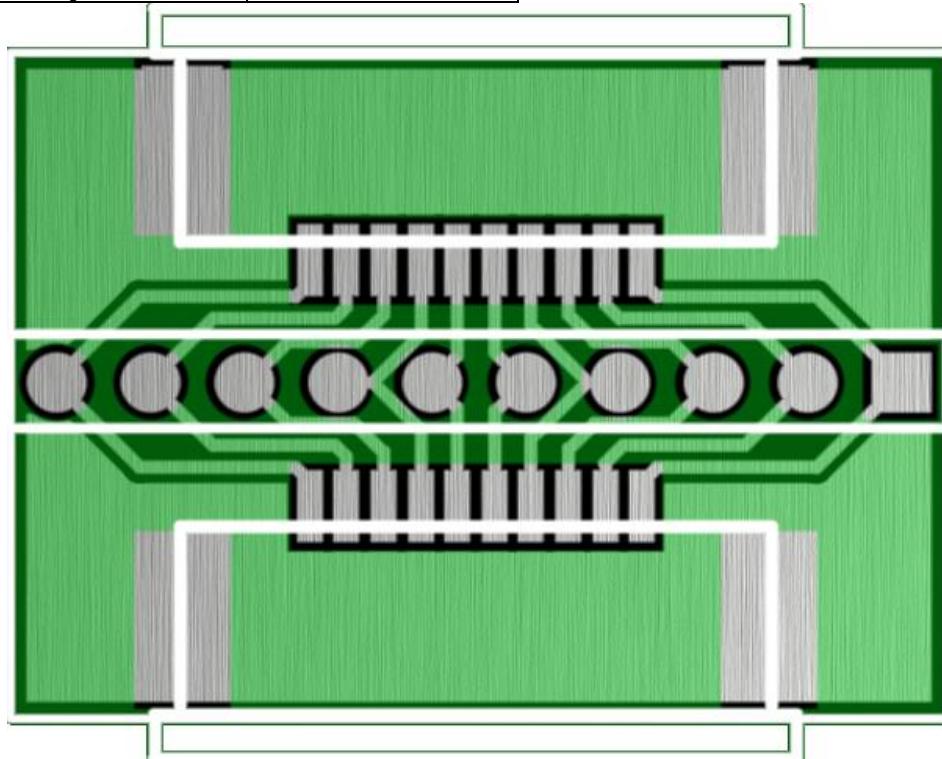
- INA213 ipv ADM4073F
- Filter geüpdateet
- Kleine tweaks in het schema

PCB

- ca. 2,5x zo lang flexibel deel
- Meer ruimte rondom aansluitpunten gelaten
- PCB_vorm zodanig dat hij met meerderen in een paneel kan worden geproduceerd (in dit geval: met 4 bij elkaar op een 10x10cm PCB, dan krijgen we er 40 voor \$100 ipv 10 stuks voor \$85)

FF_0 -FFC breakout/verlengbordje - revisie 0

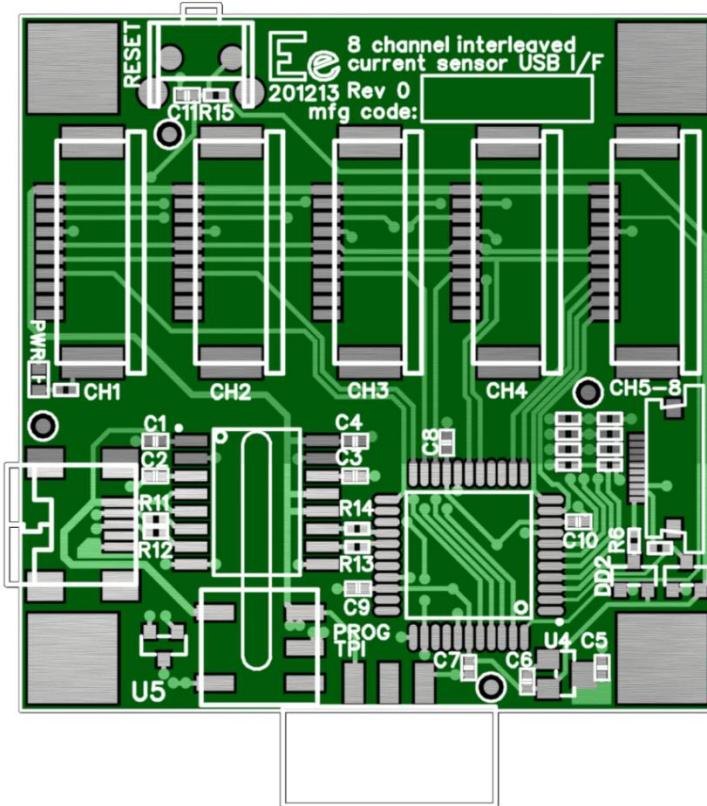
Productie	10x
Serienummers	FF_0-1 tot FF_0-10
Let op:	0.8mm PCB-dikte



Deze printplaat dient om de 1mm 10-polige FFC kabel die gebruikt wordt in alle digitale ontwerpen om te zetten naar een standaard 2,54mm pitch header. Het bordje kan ook worden gebruikt om een FFC-kabel te verlengen.

UD_0 - 4+4-kanaals USB-naar-digitaal stroommeetbord - revisie 0

Productie	2x
Serienummers	UD_0-1 tot UD_0-2

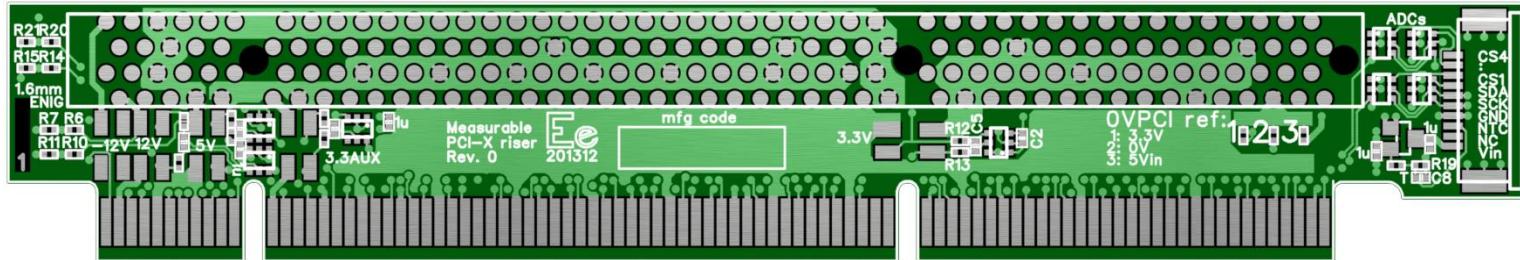


Deze printplaat accepteert 4 standaard digitale stroommeetbordjes + één 4-kanaals bord (bijvoorbeeld: PCI riser-bord). Alle kanalen delen dezelfde SPI-poort voor het uitlezen, dus de uitleessnelheid gaat proportioneel omlaag wanneer meerdere bordjes erop worden aangesloten. De gebruikte microcontroller (ATXMega128A4U) gebruikt de populaire LUFA-library om zichzelf als USB-CDC (virtuele seriële poort) aan te dienen bij de host-computer. Hiervoor dient op de computer de bijgeleverde driver (LufaVirtualSerial.inf) geïnstalleerd te worden. Let op: onder Windows 8 vereist dit het herstarten van de computer in een modus waar niet-ondersteunde drivers mogen worden geïnstalleerd. Vervolgens kan een willekeurige seriële terminal (aanbevolen: Termite) gebruikt worden om het bord uit te lezen.

Het bord bevat volledige galvanische isolatie van USB data en power, dus het kan gebruikt worden zonder isolatietransformator.

PCIX_0 - PCI-X meetbaar riser-bord – revisie 0

Productie	2x
Serienummers	PCIX_0-1 tot PCIX_0-2

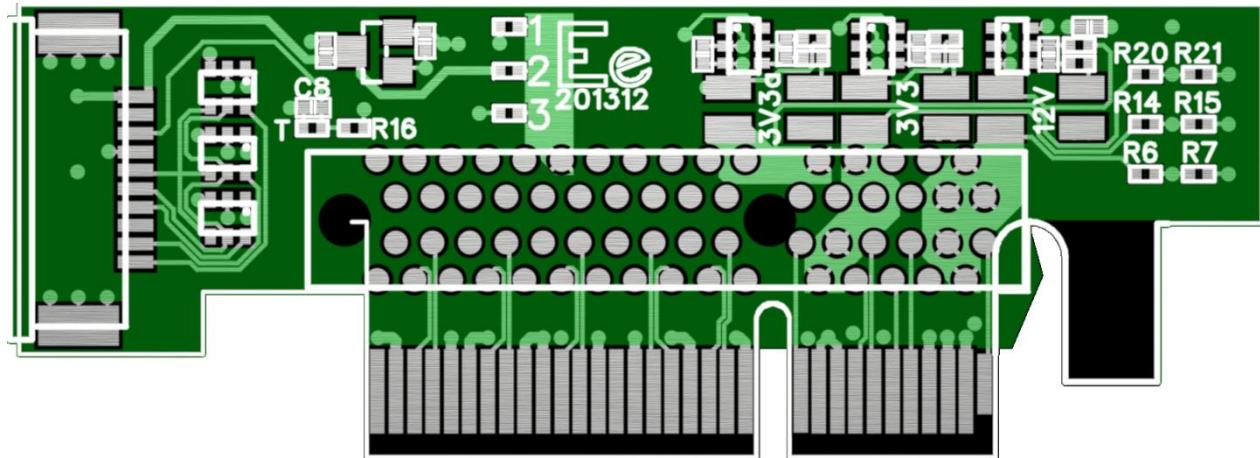


Deze printplaat vervangt de bestaande PCI-X (64-bits PCI) riser in de SUN meetcomputer en voegt meetweerstanden, versterkers en ADCs toe aan het bord. Het is in feite een riser-bord met 4 revisie 1 digitale meetbordjes erop geïntegreerd. De bemeten spanningen zijn: 12V, 5V, 3.3V en 3.3Vaux (standby-voeding). De -12V-lijn, die ook aanwezig is op PCI-X, is voorzien van een footprint voor een meetweerstand maar geen versterker of ADC. Indien gewenst kan hier via een aantal testpunten op de achterkant een analoge DAQ op worden aangesloten. Echter, in 99% van de gevallen is de -12V geheel ongebruikt.

De uitgang is een FFC net als de digitale meetbordjes, maar zonder spanningsmeting en met extra selectielijnen voor de 4 ADCs.

PCIEx4_0 - PCI Express x4 meetbaar riser-bord – revisie 0

Productie	2x
Serienummers	PCIEx4_0-1 tot PCIEx4_0-2



Deze printplaat is hetzelfde idee als het PCI-X meetbaar riser-bord, maar dan voor PCI-Express x4. Op dit bord wordt 12V, 5V en 3.3V bemeten.

Dit bord heeft dezelfde uitgang als het PCI-X bord, maar met slechts 3 ipv 4 selecteerbare ADCs.

PCIEx16_0 - PCI Express x16 meetbaar riser-bord - revisie 0

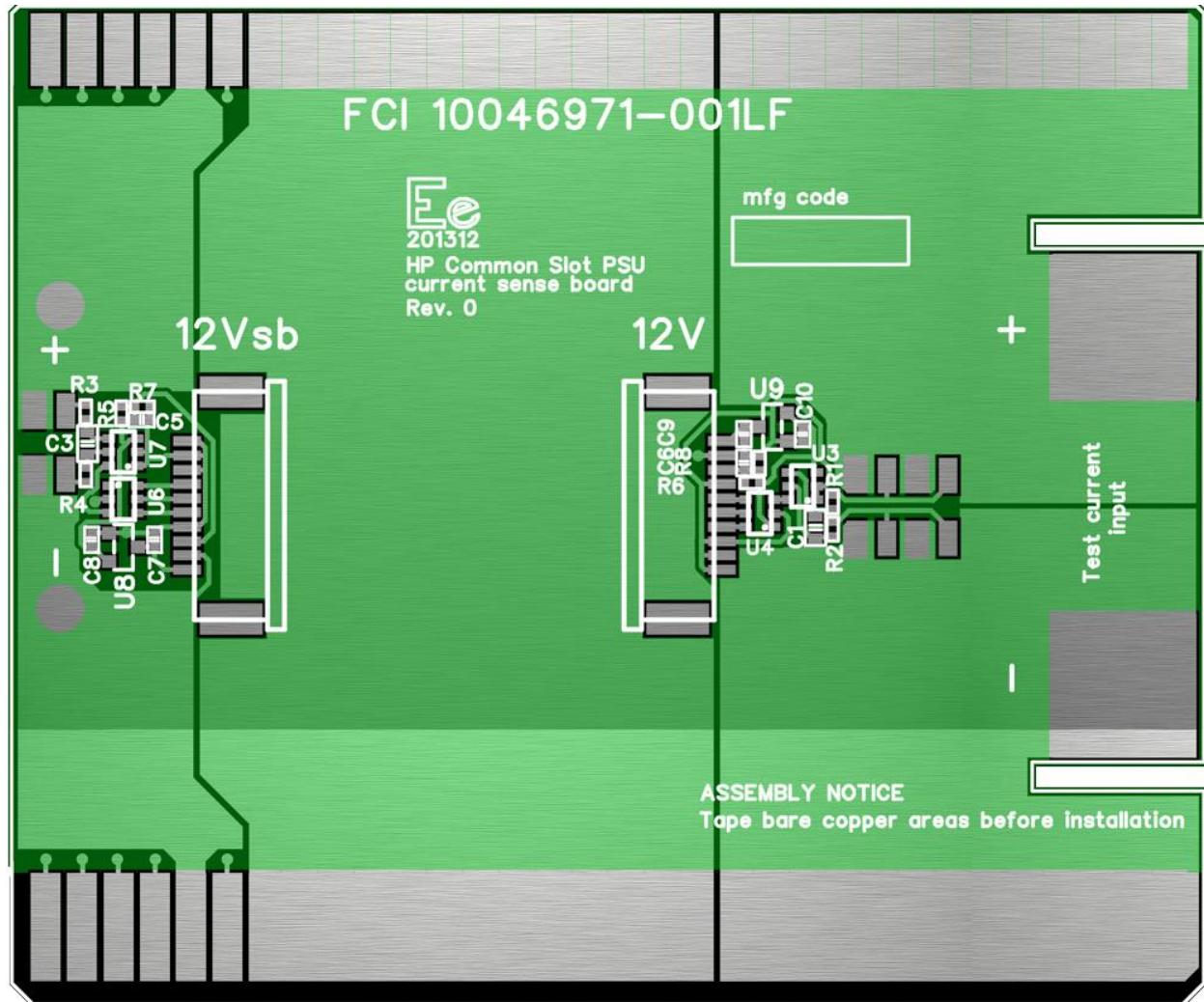
Productie	2x
Serienummers	PCIEx16_0-1 tot PCIEx16_0-2



Dit bord is geheel identiek aan het PCIe x4 meetbord, maar dan geschikt voor een PCI-express x16 slot en met de juiste vorm om in de DL360 G7 te passen. In verband met de uitzonderlijke afmetingen van dit bord (264x29mm) kan het enkel besteld worden bij Eurocircuits, geen van de Chinese printplaatfabrikanten levert deze afmetingen 4-laags borden.

PSUHP_0 Common Slot PSU meetbord – revisie 0

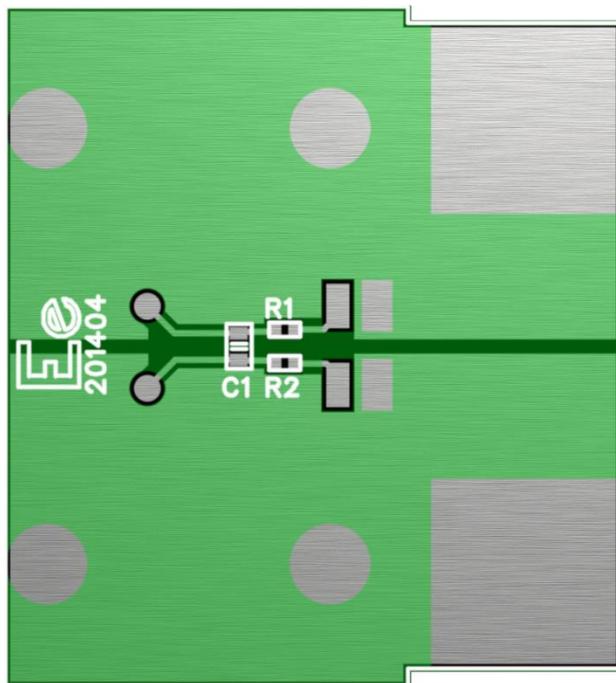
Productie	2x
Serienummers	PSUHP_0-1 tot PSUHP_0-2



Deze printplaat wordt tussen de PSU in de DL360 G7 (of elke andere HP Common Slot voeding) en het moederbord geplaatst, en meet de totaalstroom die hier doorheen vloeit (=het totale DC verbruik van de server). Er zitten hier twee digitale meetbordjes op geïntegreerd: de linkse meet 12V standby (12Vsb), de rechtse de 12V-lijn als de computer aanstaat. Zeker de zwaardere typen HP Common Slot-voedingen zijn in staat om meer dan 60A te leveren, dus de rechtse stroommeter is voorzien van twee footprints voor een meetweerstand in parallel. Ook kan het 12V-meetcircuit gekalibreerd worden met de standaard Molex Poweredge-kalibratieopstelling. Het linkse, minder kritieke meetcircuit kan worden gekalibreerd met draden die aan de '+' en '-' kopervlakken kunnen worden gesoldeerd.

TD_0 – Dummy-bord met meetaansluitingen – revisie 0

Productie	5x
Serienummers	TD0-0 t/m TD0-5



Dit is een bordje dat gewoonlijk dezelfde functie vervult als het dummy-bordje, maar daarnaast ook:

- Met banaanstekker-aansluitingen kan worden uitgerust om zo met een multimeter de stroom te kunnen meten
- Met een weerstand en filter kan worden uitgerust om met een multimeter of oscilloscoop te kunnen worden gemeten

Bijlage II: Kalibratietabel

Kalibratietabel van de meetbordjes die gebruikt zijn tijdens dit project. De kalibratietabel is gemaakt en aangeleverd door Eric.

Printplaat	Datum Kalibratie	Board_oms	Shunt	Implicati	V_offset	R_Calibrated	R_calibrated_val	Accuracy > 2Amp	Accuracy 1Amp	Opmekingen	Geplaatst in	Kanaal	Calibrated / OFFSET	
			milli Ohm	(x)	mV	Ohm	Amperes	max deviation in (%)	max deviation in (%)				m Ohm	
I1-1	Kalibratie 1: 16 april 2014 - 11:31:4	Board 1	1,5	100	0,04932	0,14270	1330,3	4,5	0,15	0,80	Doe 1 Ampere met afwijking van 0,8%	67	12V_HDD	1,42699
I1-9	Kalibratie 1: 16 april 2014 - 11:31:4	Board 3	1,5	100	0,05130	0,14514	1447,3	4,5	0,15	0,30	prima	67	5V_HDD	1,45137
I1-0	Kalibratie 1: 16 april 2014 - 11:31:4	Board 4	1,5	100	0,05363	0,14490	1294,3	4,5	0,15	0,30	offset 54mV ipv 50 mV			1,44898
I1-2	Kalibratie 1: 16 april 2014 - 11:31:4	Board 5	1,5	100	0,05125	0,14380	1376,6	4,5	0,15	0,30	prima	67	3V3_SYSTEM	1,43796
I1-5	Kalibratie 1: 16 april 2014 - 11:31:4	Board 6	1,5	100	0,05075	0,14109	1206,9	4,5	0,15	0,30	prima	67	CPU1	1,41090
I0-11	Kalibratie 2: 16 april 2014 - 12:13:2	Board 1	1	100	0,05144	0,09950	1414,3	4,8	0,30	xx	Geen 1 Amper meting			0,99504
I1-8	Kalibratie 2: 16 april 2014 - 12:13:2	Board 3	1,5	100	0,05193	0,15202	1570,4	4,8	0,30	xx	Geen 1 Amper meting			1,52016
I1-4	Kalibratie 2: 16 april 2014 - 12:13:2	Board 4	1	100	0,05091	0,09292	1432,1	4,8	0,30	xx	Geen 1 Amper meting	67	MEM1	0,92920
I1-7	Kalibratie 2: 16 april 2014 - 12:13:2	Board 5	1	100	0,05166	0,09162	1330,8	4,8	0,30	xx	Geen 1 Amper meting	67	dummy MEM2	0,91622
I1-6	Kalibratie 2: 16 april 2014 - 12:13:2	Board 6	1	100	0,05025	0,09143	1191,4	4,8	0,30	xx	Geen 1 Amper meting	67	dummy CPU2	0,91429
I0-1	Kalibratie 3: 23 april 2014 - 13:38:0	Board 1	1	100	0,05055	0,09587	1981,1	2,42	0,30	0,50				0,95867
I0-2	(check) Kalibratie 3: 23 april 2014 - 13:38:0	Board 2	1	100	0,05034	0,09952	2312,1	2,42	0,30	0,50				
PSU_VSB	Kalibratie 3: 23 april 2014 - 13:38:0	Board 3	2	50	0,00155	0,00000-	1512,8	xx	xx	xx	Allereen OFFSET	67	PSU12V_VSB	xx
PSU	Kalibratie 3: 23 april 2014 - 13:38:0	Board 4	0,5	50	0,00154	0,02307	1487,3	+/- 7 Amp	0,6% voor stromen > 5	xx	Allereen gekalibreerd voor stromen > 5 Ampere	67	PSU12V	0,46150

Bijlage III: MATLAB code

De MATLAB code die gebruikt is tijdens dit project. De MATLAB code is in samenwerking met het SEFlab ontwikkeld.

Main.m

```
clear; %Clear the workspace in MATLAB to start clean.  
format long; %Set resolution of variables high.  
  
measurement;  
  
init_servers; %rename naar init_servers en plaats in main.  
  
for n = 1: number_of_files%number of files to read in.  
    a = char(prompt(n,:)); %the path to the datafile is selected by the n.  
    Every time n adds the next path is selected.  
    [data,number_of_rows] = read(a); %the function read reads in the  
    file and returns the data and the number of rows of the file.  
    convert%call convert.  
  
    for j = 1: number_of_slots%number_of_slots are the numer of slots given  
    in the measurement file.  
        begin = slots_begin(j,:,n); %begin defines the begin of each slot.  
        eind = slots_eind(j,:,n); %eind defines the end of each slot  
        [row_begin,colomn_begin] = min(abs(date_time-begin)); %this  
        returns the row and colomn that compairs with the given slot range.  
        [row_eind,colomn_eind] = min(abs(date_time-eind));  
        total_power =  
        sum(channel_power(colomn_begin(1):colomn_eind(1),1:size_p_voltage_channels(2))); %the total power of each channel of each slot is calculated.  
        slot_data_joules(j,:,n) = total_power/sample_frequency; %the  
        total power must be devided by the samplefrequentie toe get the total  
        energy in joules.  
    end  
    total_channel_power(number_of_rows(1),size_p_voltage_channels(2),n) =  
    zeros;  
    total_channel_power(:,:,:n) = channel_power;  
    slot_data_joules(j,:,:n) %the content of slot_data_joules is printed to  
    the screen, it is to get some feedback from MATLAB to see it is doing  
    something.  
end  
  
plotten%in this file graphs are created from de calculated data.
```

Measurement.m

```
server_name = 'G7_1_cal';  
  
number_of_files = 1;  
number_of_slots = 8;  
sample_frequency = 1;  
  
channel_to_get = 5;  
  
formatIn = 'YYYY-dd-mm HH:MM:SS';  
DateString = '2014-03-13 13:49:06';  
datenum(DateString,formatIn);
```

```

prompt = ['D:\Remy\Documents\MATLAB\Metingen\4. DNSSEC\Windows + Unbound\G7
Date 2014-05-08 Time 13;50;09,53-1.txt'];

%Define in this array the start and end time of each slot of the first
%file.
%Define in this array the start and end time of each slot of the first
%file.
slots_remy_begin = ['2014-08-05 13:50:51'; '2014-08-05 14:05:51'; '2014-08-
05 14:20:51'; '2014-08-05 14:36:12'; '2014-08-05 14:51:33'; '2014-08-05
15:06:53'; '2014-08-05 15:22:13'; '2014-08-05 15:37:33'];;
slots_remy_eind = ['2014-08-05 14:01:11'; '2014-08-05 14:16:11'; '2014-08-
05 14:31:12'; '2014-08-05 14:46:33'; '2014-08-05 15:01:53'; '2014-08-05
15:17:13'; '2014-08-05 15:32:33'; '2014-08-05 15:47:54'];;

for i = 1: number_of_slots
    slots_begin(i,:,:,1) = datenum(slots_remy_begin(i,:)),formatIn);
    slots_eind(i,:,:,1) = datenum(slots_remy_eind(i,:)),formatIn);
end

%Define in this array the start and end time of each slot of the second
%file.
%slots_remy_begin = ['2014-04-04 12:10:00'; '2014-04-04 12:25:00'; '2014-
04-04 12:40:00'];
%slots_remy_eind = ['2014-04-04 12:20:00'; '2014-04-04 12:35:00'; '2014-04-
04 12:50:00'];

%for i = 1: number_of_slots
%    slots_begin(i,:,:,2) = datenum(slots_remy_begin(i,:)),formatIn);
%    slots_eind(i,:,:,2) = datenum(slots_remy_eind(i,:)),formatIn);
%end

```

Plotten.m

```

figure; %create an empty figure.
variable1 = zeros(1,number_of_files)';
variable2 = zeros(1,number_of_files)';

if(number_of_files == 1)
    variable1(i) = slot_data_joules(1,1);
%slot_data_joules(slot(row),channel(colomn))
    variable2(i) = slot_data_joules(1,3);
%slot_data_joules(slot(row),channel(colomn))
else
    %this for loop gets in this case de CPU power of slot 1 from
    %slot_data_joules. This conversion is nessesery to plot the data. When
more
    %channels must be plot this code must be copied and add in the bar
    %function.
    %variable1 = zeros(1,number_of_files)';
    for i = 1:number_of_files
        variable1(i) = slot_data_joules(1,1,i);
%slot_data_joules(slot(row),channel(colomn),file(depth) (z))
    end

    %this for loop gets in this case de MEM1 power of slot 1 from
    %slot_data_joules.
    %variable2 = zeros(1,number_of_files)';
    for i = 1:number_of_files
        variable2(i) = slot_data_joules(1,3,i);
%slot_data_joules(slot(row),channel(colomn),file(depth) (z))

```

```

    end
end

bar(1:number_of_files, [variable1 variable2], 0.5, 'stack'); %this fill in
the empty figure with the given variables. The number of variables can be
added to plot more channels.

% Adjust the axis limits
%axis([0 13 0 100000]);

%Add title and axis labels
title('Energy consumption of two channels');
xlabel('Files');
ylabel('Energy (in joules)');

%Add a legend
legend('MEM1', 'CPU1');

%h=plot(total_channel_power(2:number_of_rows(1),1:12,file));%plot the
channel power, for file you must fil in a number, use 1 for file 1, 2 for
file 2 enz..
%legend(h,'MEM 1', 'CPU 2', 'CPU 1', 'MEM 2', 'HDD 5V', 'HDD 12V', 'System
3V3', 'PSU 12V', 'PSU 12V Standby', 'PCI-e x16 riser I3V3 aux', 'PCI-e x16
riser I3V3', 'PCI-e x16 riser I12V');

```

Read.m

```

%This piece of code reads data generated by labview. This data is processed
%in MATLAB for further use. The processed data is stored in the variable
%called data.
%The variables a and b are used as local variables, to convert data
%sometimes you need a buffer to store someting.
function [data,number_of_rows] = read(path)
a = importdata(path);
b = regexp替換(a, '\t', ' '); %regexp替換 replaces characters, this is
nessesery for later processing.
a = regexp替換(b, ',', '.');
importet_data = regexp替換(a, 'E', 'e'); %variables a and b are used as
draft.
number_of_rows = size(a); %Number of rows of the data file

%In this for loop the data is converted into an double array. The processed
%data is stored in the variable called data.
for i = 1 : number_of_rows(1)
    a = importet_data(i);
    b = cell2mat(a);
    a = char(b);
    b = str2num(a);
    if i==1
        data = b;
    else
        data = [data; b];
    end
end
end

```

init_servers.m

```

if strcmp(server_name, 'G7_1')

```

```

%The variables p_voltage_channels and p_temp_channels are pointers to
the coloms of data. p_voltage_channels means this
%numbers are voltages, p_temp_channels means temperature values.
p_voltage_channels = 2:13;
p_temp_channels = 14:22;

%The variables names_v and names_t give the names of the values of
data.
%This compairs with de pointers v and t.
names_v = {'MEM_1_Cur', 'CPU_2_Cur', 'CPU_1_Cur', 'MEM_2_Cur',
'HDD_5V_Cur', 'HDD_12V_Cur', 'System_3V3_Cur', 'PSU_12V_Cur',
'PSU_12V_Standby_Cur', 'PCI-e_x16_riser_I3V3aux_Cur', 'PCI-
e_x16_riser_I3V3_Cur', 'PCI-e_x16_riser_I12V_Cur'};
names_t = {'MEM_1_Temp', 'CPU_2_Temp', 'CPU_1_Temp', 'MEM_2_Temp',
'HDD_5V_Temp', 'HDD_12V_Temp', 'System_3V3_Temp', 'PSU_12V_Temp',
'PSU_12V_Standby_Temp'};

offset = [0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0, 0, 0, 0, 0];
%This is the offset of each channel.
multiplication_factor = [100, 100, 100, 100, 100, 100, 100, 50, 50,
100, 100, 100]; %This is the multiplication factor of the shunt amplifier.
resistor_value = [0.0015, 0.0015, 0.0015, 0.0015, 0.0015, 0.0015,
0.0015, 0.0005, 0.002, 0.001, 0.001, 0.001]; %This is the value of the
shunt resistor.
channel_voltage = [12, 12, 12, 12, 5, 12, 3.3, 12, 12, 3.3, 3.3, 12];
%The voltage of each channel.
elseif strcmp(server_name,'G7_1_cal')
    %The variables p_voltage_channels and p_temp_channels are pointers to
the coloms of data. p_voltage_channels means this
%numbers are voltages, p_temp_channels means temperature values.
p_voltage_channels = 2:13;
p_temp_channels = 14:22;

%The variables names_v and names_t give the names of the values of
data.
%This compairs with de pointers v and t.
names_v = {'MEM_1_Cur', 'CPU_2_Cur', 'CPU_1_Cur', 'MEM_2_Cur',
'HDD_5V_Cur', 'HDD_12V_Cur', 'System_3V3_Cur', 'PSU_12V_Cur',
'PSU_12V_Standby_Cur', 'PCI-e_x16_riser_I3V3aux_Cur', 'PCI-
e_x16_riser_I3V3_Cur', 'PCI-e_x16_riser_I12V_Cur'};
names_t = {'MEM_1_Temp', 'CPU_2_Temp', 'CPU_1_Temp', 'MEM_2_Temp',
'HDD_5V_Temp', 'HDD_12V_Temp', 'System_3V3_Temp', 'PSU_12V_Temp',
'PSU_12V_Standby_Temp'};

offset = [0.05091, 0.05025, 0.05075, 0.05166, 0.05130, 0.04932,
0.05125, 0.00155, 0.00154, 0, 0, 0]; %This is the offset of each channel.
multiplication_factor = [100, 100, 100, 100, 100, 100, 100, 50, 50,
100, 100, 100]; %This is the multiplication factor of the shunt amplifier.
resistor_value = [0.00092920, 0.00091429, 0.00141090, 0.00091622,
0.00145137, 0.00142699, 0.00143796, 0.00046150, 0.002, 0.001, 0.001,
0.001]; %This is the value of the shunt resistor.
channel_voltage = [12, 12, 12, 12, 5, 12, 3.3, 12, 12, 3.3, 3.3, 12];
%The voltage of each channel.
temp_r_offset = [1432.1, 1191.4, 1206.9, 1330.8, 1447.3, 1330.3,
1376.6, 1487.3, 1512.8]; %temp calculation
elseif strcmp(server_name,'G7_2')
    %The variables p_voltage_channels and p_temp_channels are pointers to
the coloms of data. p_voltage_channels means this
%numbers are voltages, p_temp_channels means temperature values.
p_voltage_channels = 2:13;
p_temp_channels = 14:22;

```

```

%The variables names_v and names_t give the names of the values of
data.
%This compairs with de pointers v and t.
names_v = {'MEM_1_Cur', 'CPU_2_Cur', 'CPU_1_Cur', 'MEM_2_Cur',
'HDD_5V_Cur', 'HDD_12V_Cur', 'System_3V3_Cur', 'PSU_12V_Cur',
'PSU_12V_Standby_Cur', 'PCI-e_x16_riser_I3V3aux_Cur', 'PCI-
e_x16_riser_I3V3_Cur', 'PCI-e_x16_riser_I12V_Cur'};
names_t = {'MEM_1_Temp', 'CPU_2_Temp', 'CPU_1_Temp', 'MEM_2_Temp',
'HDD_5V_Temp', 'HDD_12V_Temp', 'System_3V3_Temp', 'PSU_12V_Temp',
'PSU_12V_Standby_Temp'};

offset = [0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0, 0,
0]; %This is the offset of each channel.
multiplication_factor = [50, 50, 100, 50, 50, 50, 50, 50, 50, 50, 50,
50]; %This is the multiplication factor of the shunt amplifier.
resistor_value = [0.001, 0.001, 0.001, 0.0015, 0.001, 0.0015, 0.0015,
0.0005, 0.002, 0.001, 0.001, 0.001]; %This is the value of the shunt
resistor.
channel_voltage = [12, 12, 12, 12, 5, 12, 3.3, 12, 12, 3.3, 3.3, 12];
%The voltage of each channel.
end

```

get_slot.m

```

[row_begin,colomn_begin] = find(date_time == begin);
[row_end,colomn_end] = find(date_time == eind);
l = 1;
for k = row_begin : row_end
    slot_data(l,) = variable(l,channel_to_get);
    l = l+1;
end

```

Bijlage IV: Ongekalibreerde vs. gekalibreerde data.

In de afbeelding is het verschil tussen ongekalibreerde en gekalibreerde data te zien. De grafiek MET titel is ongekalibreerd, de grafiek zonder titel is gekalibreerd. Het grote verschil in kanaal MEM1 is te verklaren door een weerstand in het meetbordje dat verkeerd werd aangegeven in de MATLAB code.



Bijlage V: Replay script

Zoals eerder genoemd zal het afspelen van DNS data afgehandeld worden door Tcpreplay en een zelfgeschreven script. Het script zal de volgende stappen doorlopen:

1. Noteren van de starttijd in een logfile
2. Starten van het 1^e idle slot
3. Noteren van de eindtijd van het 1^e slot
4. 5 minuten wachten
 - a. De reden voor deze wachttijd is dat het even kan duren voordat alle queries beantwoord zijn. Er wordt aangenomen dat 5 minuten lang genoeg is voor de DNS software om alle queries te beantwoorden.
5. Noteren van de starttijd van het 2^e slot
6. Starten van het 2^e idle slot
7. Noteren van de eindtijd van het 2^e slot
8. 5 minuten wachten
9. Noteren van de starttijd van het 3^e slot
10. Noteren van de Tcpreplay output (looptijd, pakketstatus)
11. Noteren van de eindtijd van het 3^e slot
12. 5 minuten wachten
13. Stap 9 tot en met 12 herhalen voor slot 4 tot en met 8

Tcpreplay script in bash:

```
#!/bin/bash
echo "Warning! Only use this script when you have TCPReplay installed."
echo "Make sure your input is correct, as it will not be validated."
echo "Typo's will screw up your slot!"
echo ""
sleep 0.5
echo "What is the location to your source replay file? "
read sourcefile
ls $sourcefile
NOW=$(date +"%Y%d%m-%H%M%S")
output=slotrun$NOW
touch $output
echo "-----|tee -a $output
echo "Time at the start of the first slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "This slot is an idle slot, now sleeping for 10 minutes" |tee -a $output
sleep 10m
echo "Time at the end of the first slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----|tee -a $output
echo "Now sleeping 5 minutes to cool down"
sleep 5m
echo "Time at the start of the second slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "This slot is an idle slot, now sleeping for 10 minutes" |tee -a $output
sleep 10m
```

```

echo "Time at the end of the second slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the third slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the third slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the third slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the fourth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the fourth slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the fourth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the fifth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the fifth slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the fifth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the sixth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the sixth slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the sixth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the seventh slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the seventh slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the seventh slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output

```

```
echo "Now sleeping 5 min to cool down"
sleep 5m
echo "Time at the start of the eighth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "Tcpreplay output from the eighth slot: " |tee -a $output
tcpreplay --intf1=eth0 $sourcefile |tee -a $output
echo "Time at the end of the eighth slot: " |tee -a $output
date +"%Y-%d-%m %T" |tee -a $output
echo "-----" |tee -a $output
echo "Done!"
```

Bijlage VI: Voorbeeld output replayscript

Time at the start of the first slot:

2014-08-05 13:50:51

This slot is an idle slot, now sleeping for 10 minutes

Time at the end of the first slot:

2014-08-05 14:00:51

Time at the start of the second slot:

2014-08-05 14:05:51

This slot is an idle slot, now sleeping for 10 minutes

Time at the end of the second slot:

2014-08-05 14:15:51

Time at the start of the third slot:

2014-08-05 14:20:51

Tcpreplay output from the third slot:

Actual: 1240548 packets (106475701 bytes) sent in 620.08 seconds.

Rated: 171494.0 Bps, 1.37 Mbps, 1998.07 pps

Flows: 65275 flows, 105.13 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

 Attempted packets: 1240548

 Successful packets: 1240548

 Failed packets: 0

 Truncated packets: 0

 Retried packets (ENOBUFS): 0

 Retried packets (EAGAIN): 0

Time at the end of the third slot:

2014-08-05 14:31:12

Time at the start of the fourth slot:

2014-08-05 14:36:12

Tcpreplay output from the fourth slot:

Actual: 1240548 packets (106475701 bytes) sent in 619.08 seconds.

Rated: 171783.1 Bps, 1.37 Mbps, 2001.44 pps

Flows: 65275 flows, 105.31 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

 Attempted packets: 1240548

 Successful packets: 1240548

 Failed packets: 0

 Truncated packets: 0

 Retried packets (ENOBUFS): 0

 Retried packets (EAGAIN): 0

Time at the end of the fourth slot:

2014-08-05 14:46:33

Time at the start of the fifth slot:

2014-08-05 14:51:33

Tcpreplay output from the fifth slot:

Actual: 1240548 packets (106475701 bytes) sent in 619.06 seconds.

Rated: 171835.4 Bps, 1.37 Mbps, 2002.05 pps

Flows: 65275 flows, 105.34 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

 Attempted packets: 1240548

 Successful packets: 1240548

 Failed packets: 0

 Truncated packets: 0

 Retried packets (ENOBUFS): 0

 Retried packets (EAGAIN): 0

Time at the end of the fifth slot:

2014-08-05 15:01:53

Time at the start of the sixth slot:

2014-08-05 15:06:53

Tcpreplay output from the sixth slot:

Actual: 1240548 packets (106475701 bytes) sent in 619.09 seconds.

Rated: 171740.5 Bps, 1.37 Mbps, 2000.94 pps

Flows: 65275 flows, 105.28 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

 Attempted packets: 1240548

 Successful packets: 1240548

 Failed packets: 0

 Truncated packets: 0

 Retried packets (ENOBUFS): 0

 Retried packets (EAGAIN): 0

Time at the end of the sixth slot:

2014-08-05 15:17:13

Time at the start of the seventh slot:

2014-08-05 15:22:13

Tcpreplay output from the seventh slot:

Actual: 1240548 packets (106475701 bytes) sent in 619.07 seconds.

Rated: 171805.7 Bps, 1.37 Mbps, 2001.70 pps

Flows: 65275 flows, 105.32 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

 Attempted packets: 1240548

 Successful packets: 1240548

 Failed packets: 0

 Truncated packets: 0

 Retried packets (ENOBUFS): 0

 Retried packets (EAGAIN): 0

Time at the end of the seventh slot:

2014-08-05 15:32:33

Time at the start of the eighth slot:

2014-08-05 15:37:33

Tcpreplay output from the eighth slot:

Actual: 1240548 packets (106475701 bytes) sent in 619.07 seconds.

Rated: 171809.8 Bps, 1.37 Mbps, 2001.75 pps

Flows: 65275 flows, 105.32 fps, 1240548 flow packets, 0 non-flow

Statistics for network device: eth0

Attempted packets: 1240548

Successful packets: 1240548

Failed packets: 0

Truncated packets: 0

Retried packets (ENOBUFS): 0

Retried packets (EAGAIN): 0

Time at the end of the eighth slot:

2014-08-05 15:47:54

Bijlage VII: Woordenlijst

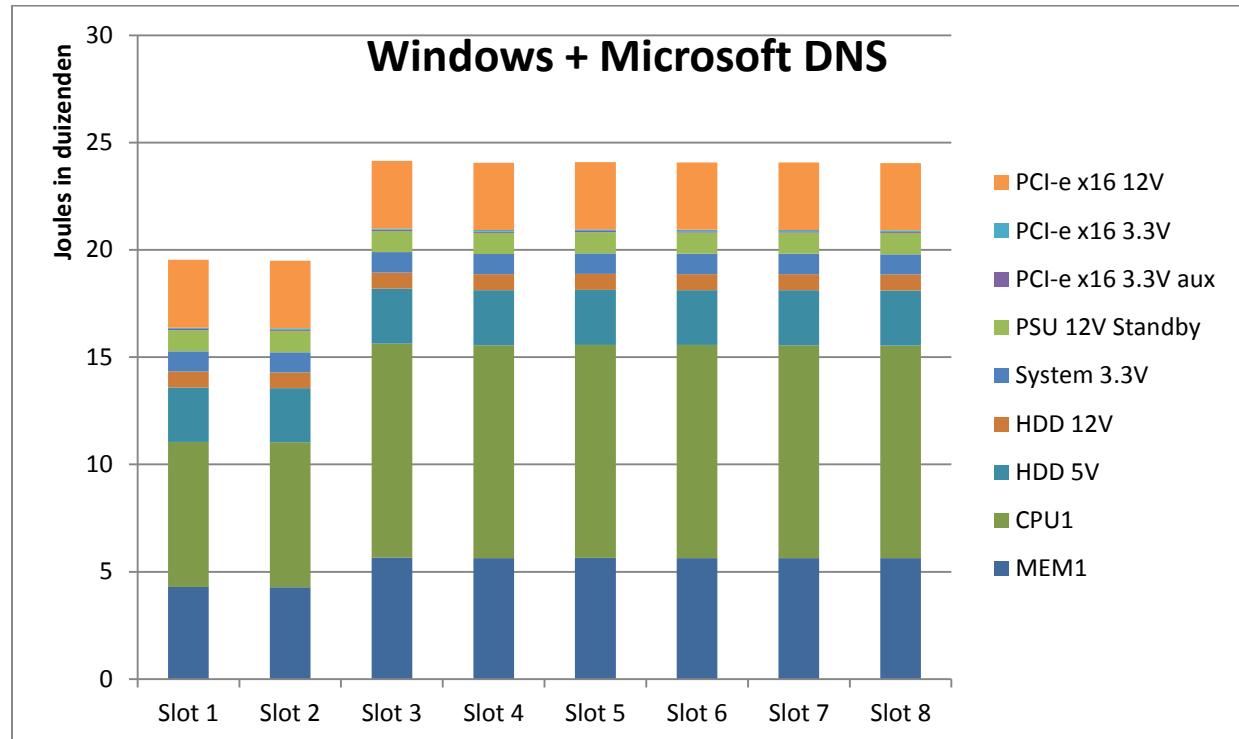
NREN	Een National Research & Education Network (NREN) is een gespecialiseerde internetprovider die onderzoek en onderwijs ondersteunt op het gebied van netwerktechnologie.
IPv4	De 4 ^e versie van het internet protocol (IP), verantwoordelijk voor het routeren van het meeste van het internetverkeer.
IPv6	De 6 ^e versie van het internet protocol. IPv6 is in het leven geroepen om het opraken van de IP adressen tegen te gaan.
DNS	Het Domain Name System is een internet service onder andere verantwoordelijk voor het vertalen van domein namen naar IP adressen.
Resolver	Een set aan software tools verantwoordelijk voor het resolven van domein namen.
DNSSEC	DNS Security, een uitbreiding op het DNS systeem. DNSSEC biedt de mogelijkheid tot authenticatie en data integriteit.
Glasvezel	Doorzichtige flexibele vezel, gebruikt voor internetverbindingen van hoge snelheid.
DNS Queries	Een verzoek van een client om een domein naam te vertalen naar een IP adres
PCAP	Packet Capture(pcap) is een API voor het opnemen van netwerkverkeer
Libpcap	De Unix implementatie van PCAP.
MAC-adres	Een Media Access Control address is een unieke aanduiding van netwerk interfaces gebruikt voor communicatie.
Unix	Een groep operating systems, CentOS is een Unix variant.
TCP/UDP	Transmission control protocol en User datagram protocol. Protocollen gebruikt om over het netwerk te communiceren.
Wireshark	Wireshark is een packet analyzer. Wireshark kan gebruikt worden om pcap files uit te lezen en te bewerken.

Bijlage VIII: Resultaten

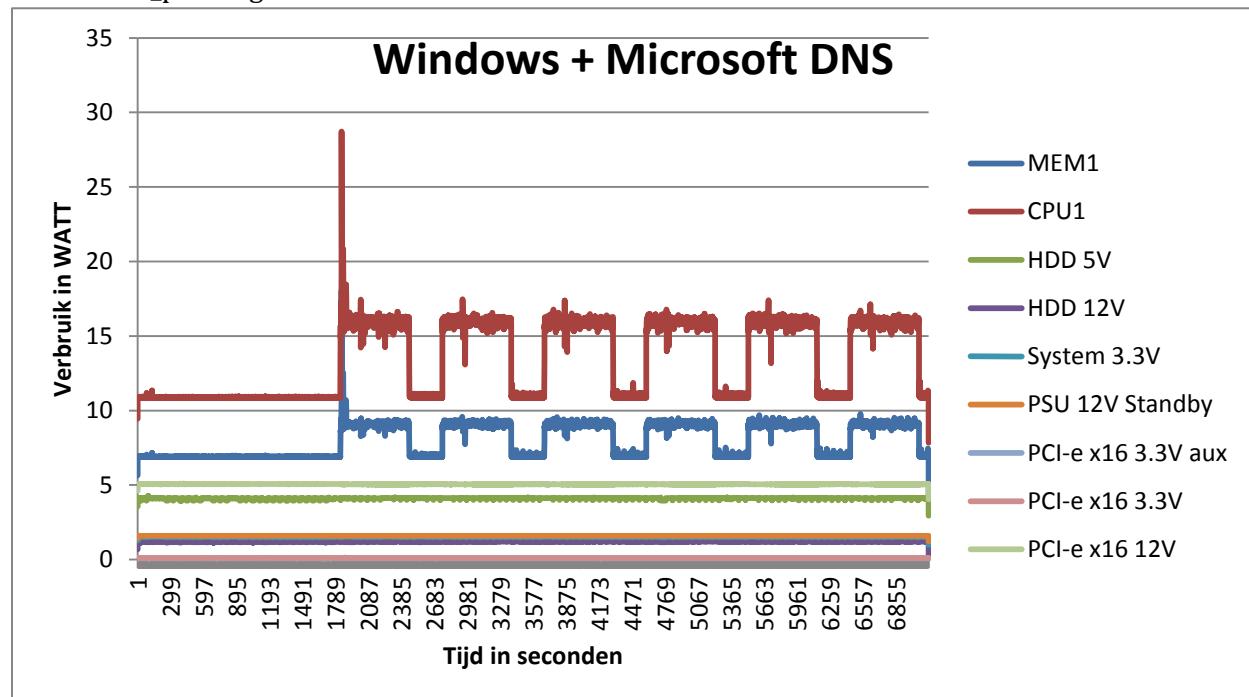
Windows

Windows + Microsoft DNS

De grafiek met de verbruiksgegevens ziet er als volgt uit:

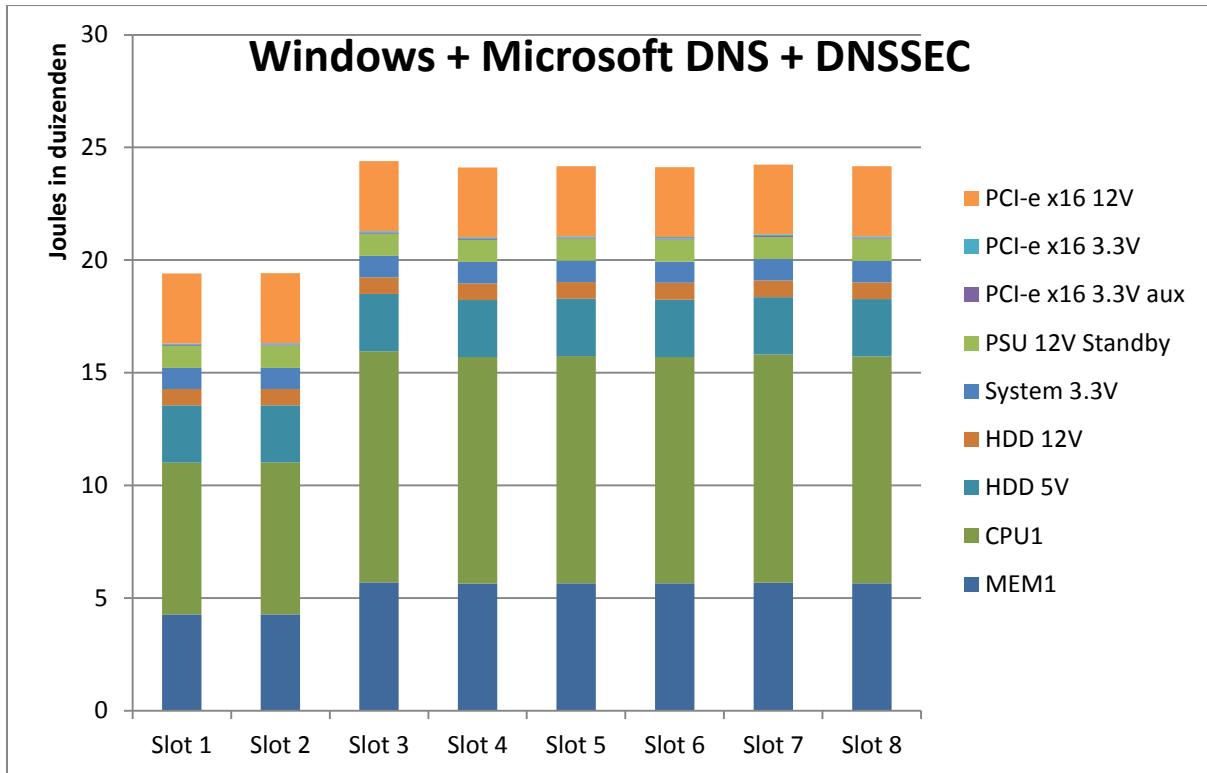


De channel_power grafiek:

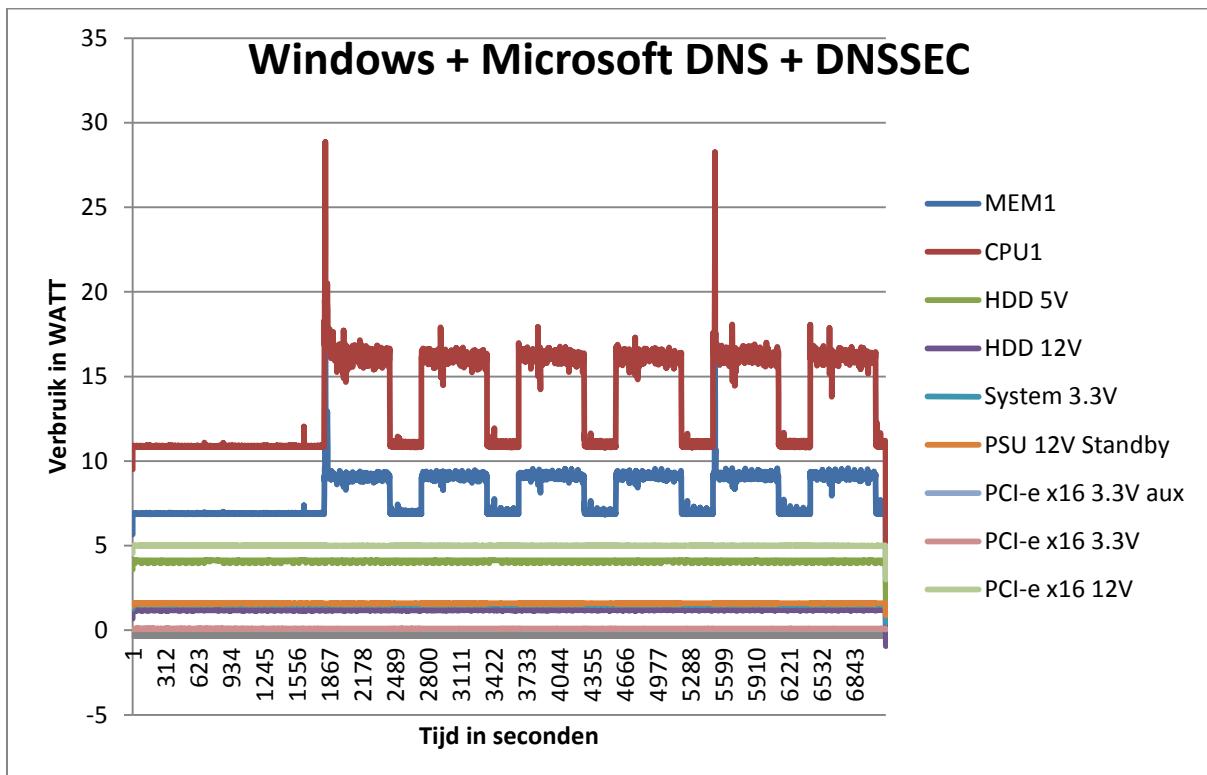


DNSSEC

De grafiek met de verbruiksgegevens ziet er als volgt uit:

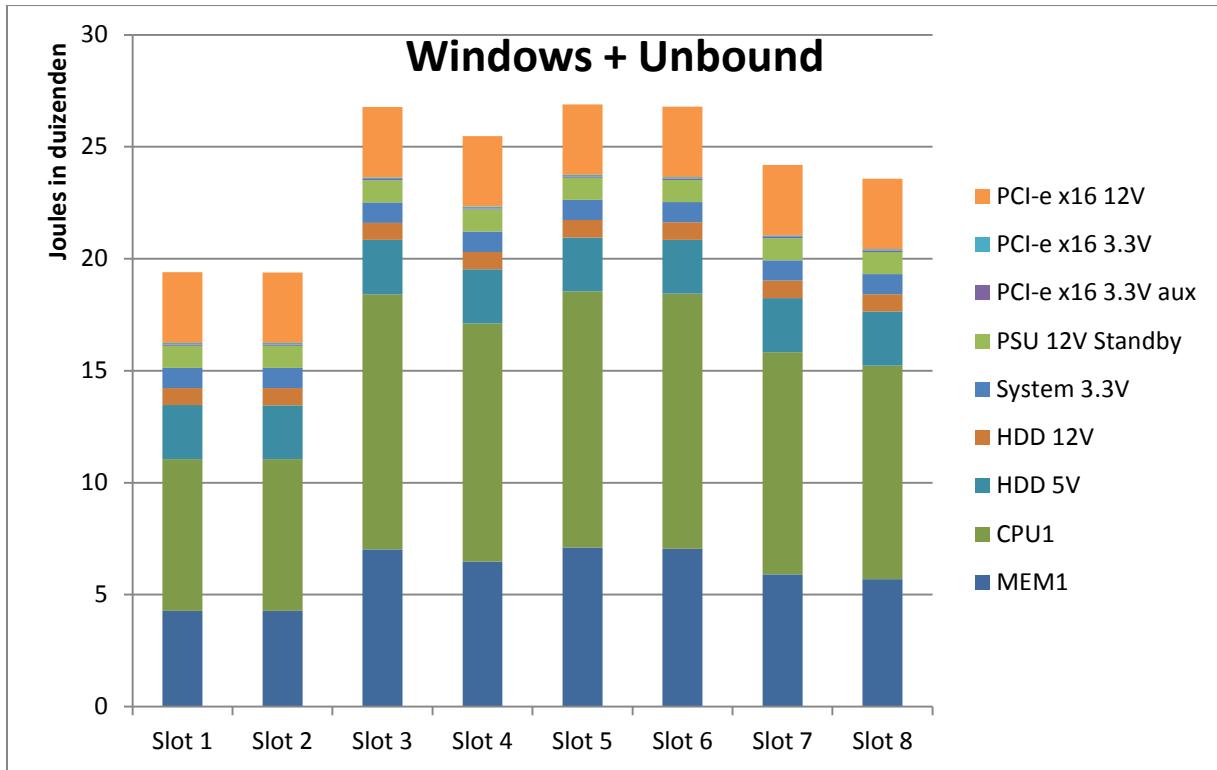


De channel_power grafiek is te zien op de volgende pagina.

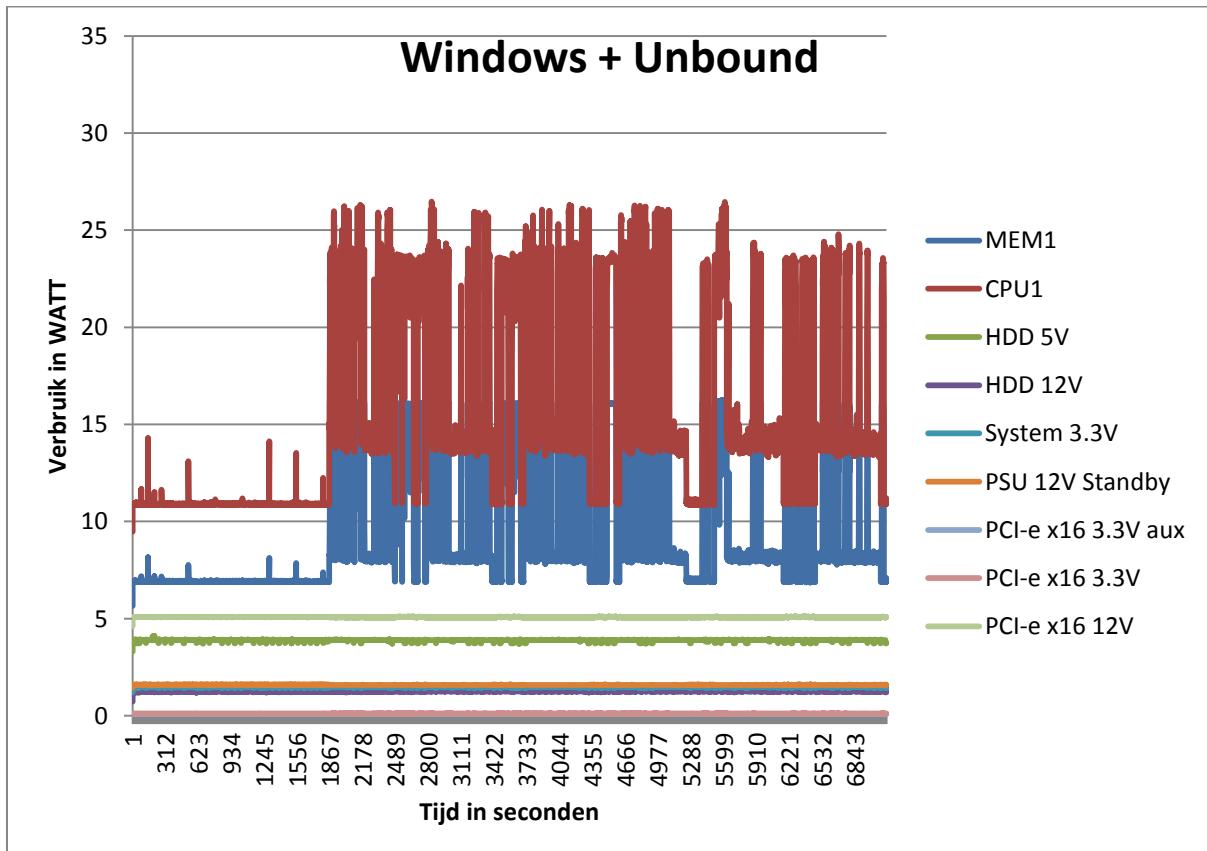


Windows + Unbound

De grafiek met verbruiksgegevens ziet er als volgt uit:

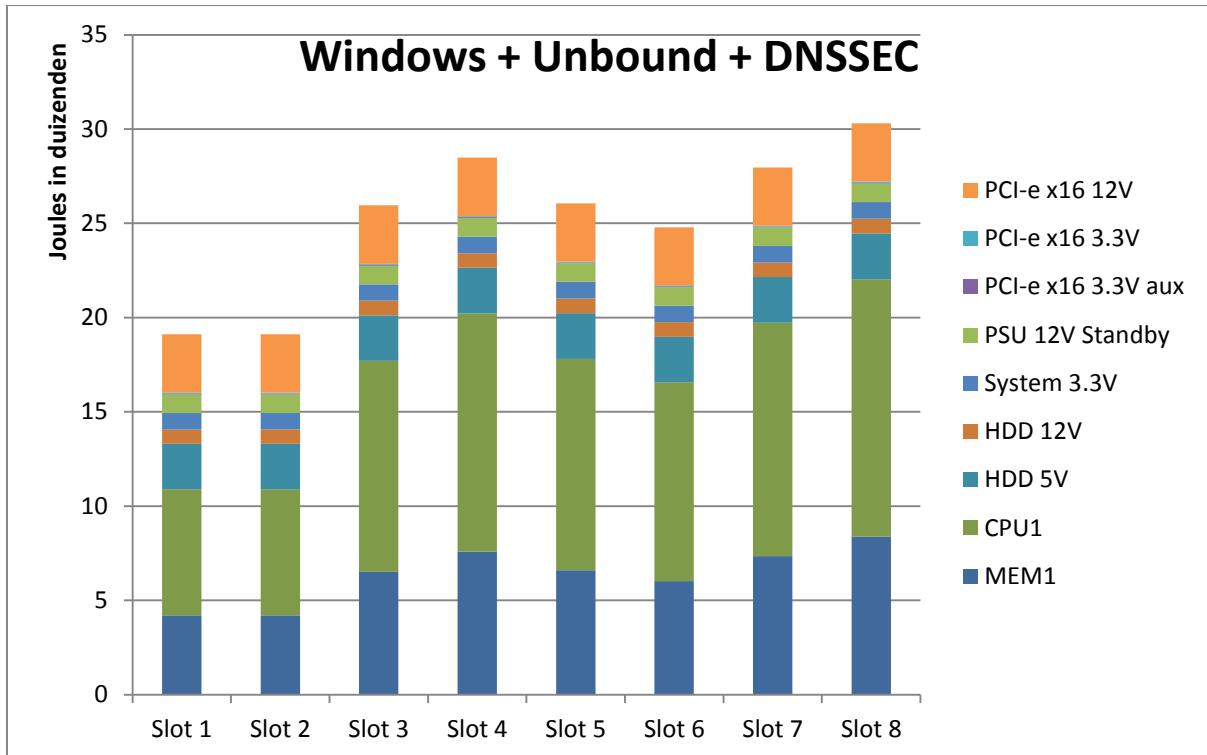


De channel_power grafiek is te zien op de volgende pagina.

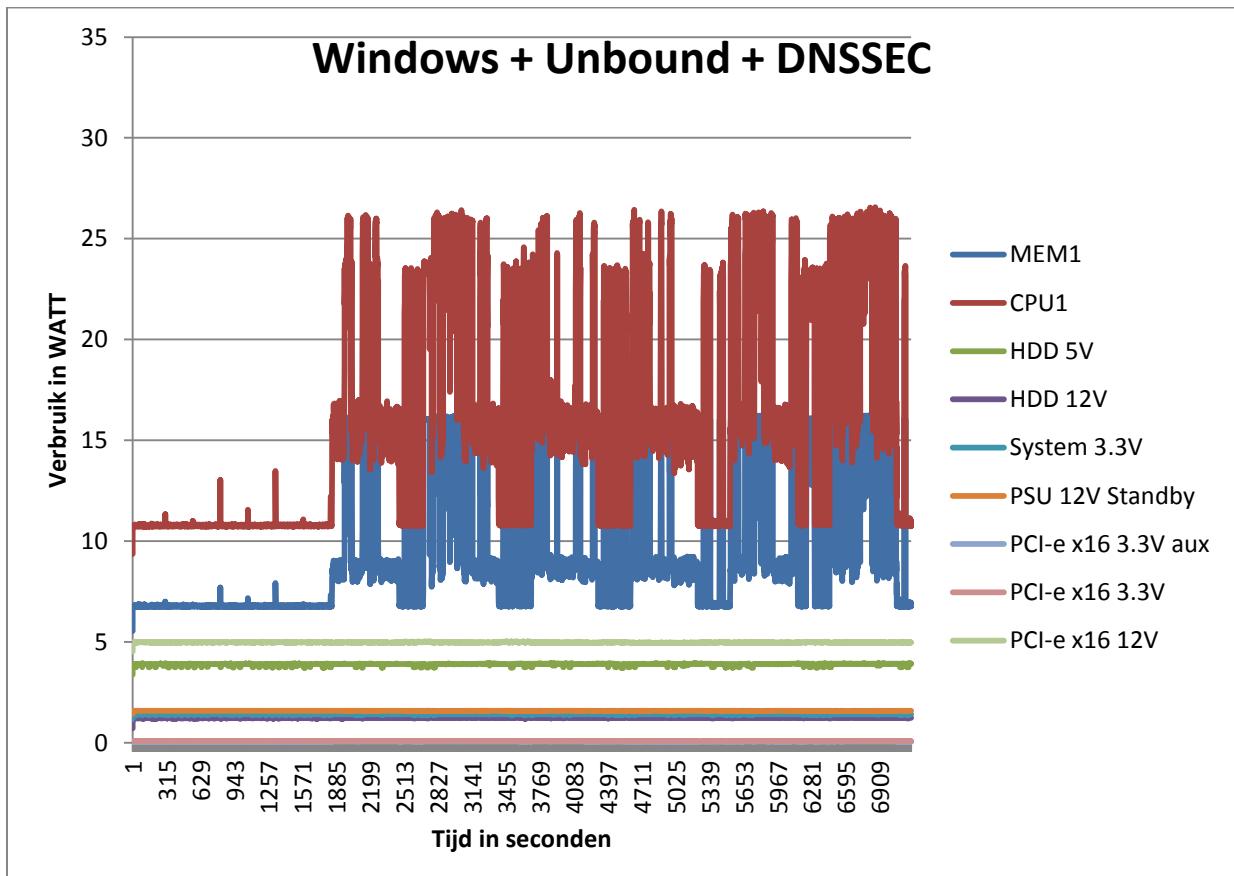


DNSSEC

De grafiek met verbruiksgegevens ziet er als volgt uit:

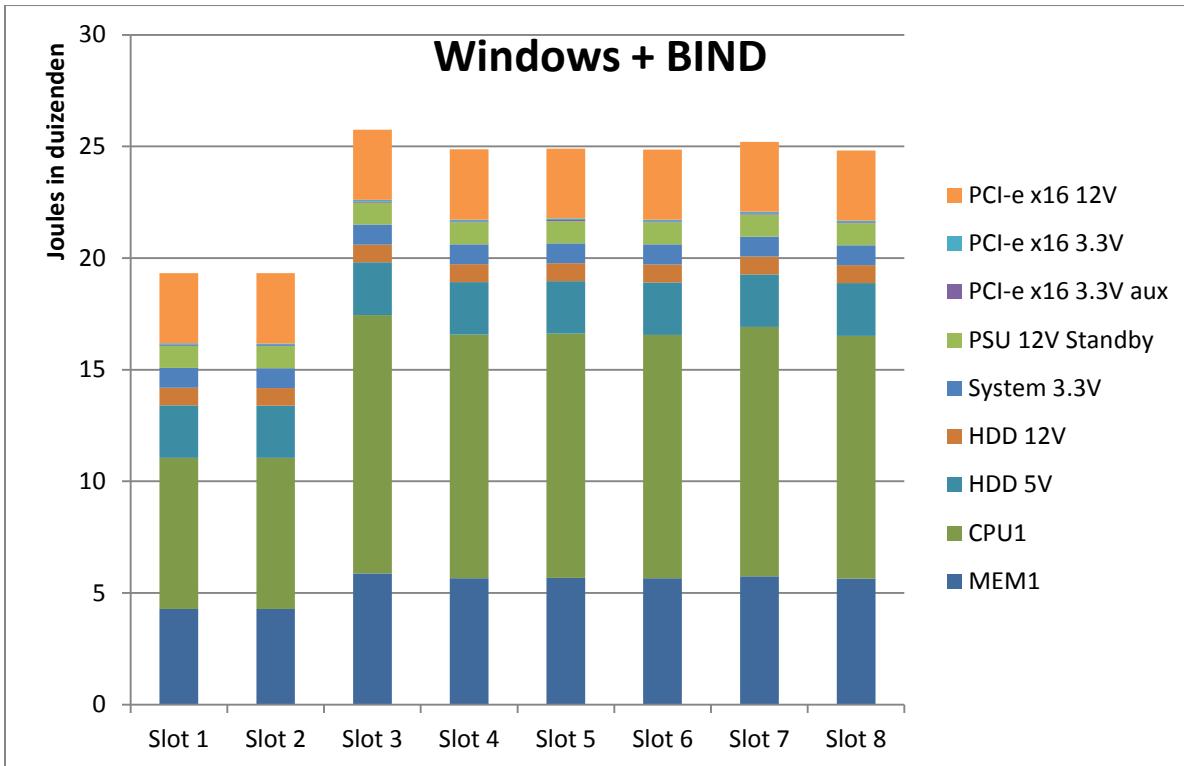


De channel_power grafiek ziet er als volgt uit:

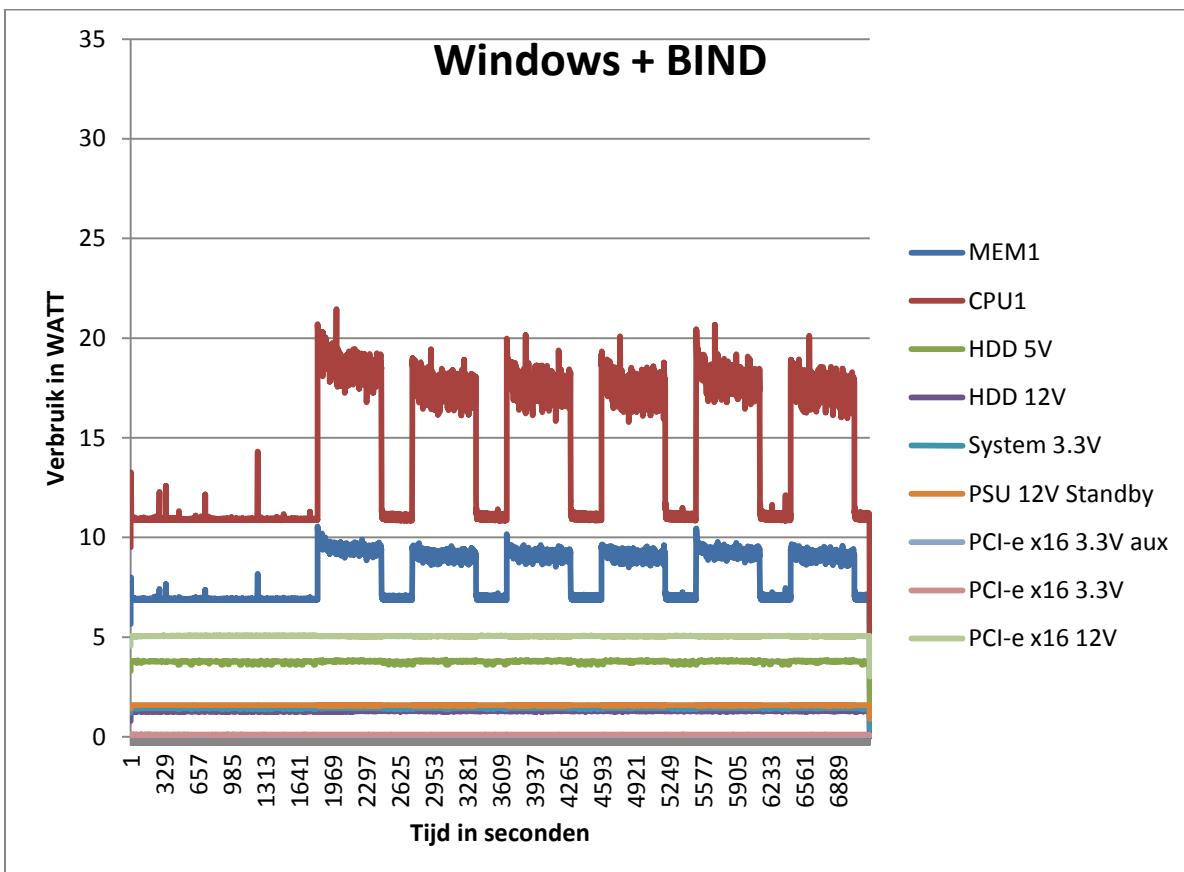


Windows + BIND

De grafiek met verbruiksgegevens ziet er als volgt uit:

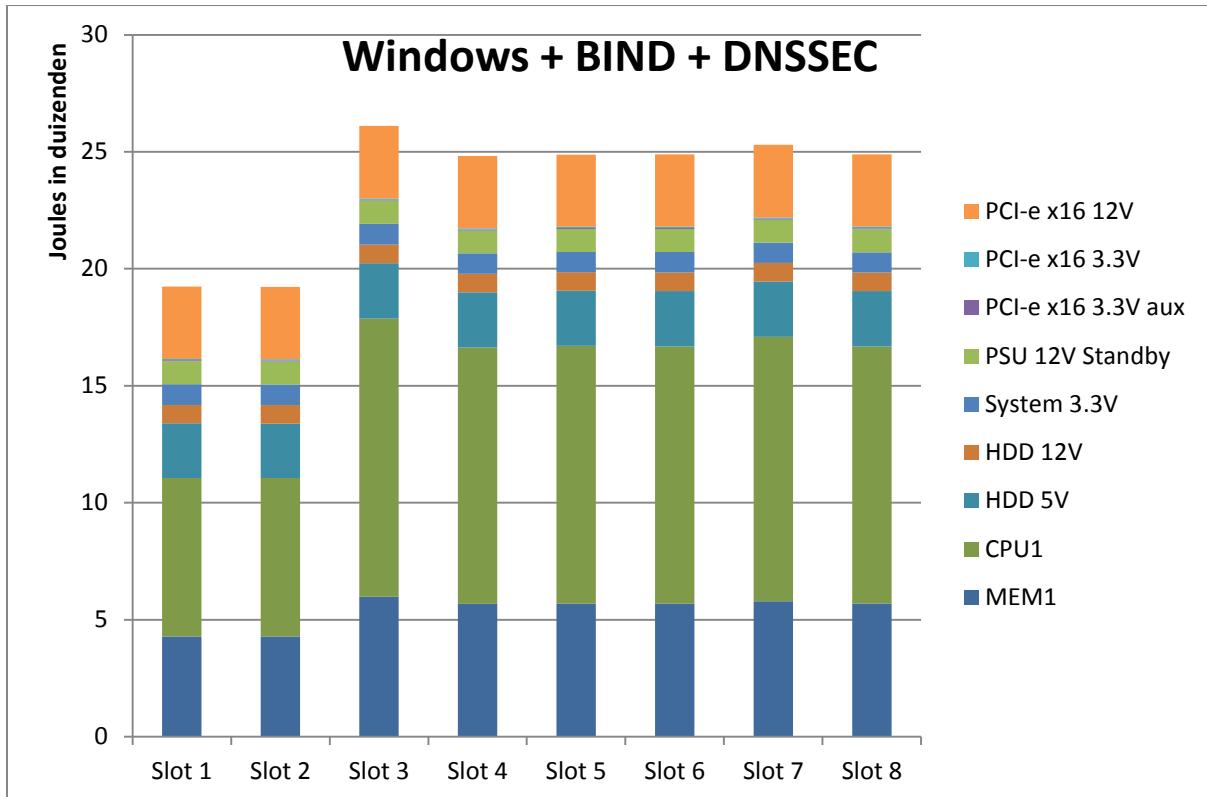


De channel_power grafiek ziet er als volgt uit:

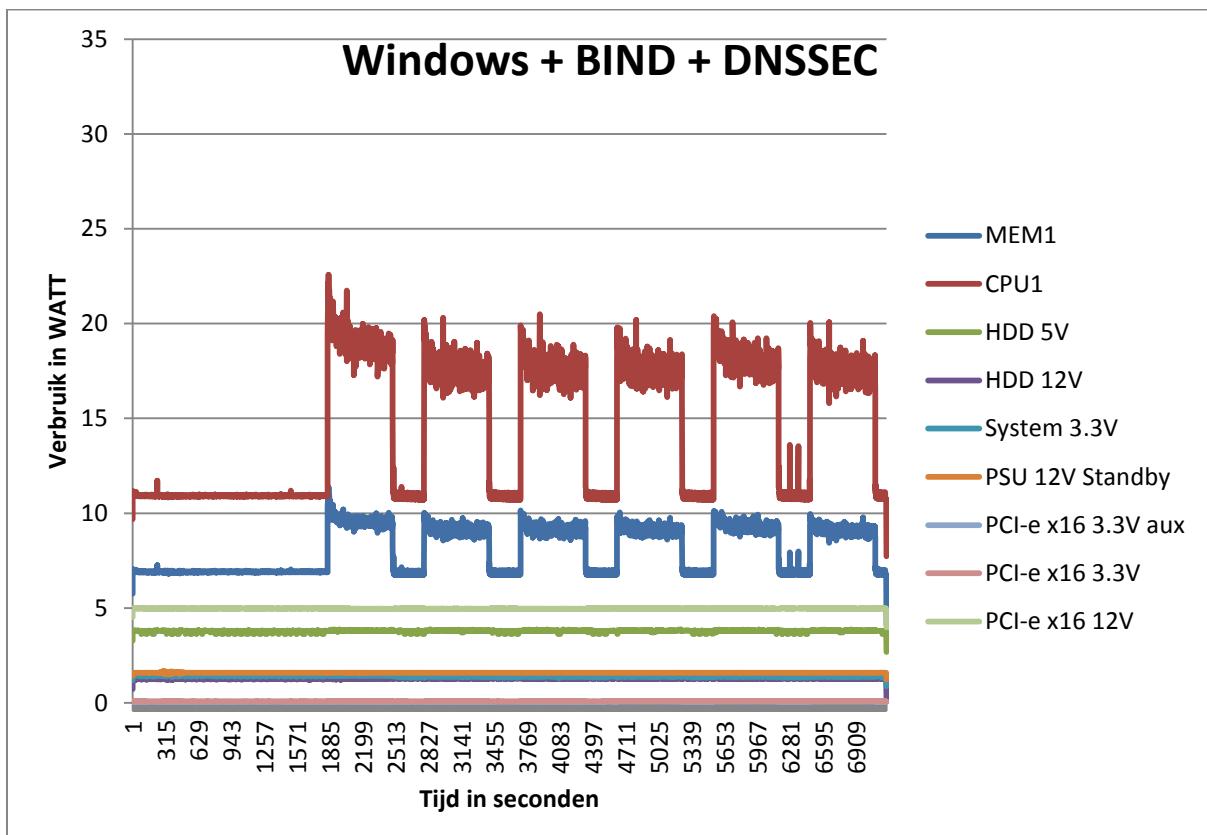


DNSSEC

De grafiek met verbruiksgegevens ziet er als volgt uit:



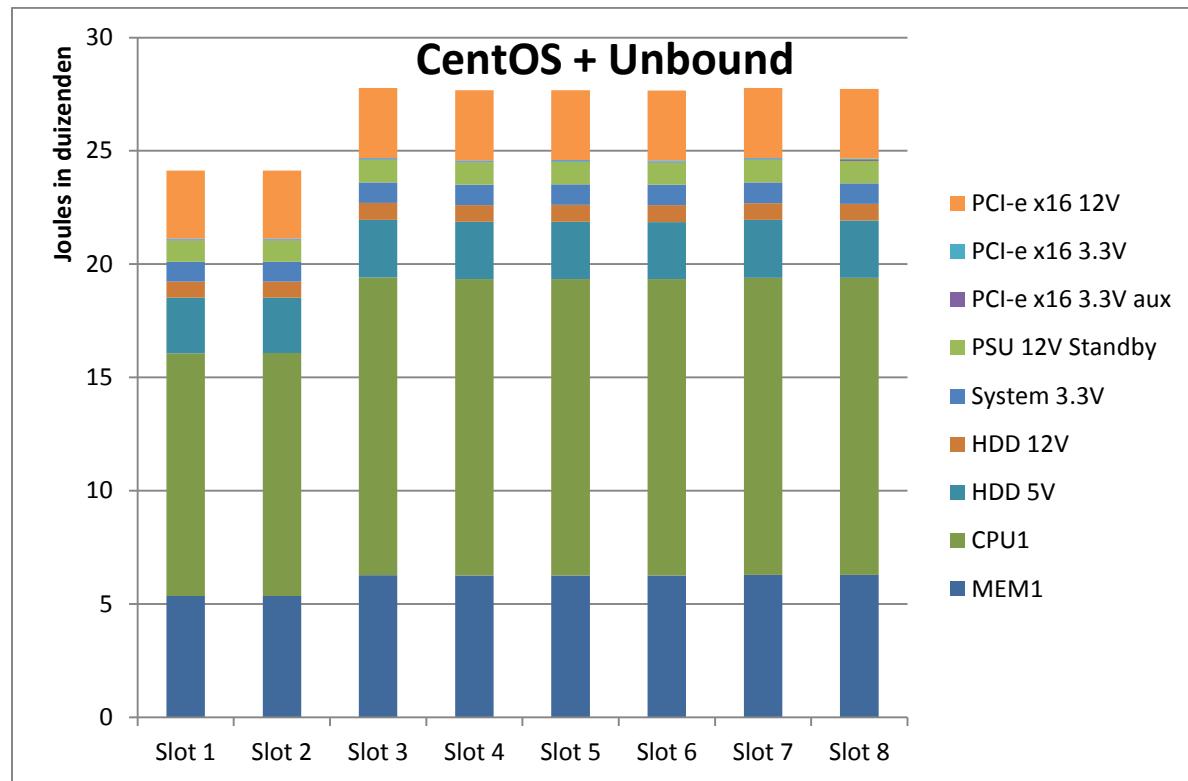
De channel_power grafiek ziet er als volgt uit:



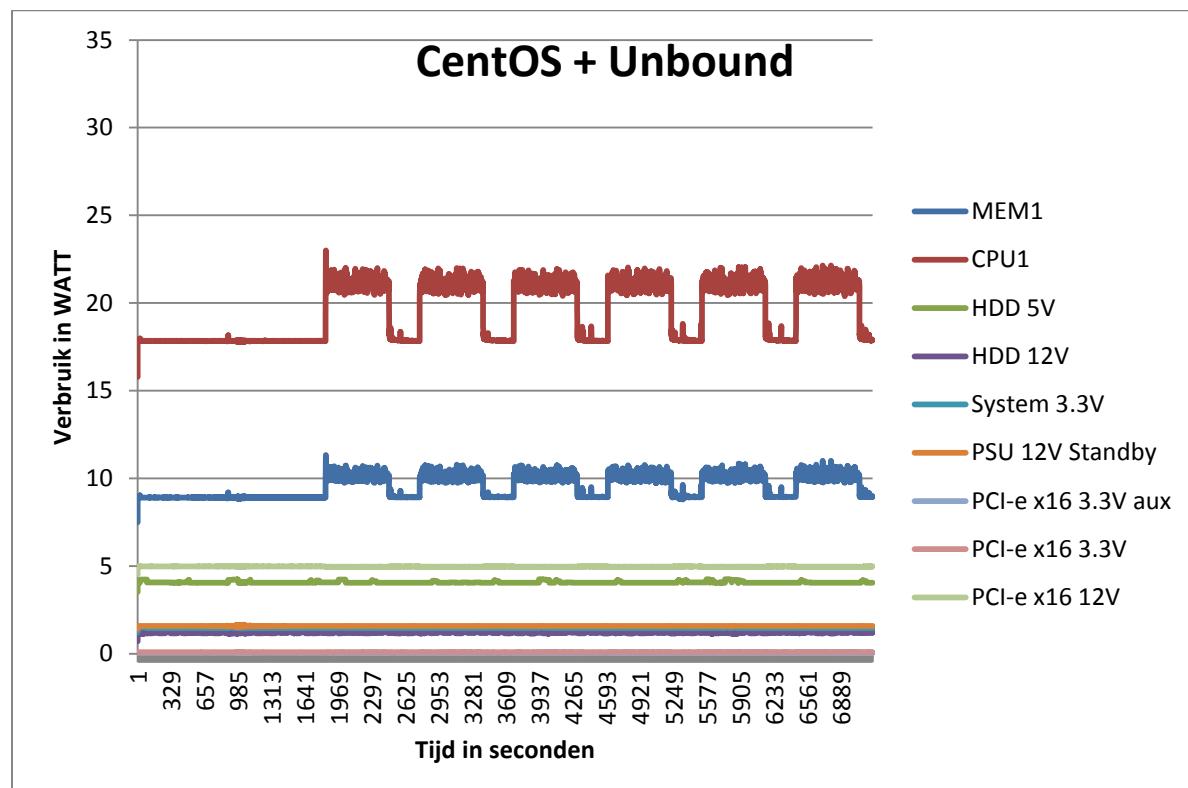
Linux

Linux + Unbound

De grafiek met verbruiksgegevens ziet er als volgt uit:

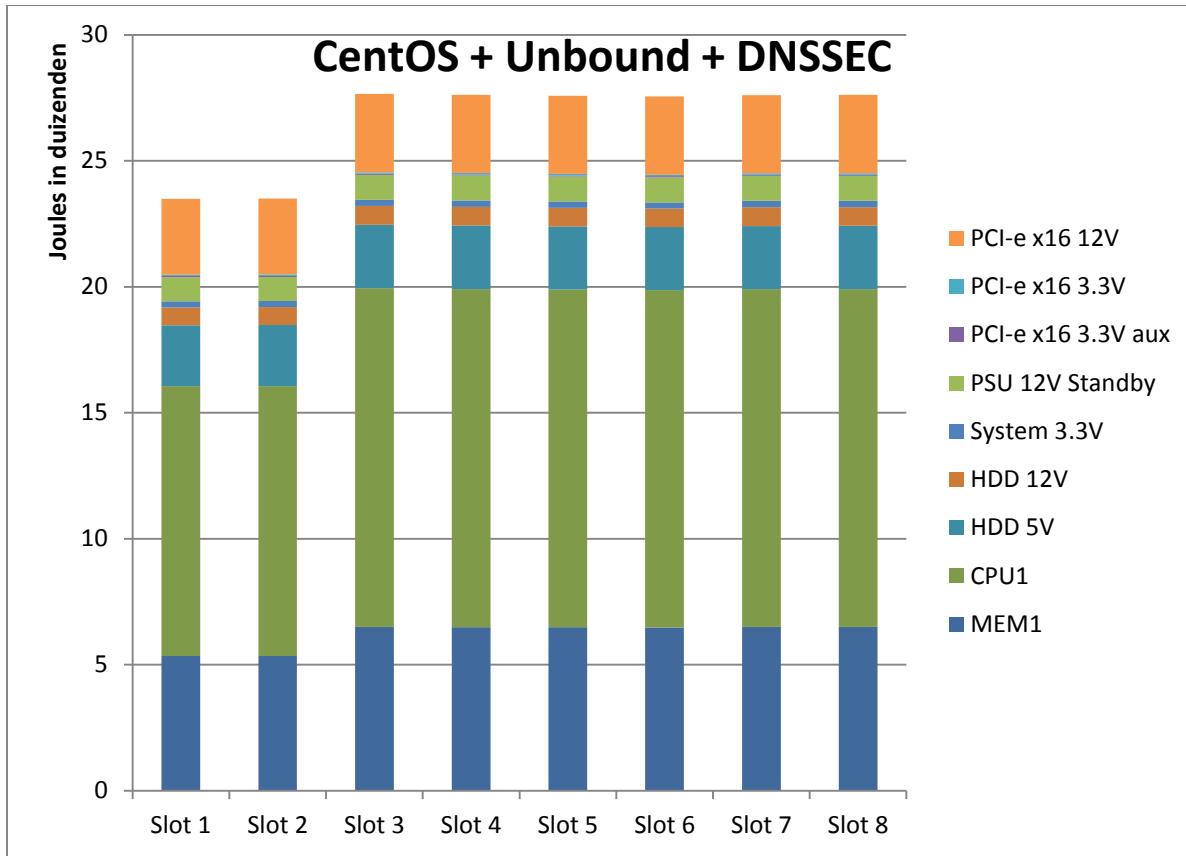


De channel_power grafiek ziet er als volgt uit:

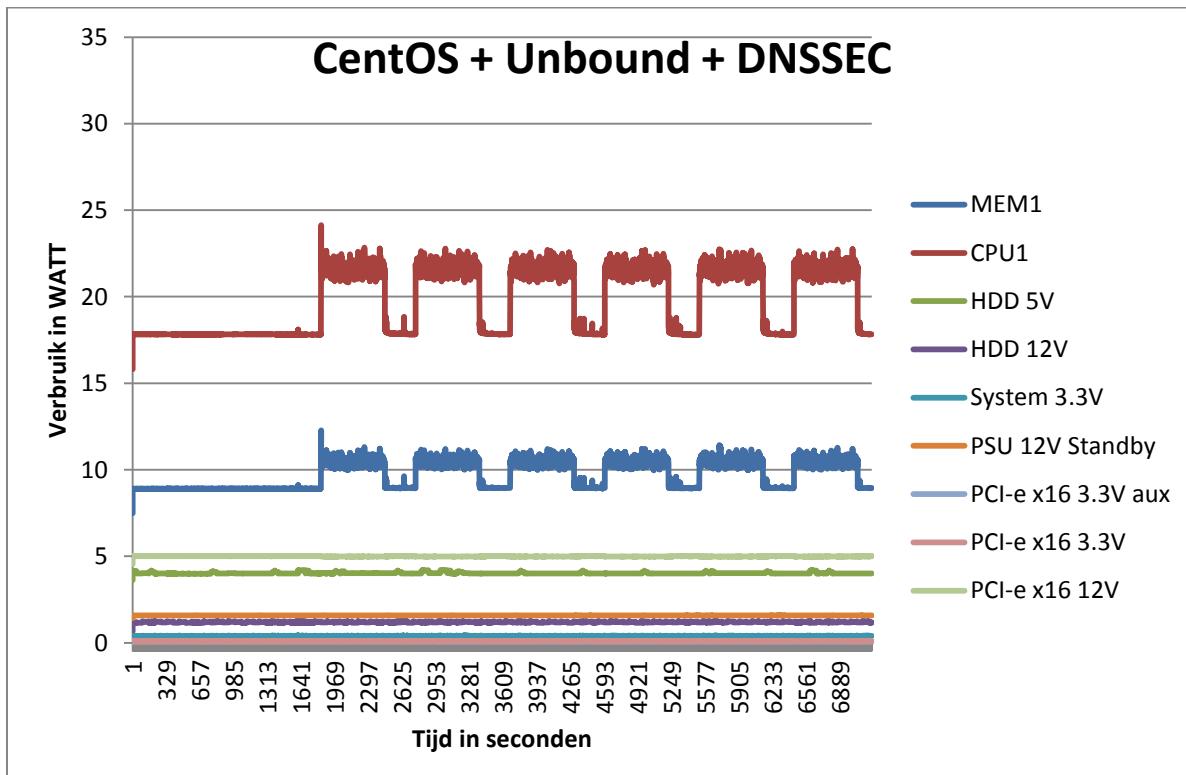


DNSSEC

De grafiek met verbruiksgegevens ziet er als volgt uit:

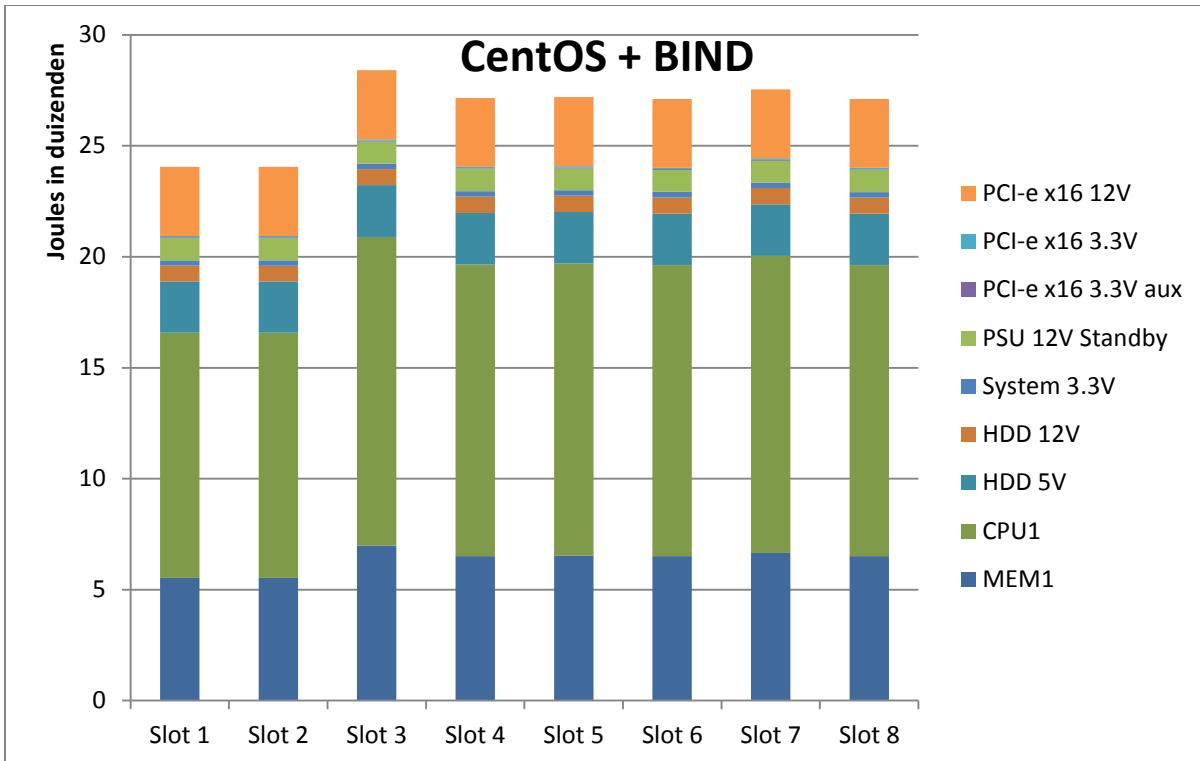


De channel_power grafiek ziet er als volgt uit:

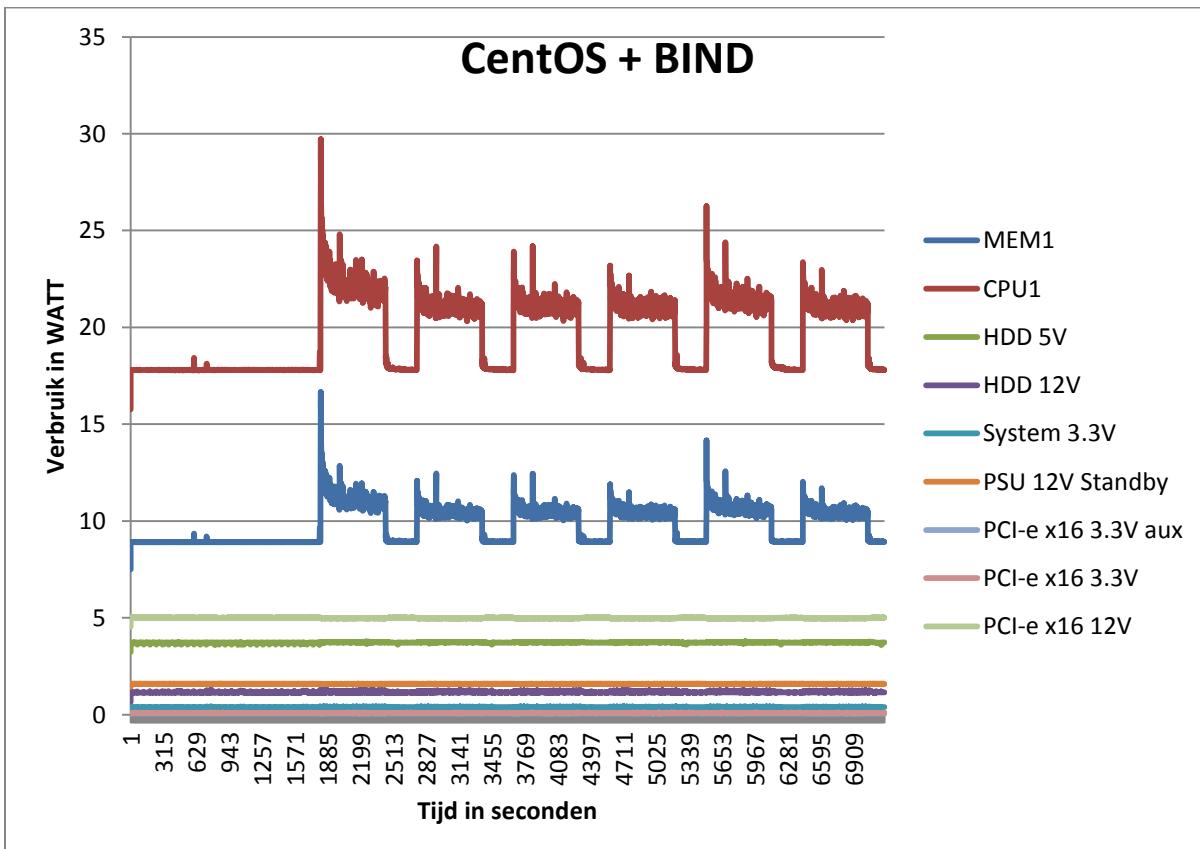


Linux + BIND

De grafiek met verbruiksgegevens ziet er als volgt uit:

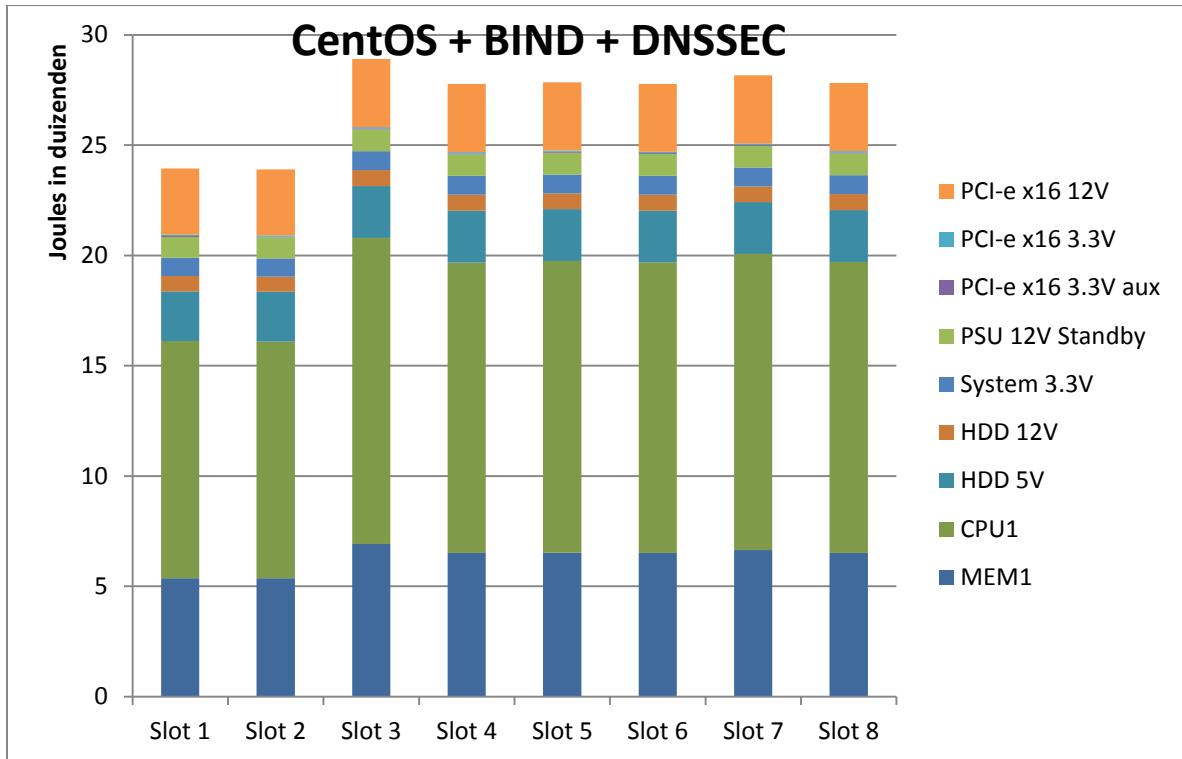


De channel_power grafiek ziet er als volgt uit:

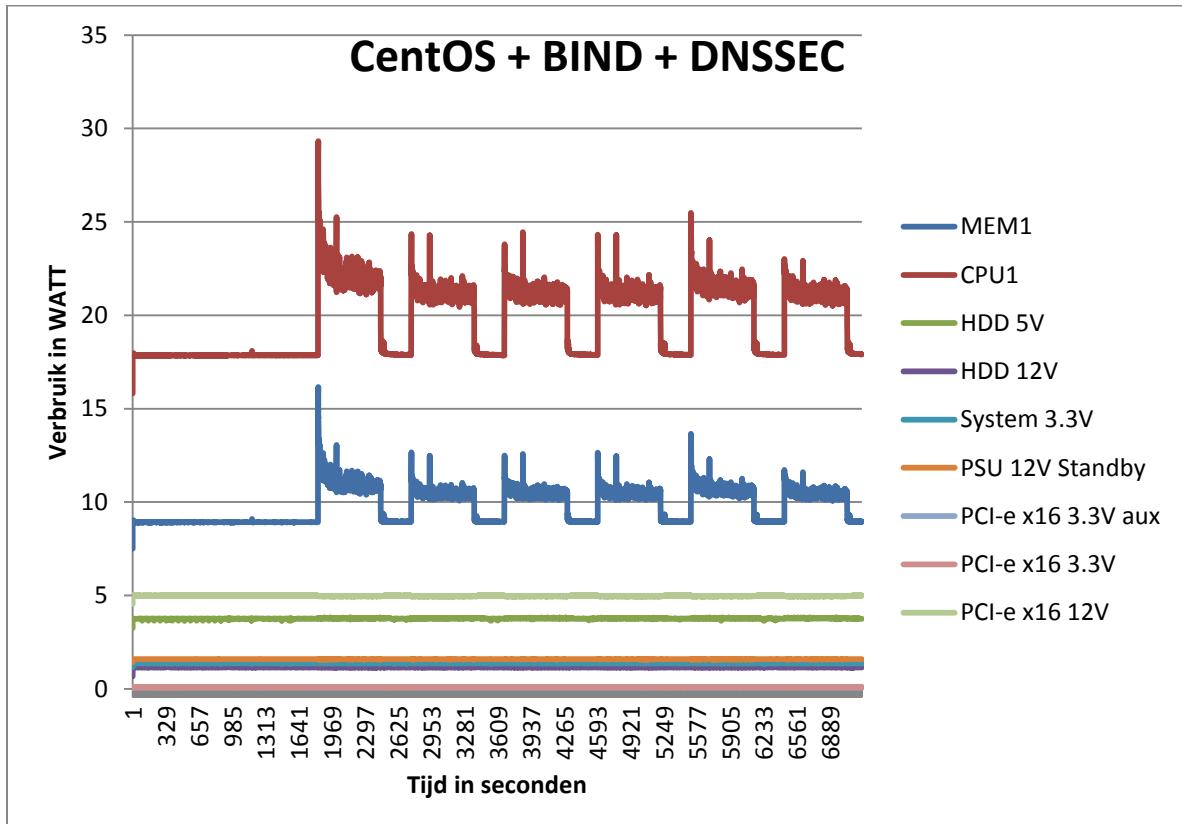


DNSSEC

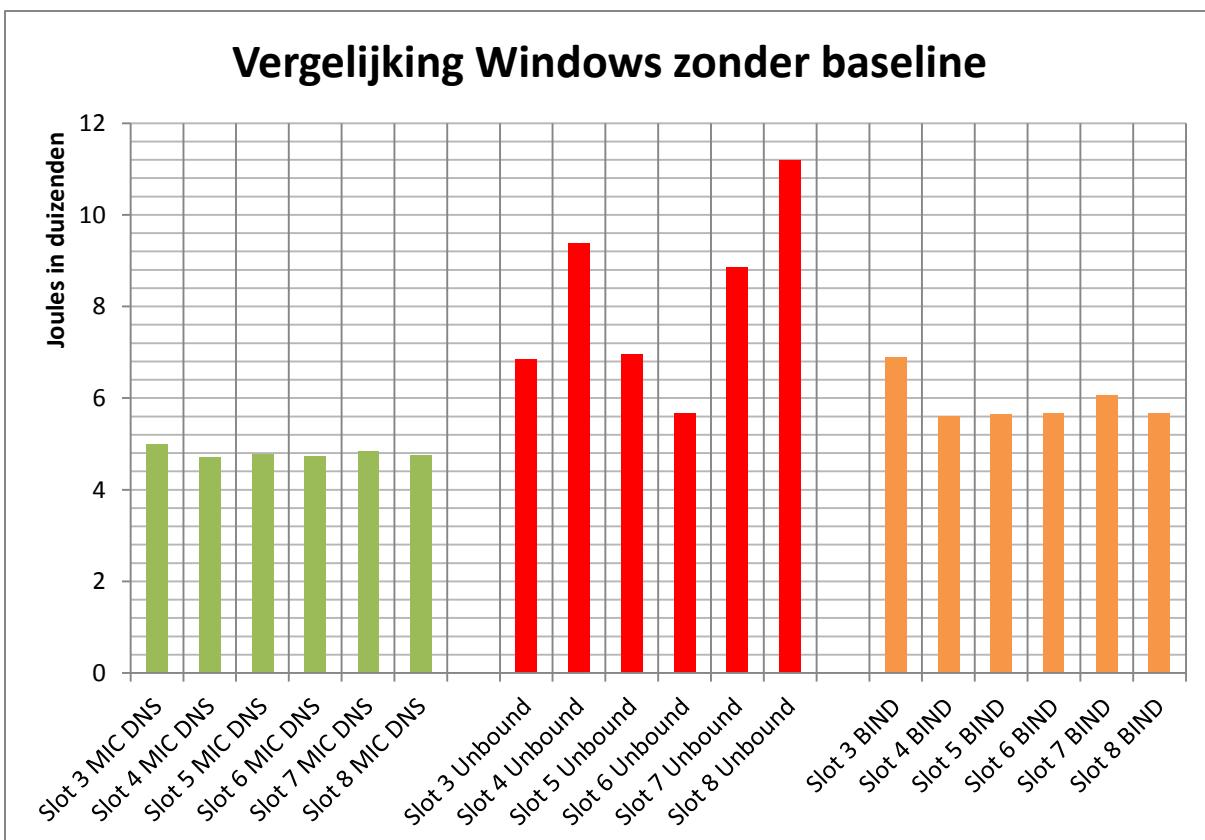
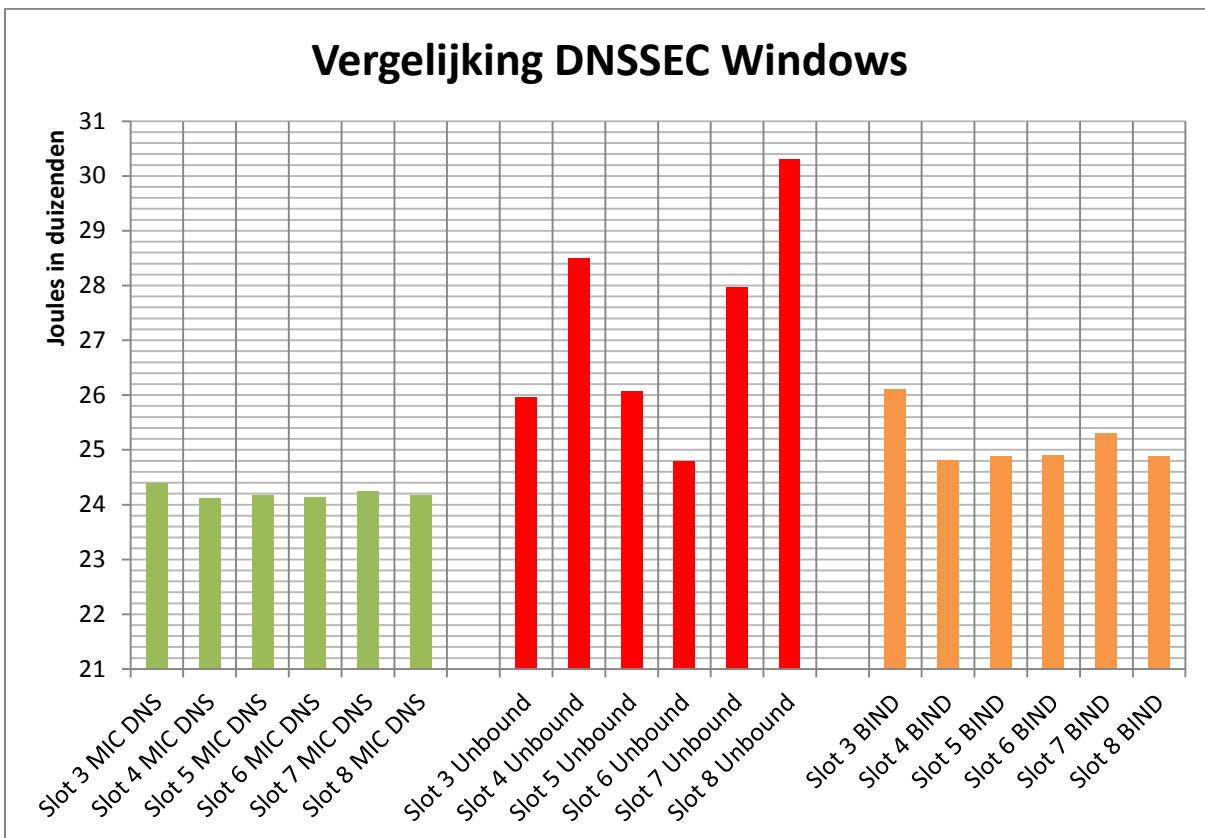
De grafiek van verbruiksgegevens ziet er als volgt uit:



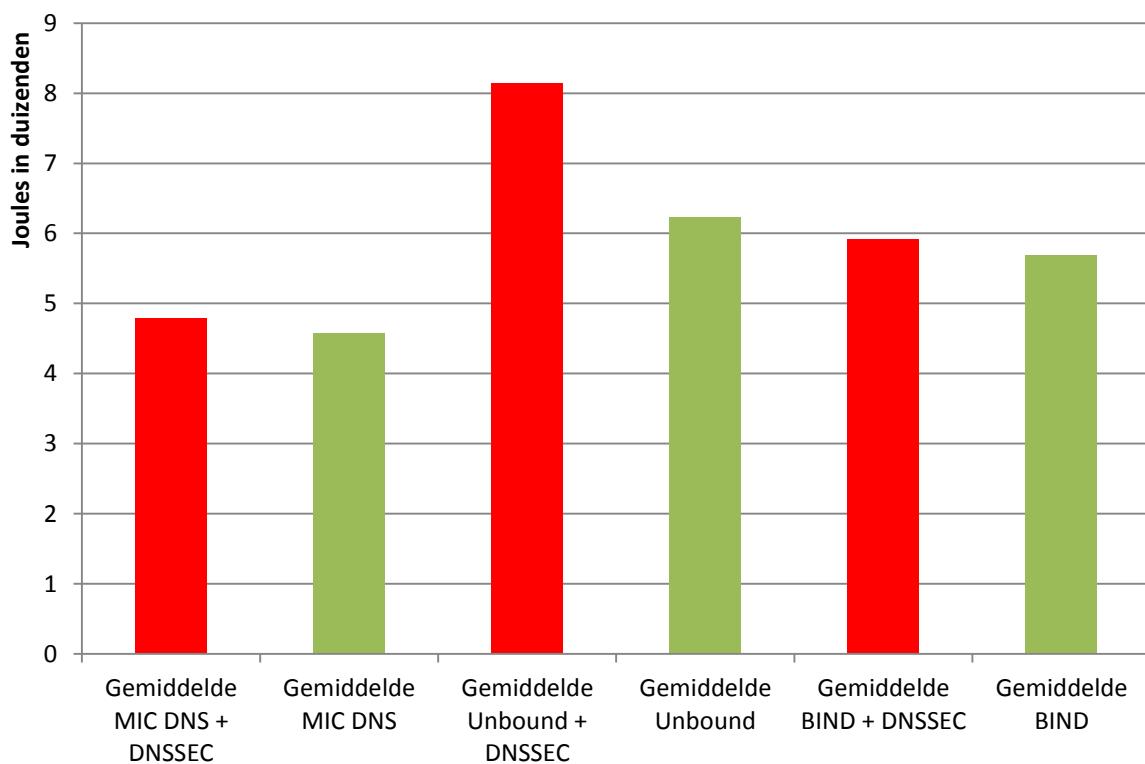
De channel_power grafiek ziet er als volgt uit:



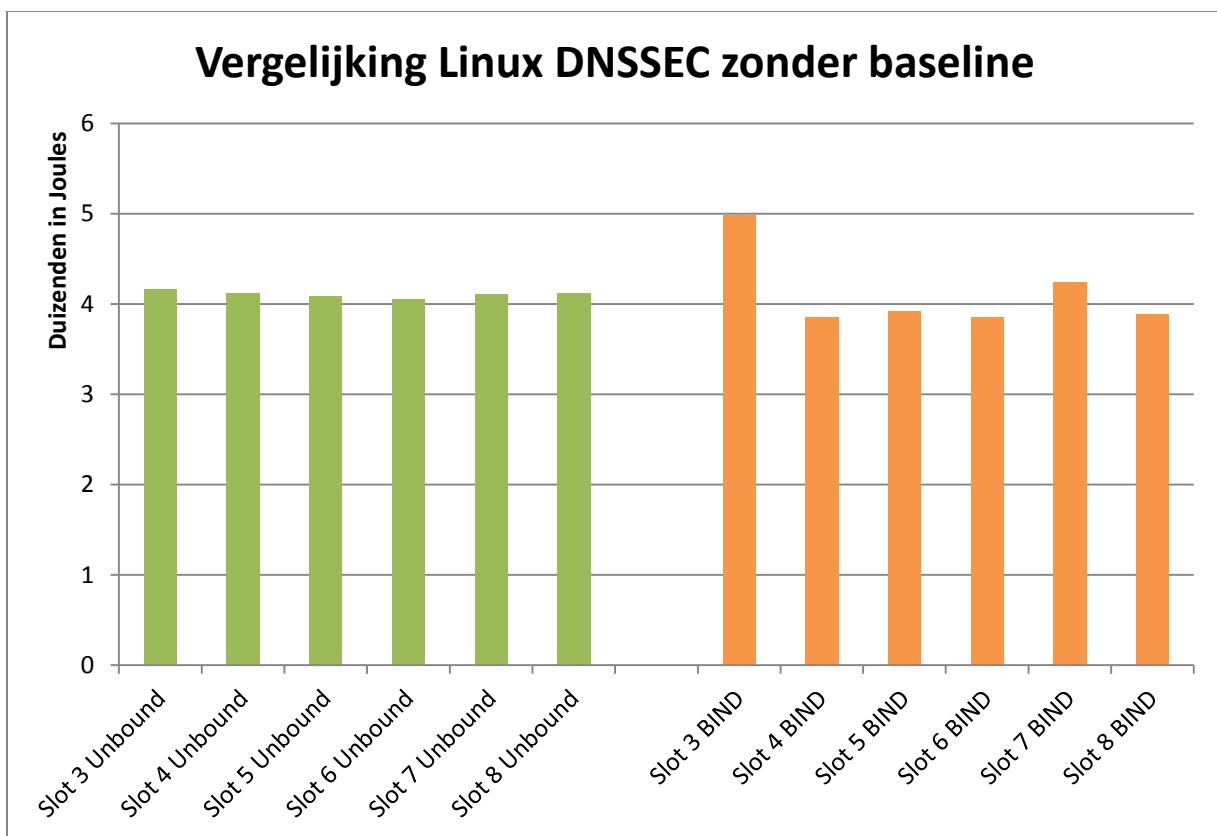
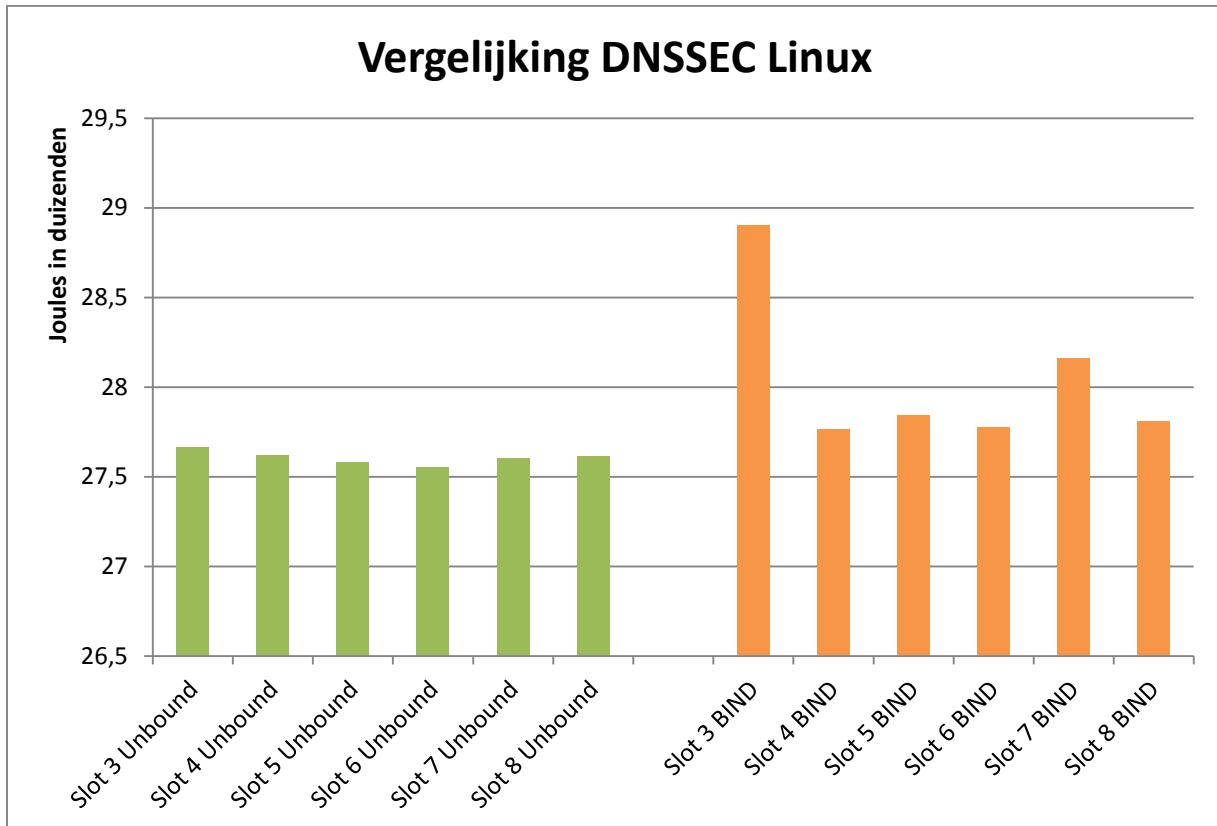
Vergelijkingen Windows



Windows DNSSEC vs Geen DNSSEC



Vergelijkingen Linux



Linux DNSSEC vs geen DNSSEC



Bijlage IX: Downsample tool verschil

Originele data Windows + Unbound + DNSSEC:

Original	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	4200,66079	6670,612	2410,242	757,5437	874,2376	986,992	36,506	55,69043	3091,907
Slot 2	4201,63058	6674,341	2407,901	759,8132	873,6223	987,0644	36,56447	55,80252	3091,92
Slot 3	6514,36445	11165,77	2419,819	765,2238	878,0567	985,7542	39,72134	58,96142	3091,584
Slot 4	7571,7314	12609,55	2425,996	774,783	879,8906	984,5017	40,7139	59,91348	3093,088
Slot 5	6588,20986	11205,89	2421,588	772,342	878,0983	983,4268	38,76352	57,95311	3081,014
Slot 6	6009,78238	10528,95	2419,07	770,3908	877,3229	983,5945	37,32338	56,51451	3074,718
Slot 7	7340,35645	12361,77	2412,671	771,361	875,8106	982,6679	38,78352	57,94575	3080,884
Slot 8	8383,12332	13624,2	2430,466	775,1093	881,1978	983,6344	40,35507	59,4385	3090,33

Downsampled data Windows + Unbound + DNSSEC:

Downsample	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	4207,299	6681,15	2414,074	758,7432	875,6254	988,5511	36,56399	55,77814	3096,795
Slot 2	4208,274	6684,89	2411,733	761,0125	875,0107	988,6245	36,62113	55,89148	3096,811
Slot 3	6521,853	11178,63	2423,656	766,4321	879,4457	987,3162	39,78072	59,05163	3096,463
Slot 4	7583,66	12628,65	2429,82	776,0038	881,278	986,0562	40,77734	60,00864	3097,981
Slot 5	6595,627	11218,42	2425,356	773,5501	879,4685	984,9837	38,82175	58,04129	3085,877
Slot 6	6017,582	10542,59	2422,904	771,6031	878,7111	985,1536	37,37896	56,60088	3079,591
Slot 7	7352,02	12380,47	2416,507	772,5847	877,2002	984,2208	38,84514	58,03925	3085,752
Slot 8	8390,852	13637,76	2434,234	776,3097	882,5692	985,1922	40,41359	59,52877	3095,207

Verschil Windows + Unbound + DNSSEC:

Difference	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	0,158%	0,158%	0,159%	0,158%	0,159%	0,158%	0,159%	0,158%	0,158%
Slot 2	0,158%	0,158%	0,159%	0,158%	0,159%	0,158%	0,155%	0,159%	0,158%
Slot 3	0,115%	0,115%	0,159%	0,158%	0,158%	0,158%	0,149%	0,153%	0,158%
Slot 4	0,158%	0,151%	0,158%	0,158%	0,158%	0,158%	0,156%	0,159%	0,158%
Slot 5	0,113%	0,112%	0,156%	0,156%	0,156%	0,158%	0,150%	0,152%	0,158%
Slot 6	0,130%	0,130%	0,158%	0,157%	0,158%	0,159%	0,149%	0,153%	0,158%
Slot 7	0,159%	0,151%	0,159%	0,159%	0,159%	0,158%	0,159%	0,161%	0,158%
Slot 8	0,092%	0,100%	0,155%	0,155%	0,156%	0,158%	0,145%	0,152%	0,158%

Originele data Windows + Microsoft DNS + DNSSEC:

Original	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	4200,66079	6670,612	2410,242	757,5437	874,2376	986,992	36,506	55,69043	3091,907
Slot 2	4201,63058	6674,341	2407,901	759,8132	873,6223	987,0644	36,56447	55,80252	3091,92
Slot 3	6514,36445	11165,77	2419,819	765,2238	878,0567	985,7542	39,72134	58,96142	3091,584
Slot 4	7571,7314	12609,55	2425,996	774,783	879,8906	984,5017	40,7139	59,91348	3093,088
Slot 5	6588,20986	11205,89	2421,588	772,342	878,0983	983,4268	38,76352	57,95311	3081,014
Slot 6	6009,78238	10528,95	2419,07	770,3908	877,3229	983,5945	37,32338	56,51451	3074,718
Slot 7	7340,35645	12361,77	2412,671	771,361	875,8106	982,6679	38,78352	57,94575	3080,884
Slot 8	8383,12332	13624,2	2430,466	775,1093	881,1978	983,6344	40,35507	59,4385	3090,33

Downsampled Windows + Microsoft DNS + DNSSEC:

Downsample	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	4207,299	6681,15	2414,074	758,7432	875,6254	988,5511	36,56399	55,77814	3096,795
Slot 2	4208,274	6684,89	2411,733	761,0125	875,0107	988,6245	36,62113	55,89148	3096,811
Slot 3	6521,853	11178,63	2423,656	766,4321	879,4457	987,3162	39,78072	59,05163	3096,463
Slot 4	7583,66	12628,65	2429,82	776,0038	881,278	986,0562	40,77734	60,00864	3097,981
Slot 5	6595,627	11218,42	2425,356	773,5501	879,4685	984,9837	38,82175	58,04129	3085,877
Slot 6	6017,582	10542,59	2422,904	771,6031	878,7111	985,1536	37,37896	56,60088	3079,591
Slot 7	7352,02	12380,47	2416,507	772,5847	877,2002	984,2208	38,84514	58,03925	3085,752
Slot 8	8390,852	13637,76	2434,234	776,3097	882,5692	985,1922	40,41359	59,52877	3095,207

Verschil Windows + Microsoft DNS + DNSSEC:

Difference	MEM1	CPU1	HDD 5V	HDD 12V	System 3.3V	PSU 12V Standby	PCI-e x16 3.3V aux	PCI-e x16 3.3V	PCI-e x16 12V
Slot 1	0,158%	0,158%	0,159%	0,158%	0,159%	0,158%	0,159%	0,158%	0,158%
Slot 2	0,158%	0,158%	0,159%	0,158%	0,159%	0,158%	0,155%	0,159%	0,158%
Slot 3	0,115%	0,115%	0,159%	0,158%	0,158%	0,158%	0,149%	0,153%	0,158%
Slot 4	0,158%	0,151%	0,158%	0,158%	0,158%	0,158%	0,156%	0,159%	0,158%
Slot 5	0,113%	0,112%	0,156%	0,156%	0,156%	0,158%	0,150%	0,152%	0,158%
Slot 6	0,130%	0,130%	0,158%	0,157%	0,158%	0,159%	0,149%	0,153%	0,158%
Slot 7	0,159%	0,151%	0,159%	0,159%	0,159%	0,158%	0,159%	0,161%	0,158%
Slot 8	0,092%	0,100%	0,155%	0,155%	0,156%	0,158%	0,145%	0,152%	0,158%

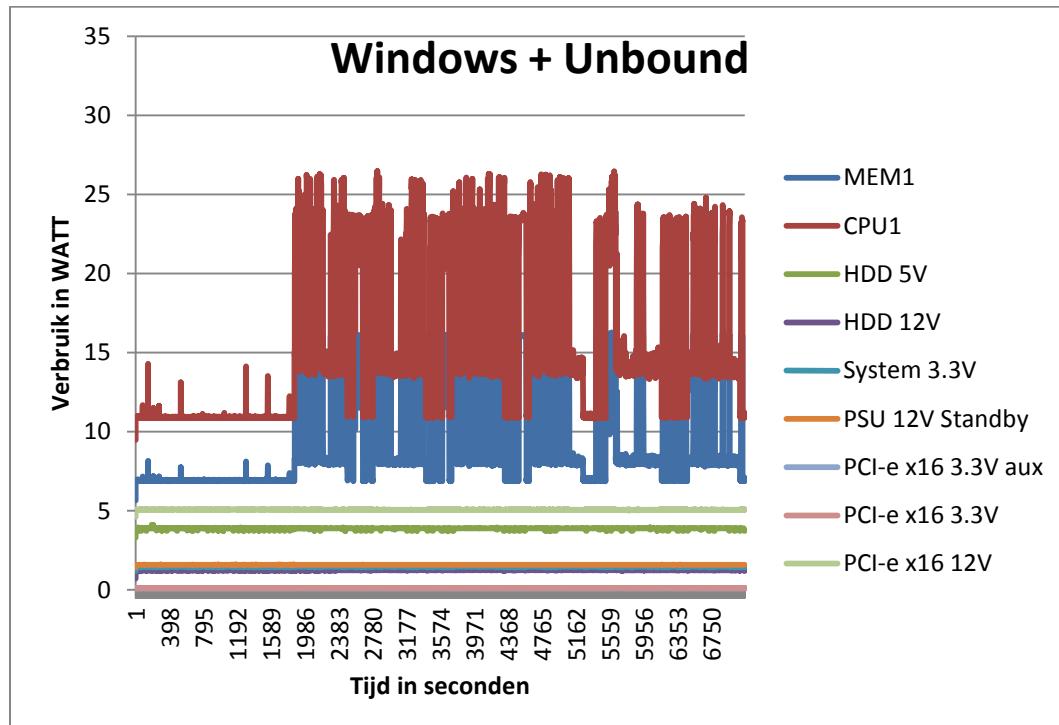
Bijlage X: MATLAB profiler

Profile Summary

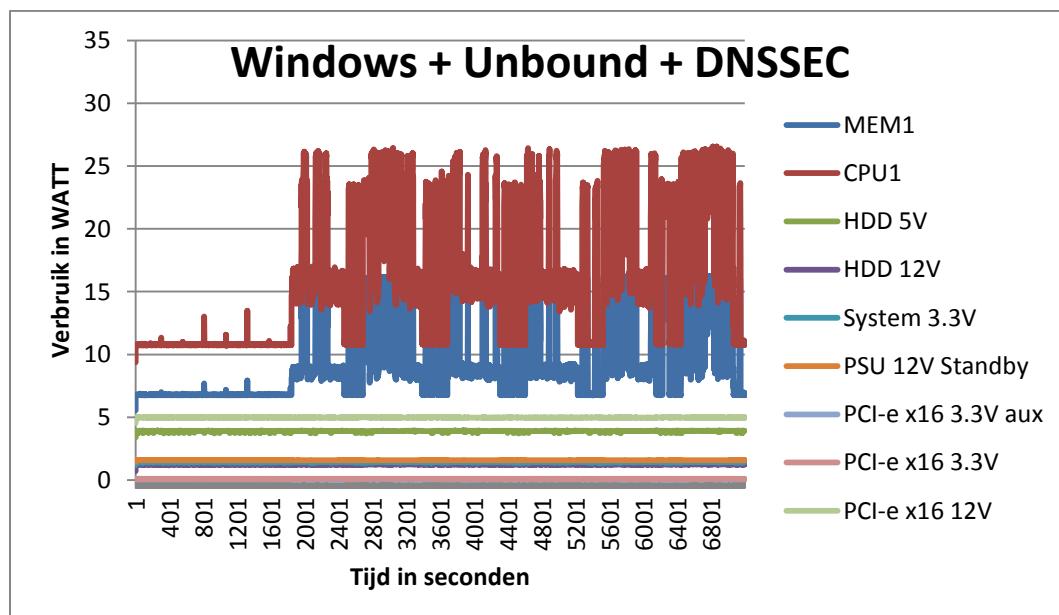
Generated 28-May-2014 14:11:40 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
main	1	14484.692 s	1.994 s	
read	1	8001.976 s	7663.964 s	
convert	1	6475.919 s	6475.615 s	
importdata	1	265.443 s	0.061 s	
importdata>LocalTextRead	1	264.388 s	0.039 s	
importdata>parse	1	261.819 s	11.762 s	
importdata>analyze	1	249.946 s	249.616 s	
str2num	357030	52.613 s	24.231 s	
str2num>protected_conversion	357030	28.381 s	28.381 s	
cell2mat	357030	19.957 s	19.957 s	
plotten	1	4.600 s	3.622 s	
fileread	1	2.525 s	2.525 s	
finfo	1	0.976 s	0.042 s	
bar	1	0.945 s	0.427 s	
finfo>getFileInfo	1	0.919 s	0.003 s	
datenum	18	0.494 s	0.379 s	
finfo>getVideoInfo	1	0.395 s	0.001 s	
...r.VideoReader>VideoReader.VideoReader	1	0.394 s	0.013 s	
...Reader.VideoReader>VideoReader.delete	1	0.330 s	0.322 s	
importdata>isvaliddata	1000	0.330 s	0.170 s	
finfo>getAudioInfo	1	0.321 s	0.002 s	
audioinfo	1	0.319 s	0.002 s	
...anager>PluginManager.getPluginForRead	1	0.307 s	0.000 s	
+file\private\mexPluginManager (MEX-file)	2	0.261 s	0.261 s	
measurement	1	0.200 s	0.010 s	
xychk	1	0.186 s	0.186 s	
usev6plotapi	1	0.168 s	0.168 s	
importdata>isdata	1000	0.160 s	0.160 s	

Bijlage XI: Analyse gedrag Unbound



Zoals te zien is in de grafiek zijn de 8 slots maar moeilijk terug te vinden bij Unbound op Windows. Dit gedrag komt niet terug op CentOS en komt ook niet voor bij de andere pakketten. In eerste instantie werd er waargenomen dat dit gedrag zich alleen vertoonde op het moment dat het systeem idle was, maar dit is in latere metingen niet opnieuw zo gegaan. Zoals in bovenstaande grafiek te zien is schiet het verbruik meteen omhoog. Onderstaande grafiek bevat de test met DNSSEC:



Het is vooralsnog onbekend waar dit gedrag vandaan komt. Er zou gekeken kunnen worden naar de timer resolutie van Windows. Het vermoeden bestaat dat dit er iets mee te maken kan hebben. Unbound was ook niet in staat alle queries te beantwoorden. De pcap file bevatte achteraf maar +/- 750 duizend antwoorden terwijl er +/- 1 miljoen antwoorden verwacht werden. Unbound op Windows is wederom de enige die zo weinig queries beantwoord heeft.

Hier de resultaten van de PCAP capture van de DNSSEC meting van unbound:

```
RSIZE_COUNTED:833314  
RCODE_NOERROR:704635  
RCODE_FORMERR:0  
RCODE_SERVFAIL:3028  
RCODE_NXDOMAIN:125611  
RCODE_NOTIMPL:0  
RCODE_REFUSED:40  
RCODE_UNKNOWN:0
```

Rsize counted geeft aan hoeveel antwoorden er gegeven zijn door Unbound, zoals te zien is zijn dat er iets meer dan 800 duizend. Hieronder de resultaten van Unbound op Linux:

```
RSIZE_COUNTED:1224123  
RCODE_NOERROR:1025731  
RCODE_FORMERR:0  
RCODE_SERVFAIL:13918  
RCODE_NXDOMAIN:184432  
RCODE_NOTIMPL:0  
RCODE_REFUSED:42  
RCODE_UNKNOWN:0
```

Zoals je hier ziet heeft Unbound op Linux 1,2 miljoen queries beantwoord. Dit is ook de trend die terug te zien is bij alle andere metingen die gedaan zijn. Alle metingen van Unbound op Windows vertonen echter het vreemde energieverbruik, en beantwoorden maar een relatief klein aantal queries.

Bijlage XII: Formulier eindbeoordeling

Formulier eindbeoordeling (bedrijf) Albert Hankel

Naam student: Remy Bien

Naam bedrijf: SURFnet

Criteria	--	-	+-	+	++
Organisatie van het werk					X
Taakopvatting					X
Zelfstandigheid (beslissen en handelen)				X	
Initiatief				X	
Contacten met collega's				X	
Contacten met leidinggevenden	*	*	*	*	*
Contacten met derden(klanten, opdrachtgevers)	*	*	*	*	*
Flexibiliteit			X		
Zelfkritiek				X	
Overtuigingskracht				X	
Stressbestendigheid					X
Schriftelijke uitdrukkingssvaardigheid				X	
Mondelinge uitdrukkingssvaardigheid				X	
Inzicht in eigen mogelijkheden en voorkeuren				X	
Heeft de student voldoende kennis				X	
Heeft de student voldoende vakinhoudelijke vaardigheden					X
Kwaliteit uitvoering van de werkzaamheden				X	
Kwantiteit uitvoering van de werkzaamheden					X
Toepasbaarheid resultaten in het bedrijf				X	
Begrip en inzicht in organisatie					X
Luistervaardigheid			X		

Datum:

05-06-2014

Opmerkingen stagebegeleider:

Remy toonde gedurende de gehele stage veel enthousiasme en inzet. We hadden hem een uitdagende opdracht gegeven die met enige aanpassingen ook geschikt zou zijn als Master-afstudeeropdracht. Het was een zelfstandige opdracht waarin hij veel samenwerkte met het SEFLab van de HvA en onderzoek deed naar het energiegebruik van DNS. We vroegen dus ook om vaardigheden die wat minder gebruikelijk waren voor de gemiddelde HBO-student (hiermee heb ik rekening gehouden in de beoordeling). Remy heeft zich echter uitstekend gehouden gedurende de opdracht en prima werk afgeleverd.

Wat ik hem mee wil geven is dat hij soms te snel wil en dan minder goed luistert en denkt te weten wat er moet gebeuren. Vaak gaat dat goed, soms niet en hadden onze suggesties hem tijd en moeite kunnen besparen.

Al met al zijn we erg tevreden over de afstudeerstage die Remy bij ons heeft gelopen.

Formulier eindbeoordeling (bedrijf) Roland van Rijswijk

Naam student: Remy Bien

Naam bedrijf: SURFnet

Criteria	--	-	+-	+	++
Organisatie van het werk					X
Taakopvatting					X
Zelfstandigheid (beslissen en handelen)					X
Initiatief					X
Contacten met collega's				X	
Contacten met leidinggevenden	nvt	nvt	nvt	nvt	nvt
Contacten met derden(klanten, opdrachtgevers)				X	
Flexibiliteit			X		
Zelfkritiek				X	
Overtuigingskracht				X	
Stressbestendigheid				X	
Schriftelijke uitdrukkingsvaardigheid				X	
Mondelinge uitdrukkingsvaardigheid				X	
Inzicht in eigen mogelijkheden en voorkeuren				X	
Heeft de student voldoende kennis					X
Heeft de student voldoende vakinhoudelijke vaardigheden				X	
Kwaliteit uitvoering van de werkzaamheden					X
Kwantiteit uitvoering van de werkzaamheden					X
Toepasbaarheid resultaten in het bedrijf				X	
Begrip en inzicht in organisatie				X	
Luistervaardigheid				X	

Datum:

10 juni 2014

Opmerkingen stagebegeleider:

Remy heeft tijdens de uitvoering van zijn opdracht laten zien dat hij uitstekend in staat is om zelfstandig een complexe opdracht uit te voeren waarbij meerdere partijen (SURFnet, SEFlab) betrokken zijn en waarbij cutting edge onderzoek gedaan wordt. De originele opdracht was wat SURFnet betreft ook geschikt voor een universitaire master; we vroegen van Remy dus meer dan van de gemiddelde HBO student verwacht mag worden; hier is rekening mee gehouden in de beoordeling, wat ons betreft heeft Remy laten zien dat hij niet terugschrikt om zijn tanden te zetten in een pittige klus.

Tijdens het uitvoeren van de opdracht werd duidelijk dat Remy kritisch is over zijn eigen werk en in staat is om – met enige hulp van ons – dat om te zetten in verbeteringen. Ook heeft hij laten zien goed te kunnen plannen en zijn plan aan te kunnen passen onder voortschrijdend inzicht (bv. dat metingen langer duren dan verwacht, en daar vervolgens zijn strategie op aanpassen zodat het toch binnen de beschikbare tijd kon worden uitgevoerd).

Een klein punt van kritiek is dat Remy niet altijd direct of niet goed luistert naar suggesties van zijn begeleiders. Hierop hebben we hem tijdens de opdracht aangesproken, waarna hij ons inziens zijn best heeft gedaan om hier verbetering in aan te brengen.

Afsluitend: we zijn zeer tevreden over zowel de uitvoering als de resultaten van Remy's afstudeerwerkzaamheden.