jamk.fi

Content Distribution Networks and GeoDNS load balancing

Juha Tawaststjerna

Opinnäytetyö
Toukokuu 2014

Tietotekniikan koulutusohjelma
Tekniikan ja liikenteen ala

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

KUVAILULEHTI

Tekijä(t) Tawaststjerna, Juha	Julkaisun laji Opinnäytetyö	Päivämäärä 07.05.2014
	Sivumäärä 79 + 25	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty (X)

Työn nimi
Content Distribution Networks and GeoDNS load balancing

Koulutusohjelma
Tietotekniikan koulutusohjelma

Työn ohjaaja(t)
Piispanen, Juha
Rantonen, Mika

Toimeksiantaja(t)
Jyväskylän Ammattikorkeakoulu- JYVSECTEC
Vatanen, Marko

Tiivistelmä

Opinnäytetyön toimeksiantajana toimi JYVSECTEC (Jyväskylä Security Technology), joka on vuonna 2011 alkanut kyberturvallisuusteknologian kehittämishanke. Opinnäytetyön tavoitteena oli toteuttaa toimeksiantajan pyytämät referenssiratkaisut käyttäjän geolokaatioon pohjautuvalle kuormantasaukselle. Työssä luotiin myös erillinen sisällönjakoverkko, jolla maantieteelliseen sijaintiin pohjautuvaa kuormantasausta voitiin testata.

Opinnäytetyö toteutettiin JYVSECTEC-projektin RGCE-verkkoon virtualisoituna. Työssä saavutettiin sille osoitetut tavoitteet. Työssä tutkittiin eri toteutustapoja maantieteelliselle kuormantasaukselle ja tutkimusten jälkeen toteutettiin DNS:ään pohjautuvat ratkaisut. Datakeskusten, eli sisällönjakoverkon sisäiseen osuuteen työssä ei ollut tarkoitus mennä syvemmin, vaan tarkoituksena oli luoda yksinkertainen esitys sisällönjakoverkon toteuttamiselle. Sisällönjakoverkko luotiin käyttäen Microsoftin ARR (Application Request Routing)-ohjelmistoa.

Lopputuloksena oli kaksi eri ratkaisua luoda kuormantasaus perustuen käyttäjän maantieteelliseen sijaintiin. Toinen järjestelmän luotiin käyttäen F5 BIG-IP GTM virtuaaliohjelmistoa ja toinen käyttäen Bind9-ohjelmistoa, joka muokattiin tukemaan geolokationaalisia merkintöjä. Näitä kahta järjestelmää voitiin lopulta vaihtaa ns. ”lennosta”, jolloin järjestelmien eri ominaisuuksia voitiin vertailla tarvittaessa.

Avainsanat (asiasanat)
Bind, CDN, DNS, F5 BIG-IP, GTM, kuormantasaus, sisällönjakoverkko

Muut tiedot

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

DESCRIPTION

Author(s)	Type of publication	Date
Tawaststjerna, Juha	Bachelor´s Thesis	07.05.2014
	Pages 79 + 25	Language Finnish
		Permission for web publication (X)

Title
Content Distribution Networks and GeoDNS load balancing

Degree Programme
Information Technology

Tutor(s)
Piispanen, Juha
Rantonen, Mika

Assigned by
JAMK University of Applied Sciences- JYVSECTEC project
Vatanen, Marko

Abstract

This bachelor's thesis was carried out for JYVSECTEC project. JYVSECTEC-project started in 2011 to maintain and develop cyber security infrastructure. The goal of this thesis was to implement the reference-models for geolocation based on load balancing for the client. A content distribution network was also created as part of the thesis to test the geolocation based load balancing.

This bachelor's thesis was implemented by virtualizing the systems in the JYVSECTEC project's RGCE-network premises. In this thesis research was carried out to search for different load balancing solutions. The research concluded with the usage of DNS based load balancing. Although this thesis addresses some parts of the content distribution networks, it is only a minor subject and the main focus was on the load balancing methods. The content distribution network was created by using Microsoft ARR (Application Request Routing) program.

The outcome was two different solutions for the load balancing based on user's geolocation. The first system was created by using the F5 BIG-IP GTM virtual edition and the second one by using Bind9-software which was patched to support geolocational information. In the end these two systems could be switched "on the fly" to compare the features if needed.

Keywords
Bind, CDN, DNS, F5 BIG-IP, GTM, load balancing, content distribution network

Miscellaneous

Sisällysluettelo

KUVIOT

LYHENTEET

ARR	Application Request Routing
BIND	Berkeley Internet Name Domain
CDN	Content Distribution Network
DIG	Domain Information Groper
DNS	Domain Name Service
FTP	File Transfer Protocol
FQDN	Fully Qualified Domain Name
GTM	Global Traffic Manager
GSLB	Global Server Load Balancing
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICANN	Internet Corporation for Assigned Number and Names
ID	Identifier
IIS	Internet Information Services
IP	Internet Protocol
ISP	Internet Service Provider
IXP	Internet Exchange Point
MX	Mail Exchanger
NAT	Network Address Translation
NS	Name Server
RAM	Random Access Memory
SLD	Second-Level Domain
SOA	Start of Authority
TLD	Top-Level Domain
TTL	Time to Live
URL	Uniform Resource Identifier
WWW	World Wide Web

1 TYÖN LÄHTÖKOHDAT

1.1 Toimeksiantaja

Jyväskylän ammattikorkeakoulu (JAMK) on vetovoimainen ja kansainvälinen korkeakoulu. JAMK:lla on toimipisteitä eri puolilla Jyväskylää sekä Saarijärven yksikkö Tarvaalassa. Yksiköitä JAMK:lla ovat mm. ammatillinen opettajakorkeakoulu, hyvinvointi, liiketoiminta ja palvelut sekä teknologia. Opiskelijoita JAMK:ssa on noin 8500. (Tutustu JAMKiin 2014.)

JYVSECTEC (Jyväskylä Security Technology)-hanke aloitettiin Jyväskylän ammattikorkeakoulussa syyskuussa vuoden 2011 aikana. JYVSECTEC on turvallisuusteknologian kehittämishanke, jota ovat rahoittaneet mm. Euroopan aluekehitysrahasto (EAKR) ja Keski-Suomen liitto. Muita kumppaneita hankkeessa on JAMK, Airbus Defence and Space, Ajeco Oy, Descom Oy, Relator Oy, sekä Jyväskylän seudun kehittämisyhtiö (Jykes Oy). Projektin puitesopimuskumppaneita ovat Cassidian Oy ja TeliaSonera. (Jyväskylä Security Technology, 2014)

JYVSECTEC kehittämishankkeen tarkoituksena on rakentaa ja ylläpitää kyberturvallisuuden kehitysympäristöä Realistic Global Cyber Environment (RGCE). RGCE-ympäristössä tuotetaan tutkimus-, kehitys- ja koulutuspalveluita yhteistyöverkon käytettäväksi. Kyseistä projektia koordinoi Jyväskylän ammattikorkeakoulu (JAMK). (Jyväskylä Security Technology, 2014)

1.2 Tavoitteet

Opinnäytetyön tavoitteena oli luoda RGCE-verkkoon (Realistic Global Cyber Environment) toimeksiantajan pyytämät referenssiratkaisut maantieteelliselle sijainnille pohjautuvalle kuormantasaukselle, joka voi pohjautua joko käyttäjän käyttämän DNS-palvelimen IP-osoitteeseen tai suoraan käyttäjän käyttämään IP-osoitteeseen. Kuormantasausratkaisujen lisäksi tavoitteena oli luoda CDN (Content Distribution Network), eli sisällönjakoverkko. Kyseisellä sisällönjakoverkolla tulisi olemaan kolme reunapistettä: Euroopassa Ranskassa, Aasiassa Japanissa ja Amerikassa Yhdysvalloissa.

Työn kannalta oleellinen osuus on itse maantieteelliseen sijaintiin pohjautuva kuormantasaus, eikä niinkään palvelinkeskusten sisällä tapahtuva liikennöinti. Työssä ensin oli tarkoituksena selvittää eri valmistajien ratkaisuja globaalille kuormantasaukselle ja selvityksen jälkeen toteuttaa kyseiset ratkaisut. Työssä toteutettiin myös täysin toimiva sisällönjakoverkko (CDN, Content Distribution Network), joskin työn painopiste olikin globaalissa kuormantasauksessa. Työn lopputuloksena syntyviä geolokaatioon pohjautuvia kuormantasaajia on tarkoitus pystyä vaihtamaan keskenään, jolloin toimivuuksia voidaan tarvittaessa vertailla.

2 Content Distribution Network

2.1 Yleistä

Content Distribution Networkit (CDN) koostuvat monista palvelimista ja päätepisteistä ympäri maapalloa, jotka auttavat mm. nettisivujen latausnopeuksissa, sekä sisällön latausajoissa. Tämä etu saavutetaan tuomalla haluttu sisältö lähemmäksi loppukäyttäjää. Content Distribution Network on käytännössä joukko peilipalvelimia halutulle sisällölle, joilla saadaan kyseinen sisältö tarjottua loppukäyttäjille mahdollisimman läheiseltä palvelimelta. (Stutzbach, D., Zappala, D., Rejaie, R.)

Riippuen käytössä olevasta teknologiasta, jolla CDN tarjoaa sisällön, CDN voidaan karkeasti jakaa Peer-to-Peer (P2P) ja Peering/Private-pohjaisiin toteutiksiin. (Stutzbach, D., Zappala, D., Rejaie, R.)

2.2 Peer-to-Peer CDN

P2P-pohjaisissa CDN-ratkaisuissa haluttu sisältö välimuistitetaan käyttäjän päädyssä heti, kun jokin muu käyttäjä ottaa yhteyttä ja pyytää sisältöä. Tämän jälkeen ensimmäinen mainittu käyttäjä toimii palvelimena. Tämä sallii sisällönjaon monista lähteistä yhden lähteen sijasta. Tässä vaiheessa on kuitenkin hyvä ottaa huomioon, että P2P-verkoissa sisältö voi tulla mistä päin vain maailmaa, jolloin sisällön suhteen optimaalisinta reittiä ei tulla saamaan. (Li, J. 2007)

2.3 Peering/Private CDN

Peering-pohjaisessa CDN-ratkaisussa käytetään erinäisiä (edge) päätepistepalvelimia, jotka Content Delivery Networkin tarjoaja tarjoaa. Näitä päätepisteitä on yleensä ympäri maailmaa ja ne on yhdistetty useisiin internetpalveluiden tarjoajiin (ISP), jolloin saadaan nopeampi ja parempi saatavuus halutulle sisällölle. Tämän lisäksi Content delivery networkeihin asennetaan erilaisia algoritmeja ja ominaisuuksia, joilla saavutetaan optimaalisin reitti sisällön siirtoon. (What is CDN?. 2013)

Optimaalisin reitti ei ole kuitenkaan aina lyhin reitti, sillä mm. mahdollisissa liikenteentukostiloissa lähin palvelin saattaa vastatakin hyvin hitaasti käyttäjän pyyntöihin, jolloin voi olla järkevää siirtää lähialueen liikennettä myös vähän kaukaisempiin sivupisteisiin muita linkkejä pitkin. Käyttäjän kannalta suurempi määrä päätepisteitä (edge) on parempi, sillä mitä enempi näitä päätepisteitä on, sitä suuremmalle määrälle ihmisiä saadaan suhteellisen hyvin toimiva palvelu. (Verma, D. 2002, 1.3)

Päätepisteet pyrkivät tarjoamaan mahdollisimman monia palveluja käyttäjälle suoraan, sillä nämä reunapisteet tarjoavat peilatun sisällön palvelusta, jonka origin-palvelin tarjoaa. Ongelmia tulee siinä vaiheessa, kun reunapalvelin ei sisälläkään origin-palvelimen tarjoamaa sisältöä, vaan se on ladattava reunapisteelle origin-palvelimelta, jolloin sisältö joudutaan välimuistittamaan sivupisteelle. Tätä prosessia kutsutaan hit-ratioksi, suurempi hit-ratio tarkoittaa suurempaa palvelunsaantia suoraan reunapisteeltä, jolloin loppukäyttäjä nauttii nopeammasta toiminnasta. (Verma, D. 2002, 1.3.1)

Perinteisesti CDN-palveluntarjoajat auttavat nettisivujen suorituskyvyssä ja pienentämällä palvelinten kuormaa siirtämällä staattisen sisällön, kuten kuvat, css:n ja javascriptin reunapalvelinpisteille. Tämän lisäksi kyseiset tarjoajat auttavat myös isojen tiedostojen tarjoamisessa, sekä videon suoratoistamisessa (streaming) nopeuttaen näiden palveluiden latausnopeuksia. (What is CDN?. 2013)

Nykypäivänä käyttäjät vaativat yhä parempaa suorituskykyä verkoilta, jolloin web-optimoinnin tärkeys kasvaa entisestään. Tämän vuoksi monet CDN-palveluntarjoajat ovat alkaneet tarjoamaan myös dynaamista sisältöä reunapalvelimilta suoraan. Nykypäivänä CDN-palveluntarjoajat ovat ottaneet käyttöönsä myös monia päätepisteiden optimointeja: minify, gzip-kompressointi ja muita ratkaisuja vanhojen välimuistiratkaisujen ohelle, joilla tarjotaan palvelut entistä nopeammin loppukäyttäjille. (What is CDN?. 2013)

Suurin osa Content Delivery Networkien tarjoajista käyttää Peering-tekniikkaa saavuttaakseen nopean sisällöntarjonnan ympäri maailmaa. Näitä palveluntarjoajia ovat

mm. Akamai Technologies, Amazon CloudFront, Limelight Networks ja Microsoft Azure CDN. Tarjoajia on toki muitakin, mutta edellämainitut ovat vain muutamia esimerkkejä. (What is CDN?. 2013)

2.3.1 Staattinen ja dynaaminen sisältö

Dataa, joka siirtyy käyttäjän ja palvelimen välillä, on kahdenlaista: staattista ja dynaamista. Staattinen data on hitaasti muuttuvaa tai muuttumatonta. Dynaaminen data saattaa puolestaan muuttua hyvinkin nopeaa tahtia. Näiden määritteleminen on useinkin kiinni siitä, kuinka nopeita linkit CDN:n sisällä toimipisteiden välillä ovat. (Verma, D. 2002, 1.3.1)

Dataa, jonka tyyppi on staattinen, on helppo tarjota nopeammin loppukäyttäjille käyttäen CDN:ää. Näissä tapauksissa päätepalvelimet lataavat kyseisen datan origin-palvelimelta omaan välimuistiin, josta se voidaan nopeasti tarjota loppukäyttäjille. Jos data on puolestaan luonteeltaan dynaamista, päätepalvelinten on jatkuvasti varmistettava, että kyseinen data on yhtenevää origin-palvelimen kanssa. Tämä prosessi syö resursseja ja tietoliikennekaistaa, jolloin CDN-verkon kannalta on hyvä miettiä mitä dynaamista dataa säilytetään, vai tarjotaanko se ainoastaan suoraan origin-palvelimelta. (Verma, D. 2002, 1.3.1)

2.4 Kuormantasaus ja käyttäjän Geolokaatio

2.4.1 Kuormantasaus

Kuormantasausta on olemassa monenlaista. Pääsääntöisesti kuormantasaus keskittyy kommunikaatiojärjestelmiin ja palvelimiin. Kommunikaatiojärjestelmissä tavoitteena on kuormantasaus useiden eri järjestelmien välille. Tietokonejärjestelmissä kuormantasaus on puolestaan esimerkiksi kahden prosessorin välillä tapahtuvaa tasausta. (Held, G. 2010, 5.2.1)

Edustakuormantasaaja asetetaan koneeksi ennen varsinaisia palvelimia, johon käyttäjä ottaa yhteyttä. Kuormantasaajan avulla monta eri palvelinta saadaan näkymään yhtenä ainoana palvelimena, joskin taustalla on oikeasti useita eri palvelimia. Kyseis-

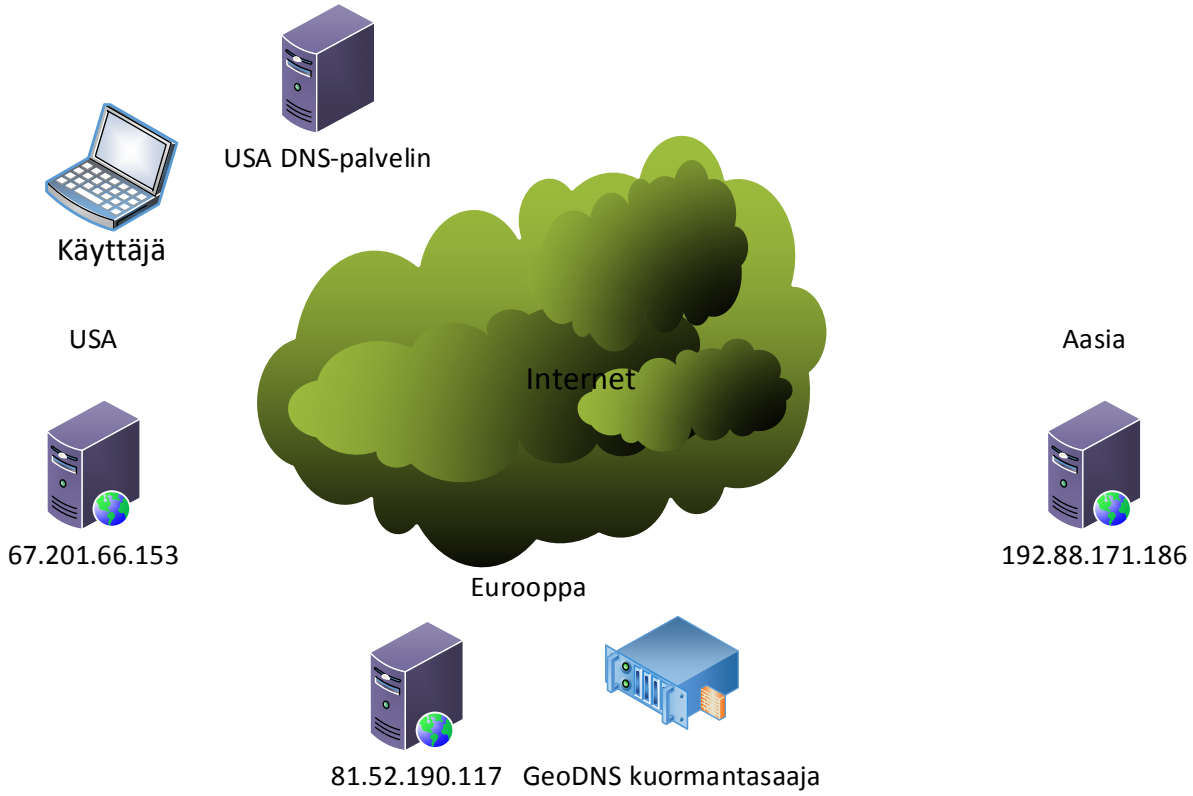
tä palvelinjoukkoa mainostetaan ulospäin käyttäen ainoastaan yhtä IP-osoitetta, jonka oikeasti omaa itse kuormantasaaja. Kyseinen toimenpide on loppukäyttäjälle täysin näkymätön, sillä kuormantasaaja reitittää kaiken liikenteen sääntöihinsä perustuen jollekin palvelimelle, ja usein tallentaa muistiin tietyt tunnistetiedot, jotta kaikki tulevatkin kyselyt samalta loppukäyttäjältä reititetään samalla palvelimelle koko keskustelun ajan. (Verma, D. 2002, 2.2.1)

Tässä vaiheessa on myöskin hyvä huomioida järjestelmän eräs heikkous. Koska kaikki liikenne tulee kulkemaan tämän yhden kuormantasaajalaitteen lävitse, tästä kohtaa voi syntyä heikoin lenkki (SPOF- single point of failure). Kyseisen laitteen kaatuessa tai tukkeutuessa mikään yhteys ei toimi, jolloin myös liikennöinti CDN:n kanssa estyy suurimmilta osin. (Verma, D. 2002, 2.2.1)

Kuormantasauksessa on myös mahdollista, että yhteen DNS-nimeen, esimerkiksi. www.camibo.com on osoitettu enemmän kuin yksi IP-osoite. Näissä tapauksissa käyttäjän pyyntö täytyy osoittaa yhteen näistä osoitteista. Pyyntö yleensä osoitetaan johonkin näistä IP-osoitteista riippuen kuormantasaus säännöstöstä. Tätä prosessia kutsutaan DNS kuormantasaukseksi. (Held, G. 2010, 5.2.5)

Kuvion 1 tapauksessa oletetaan, että “Käyttäjä” sijaitsee USA:ssa. Kun ”Käyttäjä” ottaa yhteyden sivustolle www.camibo.com, ottaa hän ensimmäisenä Internetin kautta yhteyden USA DNS-palvelimelle, joka puolestaan ottaa yhteyden camibo.com autoritääriselle nimipalvelimelle, joka tässä tapauksessa sijaitsee Euroopassa GeoDNS kuormantasaajalla. GeoDNS kuormantasaaja käsittelee siten USA:ssa sijaitsevan DNS-palvelimen DNS nimenselvityspyynnön osoitteelle www.camibo.com, joka tässä tapauksessa palauttaa IP-osoitteen 67.201.66.152, sillä tämä on geograafisesti lähin palvelin USA:n DNS-palvelimen osalta. USA:n DNS-palvelin palauttaa vastauksen ”Käyttäjälle”. Kaikki liikennöinti tämän DNS nimenselvitysprosessin jälkeen tapahtuu suoraan ”Käyttäjältä” Internetiin ja Internetistä kyseiselle USA:n palvelimelle. Vastaavasti, jos ”Käyttäjä” sijaitsisi esimerkiksi Euroopassa ja käyttäisi Euroopassa sijaitsevaa DNS-nimipalvelinta, palauttaisi GeoDNS kuormantasaaja osoitteen 81.52.190.117. GeoDNS kuormantasaaja käsittelee ainoastaan DNS pyyntöjä, eikä

myöhempää liikennettä kuljeteta enää kyseisen laitteen läpi, niin kuin paikallisen kuormantasaajan tapauksessa.



Kuvio 1 GeoDNS kuormantasaus

Kuviossa 1 on hyvä ottaa huomioon myös se, että kyseisten IP-osoitteiden takana voi hyvinkin olla useita palvelimia, jolloin kuormantasausta suoritetaan vielä kyseisten IP-osoitteiden takana. Näissä tapauksissa on hyvä ottaa huomioon myös pysyvyys, jottei jokaista pyyntöä tarvitse prosessoida uudestaan. Toinen huomionarvoinen seikka on, että IP-osoite palautetaan perustuen käytettyyn DNS-palvelimeen, jolloin esimerkiksi USA:ssa sijaitseva käyttäjä, joka käyttää Euroopassa sijaitsevaa nimipalvelinta saakin Eurooppalaisen IP-osoitteen sivustolle www.camibo.com.

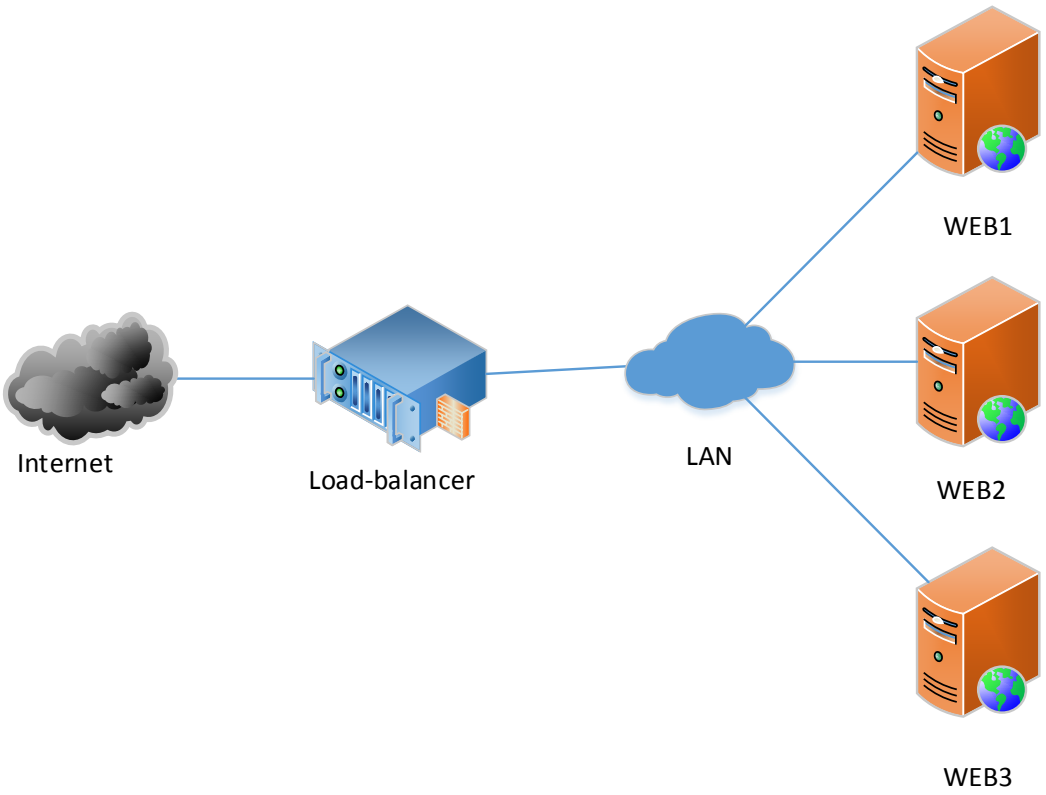
2.4.2 Sisäverkon kuormantasaus ja NAT

Hyvin usein palvelimien edessä toimii erillinen kuormantasauslaite, joka reitittää kaiken saapuvan liikenteen halutuille palvelimille. Usein kyseinen kuormantasaaja toimii myös NAT:ia suorittavana laitteena, etenkin jos organisaatiolla on enemmän palvelimia kuin julkisia IP-osoitteita. (Verma, D. 2002, 2.3)

Kuormantasaaja pystyy havaitsemaan uudet yhteydet vanhoista tutkimalla SYN-lipullisia paketteja TCP-kehyksistä. Kun uusi yhteys avataan, NAT:ia suorittava laite luo uuden merkinnän tauluun, johon tulee yhteyttä ottavan asiakkaan IP- ja portti-tiedot. Tämän lisäksi merkitään myös palvelin, johon kyseinen käyttäjä reititetään. Tiedot poistetaan kyseisestä taulusta, kun yhteys puretaan. Yhteyden purku tapahtuu joko TCP-kehyksen RST-lipun ollessa päällä, FIN-lippuun tai mahdollisesti, jos yhteys on ollut käyttämättömänä tarpeeksi pitkään. (Verma, D. 2002, 2.3)

Koska NAT:n suorittaminen on hyvin resurssi-itensiivistä, joissain tapauksissa käytetään kahta laitetta kyseisen prosessin suorittamiseen. Toinen laite suorittaa itse kuormantasauksen eri laitteiden välillä ja toinen suorittaa NAT-prosessia. Tälläisessä mallissa saadaan skaalautuvuutta verkolle, mutta lisälaitteisto tuo lisäkustannuksia ja ylimääräistä konfigurointia. (Verma, D. 2002, 2.2.1)

Kuviossa 2 on esitettynä aiemmin selitetyn teorian mukainen ratkaisumalli. Kaikki tiettyyn IP-osoitteeseen / DNS-nimelle tuleva liikenne lähetetään ensin kuormanta-saajalaitteelle, joka puolestaan jakaa kuorman haluttujen parametrien mukaisesti.



Kuvio 2 Paikallinen kuormantasaaja

2.4.3 Pysyvyys

Kun kuormantasausta suoritetaan, olisi järkevää että jokainen samasta lähteestä tuleva pyyntö reititettäisiin aina samalle palvelimelle. Tätä prosessia kutsutaan pysyvyydeksi (stickiness). Tämä prosessi on erittäin tärkeää etenkin http-liikennöinnissä, sillä esimerkiksi nettikauppasivuston ”ostoskärryn” täytyy pysyä tallennettuna, vaikka uusia HTTP pyyntöjä lähetetään jatkuvasti. Koska http on alun perin yhteydetön protokolla, ongelmaa täytyy ratkaista muilla keinoin. Mahdollisia keinoja ongelmanratkaisuun ovat keksit (joista on enemmän asiaa kappaleessa 2.6) url rewriting ja shared session storage. (Verma, D. 2002, 2.3.1)

Toinen vaihtoehto istunnon ylläpitämiselle keksien lisäksi on ”url rewrite”. Url rewrite perustuu ratkaisuun, jossa jokaiselle käyttäjälle näytetään ensin sama alkuperäinen sivu, esimerkiksi ”www.camibo.com”, jonka perään lisätään satunnainen satunnaisesti tuotettu uniikki tunnistin (unique identifier). Käytännössä sivusto ”www.camibo.com/linkki/johonkin/” muuttuu muotoon

”www.camibo.com/uid/linkki/johonkin/”. Kyseinen UID-tunniste säilyy koko istunnon ajan, paitsi jos käyttäjä kirjoittaa sivuston nimen ”www.camibo.com” uudestaan, jolloin sivusto generoi uuden UID-tunnisteen. (Verma, D. 2002, 2.3.1)

Viimeinen tapa suorittaa pysyvyyttä on käyttää ”yhteistä jaettua tallennetta” (shared session storage). Tässä tavassa palvelimet eivät käytä omia tilataulujaan, vaan jakavat yhteisen verkossa jaetun tilataulun. Kyseinen tilataulu voi olla jonkin palvelimen suorittama prosessi tai tiedostopalvelimella sijaitseva tiedosto, jota kaikki HTTP-palvelimet käyttävät istunnon tallentamiseen. (Verma, D. 2002, 2.3.1)

2.4.4 DNS reititystavat

Karkeasti katsottuna reitityksen tai kuormantasauksen voi sanoa perustuvan kahteen eri päätyyppiin. Ensimmäinen tapa suorittaa kyseistä prosessia on aktiivinen ja toinen tapa passiivinen. Aktiivinen tapa suorittaa prosessia aiheuttaa lisää kuormaa verkolle, sillä verkon palvelimilta ja laitteilta on jatkuvasti kysyttävä niiden mahdollista tilaa. Passiivinen tapa ei ota kantaa onko jokin laite tai palvelu saavutettavissa, vaan pyyntö siirretään palveluun perustuen konfiguraatioon. (Verma, D. 2002, 4.1)

Aktiivinen kuormantasaus

Aktiivisessa tavassa reitittää asiakkaat perustuu kuormantasaajalaitteen määrittelemiin aktiivisiin parametreihin. Nämä aktiiviset parametrit tutkitaan käyttäen erilaisia seurantapaketteja: kuormantasaajalaite lähettää palvelimille ja verkon aktiivilaitteille jatkuvasti tietyin väliajoin kyselyitä, joihin nämä palvelimet ja aktiivilaitteet vastaavat lähettäen sen hetkisen tilannetietonsa. Aktiivisena parametrina toimii esimerkiksi tietoverkkolinkkien kuorma, tai palvelimen resurssien käyttöaste. Käyttäjien pyynnöt ohjataan lopulta näiden aktiivisten parametrien perusteella loppupisteisiin. (Verma, D. 2002, 4.1.1)

Passiivinen kuormantasaus

Passiivinen tapa siirtää asiakkaat oikeaaseen paikkaan CDN:ssä toimii siten, että kuormantasaajalaitteella on käytössään taulu, josta tarkastetaan parhaaksi soveltuva kohde parametrien perusteella. Parametreina voi toimia monikin asia: asiakkaan lo-

kaatio ja CDN-päätepisteen lokaatio, sekä näiden välinen etäisyys. Etäisyyttä voidaan mitata kanssa erilaisin parametrein: tietystä IP-osoitealueesta saapuvat pyynnöt reititetään tietylle palvelimelle, tai etäisyys voidaan ottaa suoraan reitityksen ”metric”-parametrista. (Verma, D. 2002, 4.1.2)

2.5 Välimuisti

2.5.1 Välimuisti yleisesti

Välimuistitus toimii tekniikkana, jossa aiemmin hankittu tieto siirretään välimuistiin, jos tietoa haetaankin uudestaan. Asiakas-palvelin-ympäristössä on muutamia syitä käyttää välimuistia. Ensimmäisenä syynä on se fakta, että tämä pienentää viiveitä tai toisin sanoen latenssia: Data pyydetään lähemmältä pisteeltä asiakkaan näkökulmasta. Toisena hyvänä syynä on tietoverkkojen liikenteen vähentäminen. Data ladataan suoraan asiakkaan koneelta välimuistista, samalla nopeuttaen kokonaisprosessia. (Held, G. 2010, 5.1)

Välimuistia on monenlaista ja se voi sijaita myös eri kohteissa. Mahdollisia sijaintipaikkoja ovat mm. RAM, flash-muisti, kiintolevy tai jopa näiden erilaisia kombinaatioita. Edellä olevista välimuistien sijaintipaikoista käytetään kuitenkin usein kuvaavampia termejä, kuten esimerkiksi selaimen välimuisti, proxy-palvelimen välimuisti, sovelluksen välimuisti tai jopa gatewayn välimuisti. (Held, G. 2010, 5.1)

2.5.2 Proxyt ja välimuisti

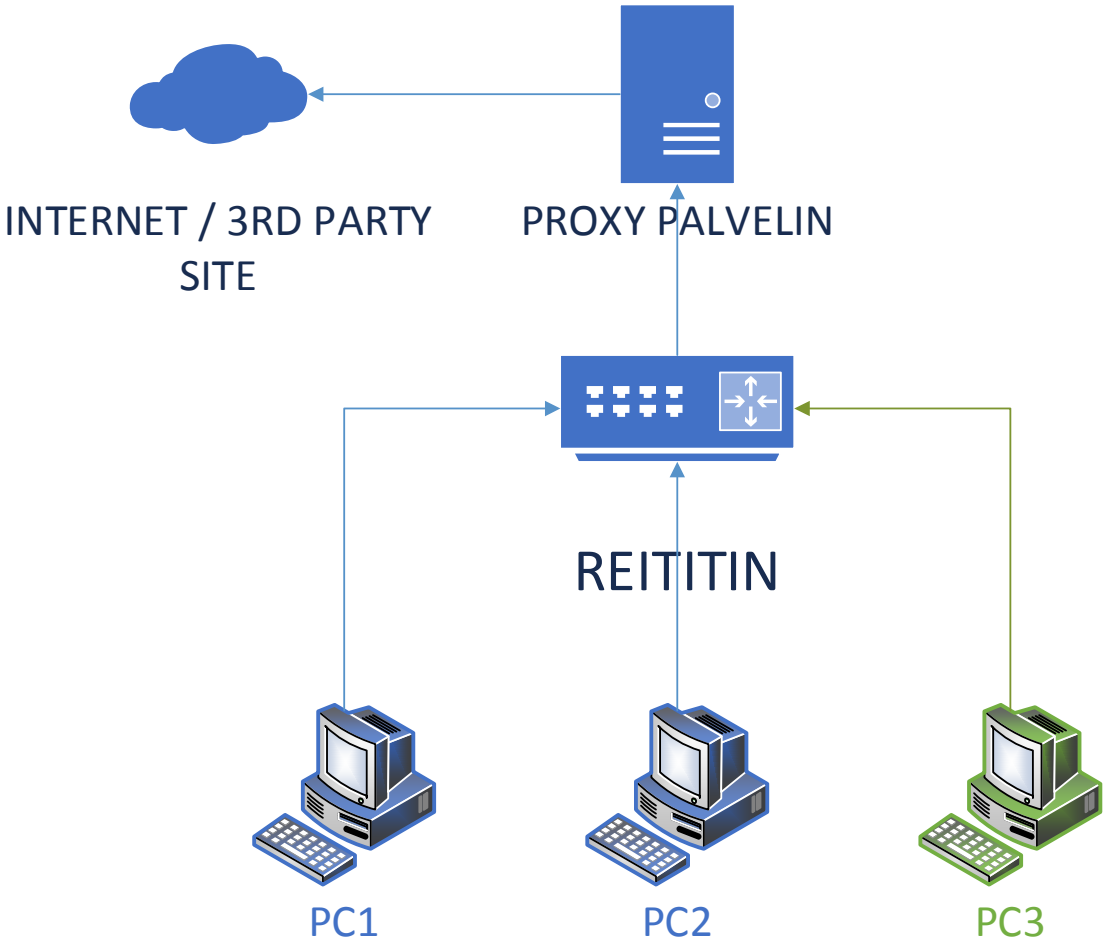
Proxyn välimuisti eroaa käyttäjän selaimen tai koneen välimuistista siten, että proxyn välimuisti on yleensäkin ottaen monen käyttäjän käytössä kerralla. Kun yksi käyttäjä pyytää informaatiota proxy:n kautta, tämä pyyntö siirretään proxy:n ja mahdollisesti myös käyttäjän välimuistiin. Kun taas seuraava käyttäjä pyytää tätä samaa informaatiota, sitä ei pyydetäkään enää alkuperäiseltä sivulta, vaan se ladataan tälle käyttäjälle suoraan proxy:ltä. Proxy-palvelin sijaitseekin usein verkon reunalla, mutta verkkolaitteiston takana. Proxy-ominaisuudet on usein myös mahdollista ympätä kiinni reitittimiin ja muihin verkon aktiivilaitteisiin. (Held, G. 2010, 5.1.2)

Edustaproxy

Edustaproxy on yleisimmin käytetty proxy-muoto. Usein edustaproxy-palvelin asetetaan verkon palomuurin taakse ja vastaa sisäverkon asiakkaiden pyyntöihin. Tämä ratkaisu luo tietoturvaa, sillä joskus asiakkailla ei ole pääsyä ulkoverkkoon muutoin kuin edustaproxy:n kautta. Tämä ratkaisu luo myös yksityisyyttä, sillä kaikki tulevat pyynnöt näkyvät tulevan proxy-palvelimelta. Edustaproxy yksinkertaisesti vaihtaa IP-kentän tiedot näyttämänä siltä, että pyynnöt tulevat proxy-palvelimelta.

Useimmat proxy-palvelimet pystyvät tekemään IP-muunnoksen monille eri protokollille. Kaikkein useimmin käytetyt protokollat tämän suhteen ovat mm. HTTP, SHTTP ja FTP. Vaikkakin tietoturva ja yksityisyys ovat hyviä syitä käyttää edustaproxy-palvelimia, suurin syy käyttää näitä nykypäivänä on yksinkertaisesti HTTP-pyyntöjen välimuistitus (caching). Käyttäjän pyytäessä yhtä sivua, tämä sivusto tallennetaan proxyn välimuistiin. Tällä tavalla useampi käyttäjä pystyy saamaan saman sisällön nopeammin, kuin jos jokainen käyttäjä lataisi erikseen kyseisen sisällön, tällä tavalla säästetään myös kaistaa. (Aulds, C. 2001, 12. How Proxies Work)

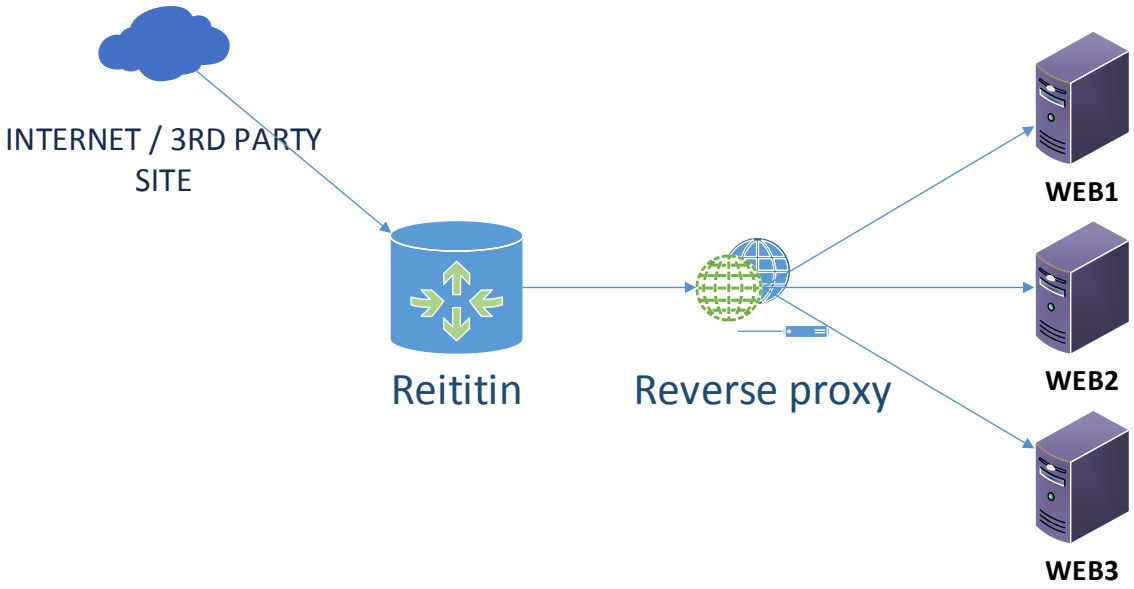
Kuviossa 3 on hiukan havainnollistettu edustaproxyn toimintaa. PC-laitteiden pyynnöt ohjataan täten aivan normaalisti ulos verkosta, mutta esimerkiksi HTTP-pyynnöt menevät proxy-palvelimen kautta. Proxy-palvelin puolestaan muuntaa kehyksen ja lähettää paketin eteenpäin internetiin.



Kuvio 3 Proxy havainnollistus

REVERSE PROXY

Toisin kuin edustaproxy, REVERSE-proxy vastaakin internetistä tuleviin pyyntöihin järjestelmille, jotka sijaitsevat verkon palomuurin sisäpuolella. Usein reverse-proxyn tehtävänä onkin reitittää sisäverkkoon tarkoitetut HTTP-pyynnöt oikeaaseen sisäverkon palvelimeen perustuen konfiguraatioon ja valittuun algoritmiin. Reverse proxy toimii täten puhtaammin tietoturvan luojana kuin edustaproxy. Reverse proxy tarjoaa mahdollisuuden päästä käsiksi sisäverkon materiaaliin, joka olisi muutoin ulkoverkon käyttäjien tavoittamattomissa. Kuviossa 4 on havainnollistettu reverseproxyn toimintaa. (Aulds, C. 2001, 12. How Proxies Work)



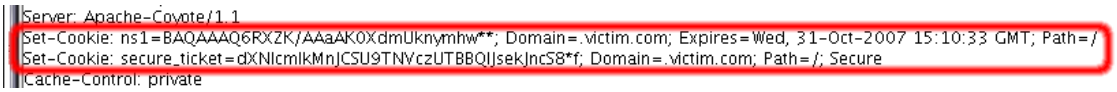
Kuvio 4 Reverse proxy

2.6 Keksit

Yksinkertaisuudessaan keksi (cookie) on lyhyt tekstiviesti, joka voidaan asettaa käyttäjän tietokoneen kiintolevylle. Alun perin kekseillä kierrettiin HTTP-protokollasta johtuva ”yhteydetön” tila, joka teki vaikeaksi istunnon ylläpitämisen. Keksin pääasiallinen tehtävä on sisällyttää tilannetietoa koskien asiakas-palvelin-yhteyksiä. Näitä yhteyksiä ovat mm. internet kauppojen ”kärrytiedot”. Vaikka alun perin tilanne oli tämä, on kekseille tullut monia muitakin tehtäviä vuosien saatossa. Kekseillä voidaan nykyään tarkkailla mm. käyttäjän käyttämää aikaa tietyllä WEB-sivulla, sekä muuta statistiikkaa. Jos WEB-sivujen ylläpitäjä ei sisällytä keksille vanhenemisaikaa, keksi luodaan, mutta sitä ei tallenneta käyttäjän kiintolevylle vaan se poistetaan selaimen sammutuksen yhteydessä. Tälläista keksiä kutsutaan ”nonpersisten” keksiksi. Nonpersistent-keksiä käytetään usein, kun jokin toimenpide vaatii tietoturvaa, ja siksi keksi asetetaan kuolemaan hyvinkin nopeasti. (Held, G. 2010, 6.2.13)

Kuviossa 5 on näkyvillä keksin vastaanottaminen käyttäjän päädyssä. Domain-kohdasta näemme sivuston nimen, Expire-kohdasta päivän jolloinka keksi vanhenee, sekä Path-attribuutin, joka kertoo mihinkä osaa sivustoa kyseinen keksi pätee. Tässä tapauksessa Path on ”/”, jolloinka keksi on käytössä koko sivustolla. Pathin määrittelemän alueen ulkopuoliset sivut eivät voi tätä keksiä lukea. Tätä aluetta voidaan laa-

jentaa Domain-attribuutilla. Domain-attribuutti sallii keksin käytön useammilla palvelimilla, jos nämä kuuluvat samaan domainiin. (Held, G. 2010, 6.2.13.1)



Kuvio 5 set-cookie (owasp-cookie)

Jokaisella keksillä täytyy olla oma uniikki nimensä. Saman nimiset keksit aiheuttavat sen, että toinen näistä ylikirjoitetaan. Kekseillä täytyy olla myös päättymispäivä, jonka jälkeen ne vanhenevat. Tämän eri palveluntarjoajat kiertävät tosin käyttämällä hyvin pitkiä päättymispäivä, jotka käytännössä estävät vanhenemisen. Vanhentuneet keksit poistetaan selaimesta, kun käyttäjä vierailee taas sivustolla, joka kyseisen keksin alun perin kirjoittikin. (Held, G. 2010, 6.2.13.2)

Tässä vaiheessa on hyvä huomioida se, että vaikka käyttäjä käyttäisikin proxy-palvelinta HTTP-liikenteen välittämiseen, niin kekseillä voidaan tunnistaa yksittäinen käyttäjä. Keksit sisältävät uniikin ID:n (tunnisteen) jokaiselle eri käyttäjälle. Keksillä voidaan laskea sivustolla vierailleet käyttäjät, sekä tuliko kyseinen käyttäjä sivustolle vielä uudestaankin. Keksien avulla voidaan saada jonkinlaista dataa uniikeista käyttäjistä. (Held, G. 2010, 6.2.13.4)

2.7 Domain Name Service (DNS)

2.7.1 Yleistä

Domain Name Service (DNS) on tapa, jolla tietokoneet muuttavat URL:t, kuten ”www.camibo.com” IP-osoitteiksi. Järjestelmä on luotu ihmisten takia, sillä on helpompi muistaa osoite www.camibo.com, kuin IP-osoite 81.62.190.150. DNS antaa täten mahdollisuuden käyttää domain-nimiä palveluiden etsimiseen, joskin käyttäjä voi myös käyttää IP-osoitteita halutessaan (Easttom, C., Palladino, S. 2012, 15 DNS) DNS-järjestelmä on suhteellisen yksinkertainen. Asiakkaan tietokoneen DNS resolver (selvittäjä) lähettää pyynnön DNS-palvelimelle, joka vastaa pyyntöön lähettämällä vastauksen, selvittämällä vastauksen jos sitä ei välimuistista löydy tai virheilmoituksen. Käyttäjän pyynnössä kulkee domain-nimi, joka menee DNS palvelimelle. DNS-palvelin tutkii sitten tätä DNS-nimeä vastaavan IP-osoitteen ja lähettää sen käyt-

täjälle, jos se sen löytää. Prosessia kutsutaan ”nimenselvitykseski” tai ”IP-osoitteen nimenselvitykseksi”. (Easttom, C., Palladino, S. 2012, 15 DNS)

Maailmassa on olemassa 13 root-autoritääristä juuri DNS-palvelinta. Nämä palvelimet ovat DNS:n päälähteitä. Tämän lisäksi on olemassa monia muita autoritäärisiä DNS-palvelimia, joita pitävät yllä mm. yksittäiset yritykset, yliopistot, internetpalveluntarjoajat ym. organisaatiot. Nämä kaikki DNS-palvelimet keskustelevat ja vaihtavat DNS-tietoja keskenään. (Easttom, C., Palladino, S. 2012, 15 DNS)

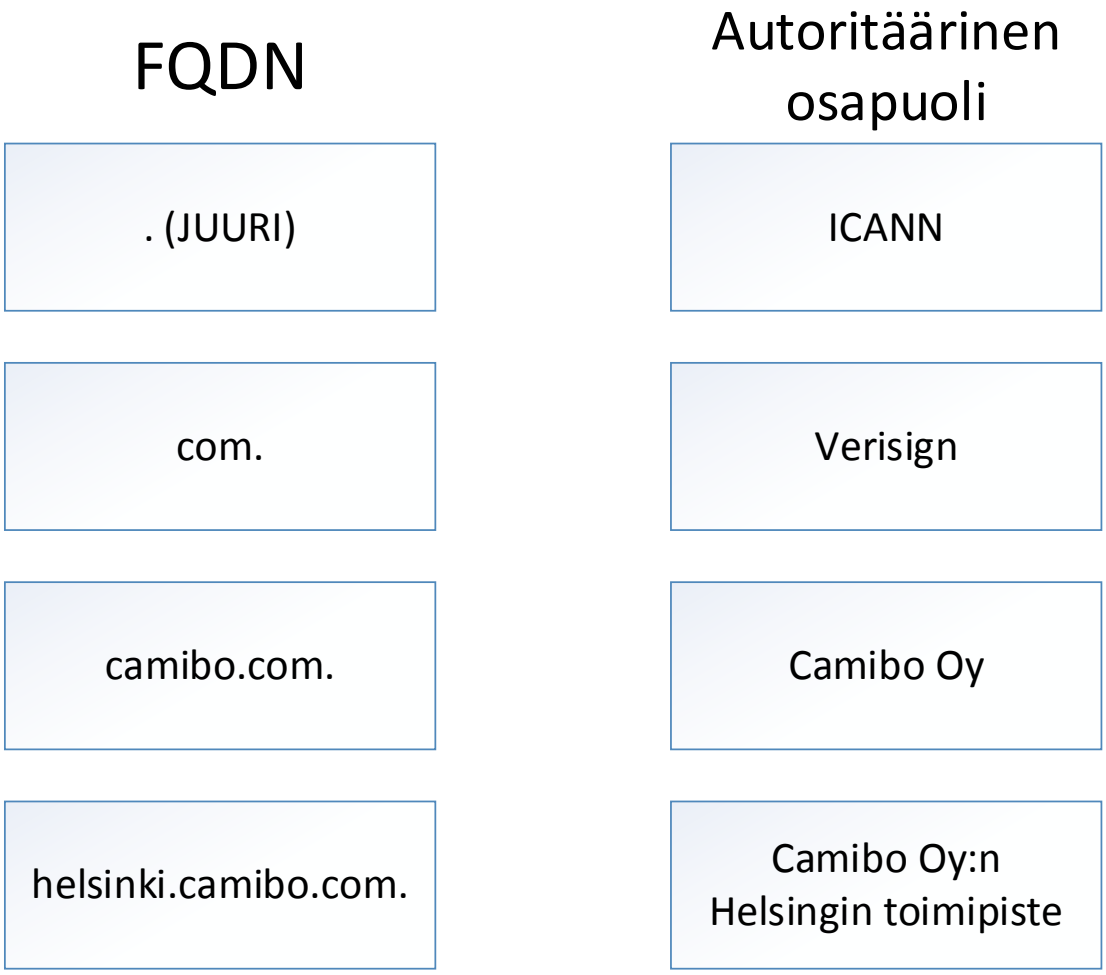
2.7.2 Domain nimien delegaatio

DNS käyttää puumaista hierarkiaa nimeämisissä. Puun ”yläpäässä” on juuri, josta polveutuvat ensin Top-Level Domainit (TLDs) ja näiden jälkeen Second-Level Domainit (SLDs). Näitä seuraa monia muita alemmat tason domaineja. TLD ovat jaettuina kahdeksi eri tyypeiksi: Generic Top-Level Domaineiksi (gTLD) ja Country Code Top-Level Domaineiksi (ccTLD). gTLD:t tunnetaan päätteistä kuten: .com, .edu, .net, .org etc. ccTLD:t tunnetaan päätteistä kuten: .fi, .sv, .tv, .uk etc. Esimerkiksi ”camibo.com” koostuu kahdesta osasta, SLD:nä on camibo, ja TLD:nä .com. Osoite on muotoa ”sld.tld”. Tämän lisäksi on hyvä huomioida, että on olemassa myös Thid-Level Domaineja, jotka ovat erityisen tärkeitä ccTLD tapauksissa. (Aitchison, R. 2011, chapter 1)

2.7.3 Domain authority

Domain nimet ovat moniosaisia ja jokainen osa domain-nimeä on asetettu jonkun tahon alaiseksi. Esimerkiksi osoite ”helsinki.camibo.com” voi olla jopa neljän eri tahon alaisuudessa, kun tämä nimi pilkotaan osiksi. Ensimmäisenä osana tulee juuri, joka on ICANN nimisen järjestön alaisena. ICANN vastaa domain nimien hierarkiasta ja näiden kaupallistamisesta ja kilpailuttamisesta. gTLD:t ovat ICANN:n alaisuudessa, mutta jaettuina ”accredited registrar” standardein hyväksytyille tahoille. ccTLD:t ovat puolestaan jaettuina valtioille. Seuraava osuus DNS-nimeä on .com. .com:n autoritaarinen organisaatio saa puolestaan jakaa .com alaisia nimiä muille organisaatioille ja antaa autorisaation kyseiselle organisaatiolle jakaa nimiä taas tämän uuden nimen alapuolella. camibo.com voi olla yrityksen Camibo Oy alaisuudessa ja näin edespäin. (Aitchison, R. 2011, chapter 1)

Kuviossa 6 on havainnollistettuna FQDN (Fully Qualified Domain Name). Kuviossa on havainnollistettuna aiemmin selitetty teoria. Kuviosta on hyvä huomioida se seikka, että vaikka selaimeen kirjoitetaan ”www.helsinki.camibo.com” ilman perään tulevaa pistettä, teknisestä FQDN:ään kuuluu päätepiste merkkaamaan juurta. (Aitchison, R. 2011, chapter 1)



Kuvio 6 DNS Autoritäärisuus

2.7.4 DNS implementaatio

Internetin DNS palvelimien implementaatio on täysin samanlainen kuin paperein määritellyt DNS autorisaatiot. Jokaisella autorisoidulla organisaatiolla on oma DNS-ohjelmistonsa jokaisella hierarkiatasolla ja tämän ohjelmiston ajaminen palvelimella on täysin kyseisen organisaation vastuulla. Juurinimipalvelimet (root name servers) ovat tärkein osa koko internet-infrastruktuuria. Kun jokin nimipalvelin maailmalla ei löydä infoa jostain domainista, se kysyy tätä informaatiota yhdeltä juuripalvelimista.

Näitä juuripalvelimia on monia ja tieto niistä jaetaan kaikille DNS-ohjelmistoille käyttäen erityistä ”zone” tiedostoa. (Aitchison, R. 2011, chapter 1)

2.7.5 DNS & välimuisti

DNS järjestelmä käyttää välimuistia hyväkseen hyvin paljon, jotta järjestelmää saataisiin yksinkertaisemmaksi ja nopeammaksi toiminnaltaan. Haetut DNS hakukyselyt tallennetaan välimuistiin ja jos kyseistä informaatiota haetaan jossakin välissä uudestaan, saadaan se nopeasti välimuistista. DNS välimuistilla on oma TTL arvonsa, jonka mukaan vanhoja tietoja poistetaan säännöllisesti. DNS-kyselyprosessi voi kulkea monellakin tapaa, mutta prosessin voi kuvata askel askeleelta seuraavanlaisesti:

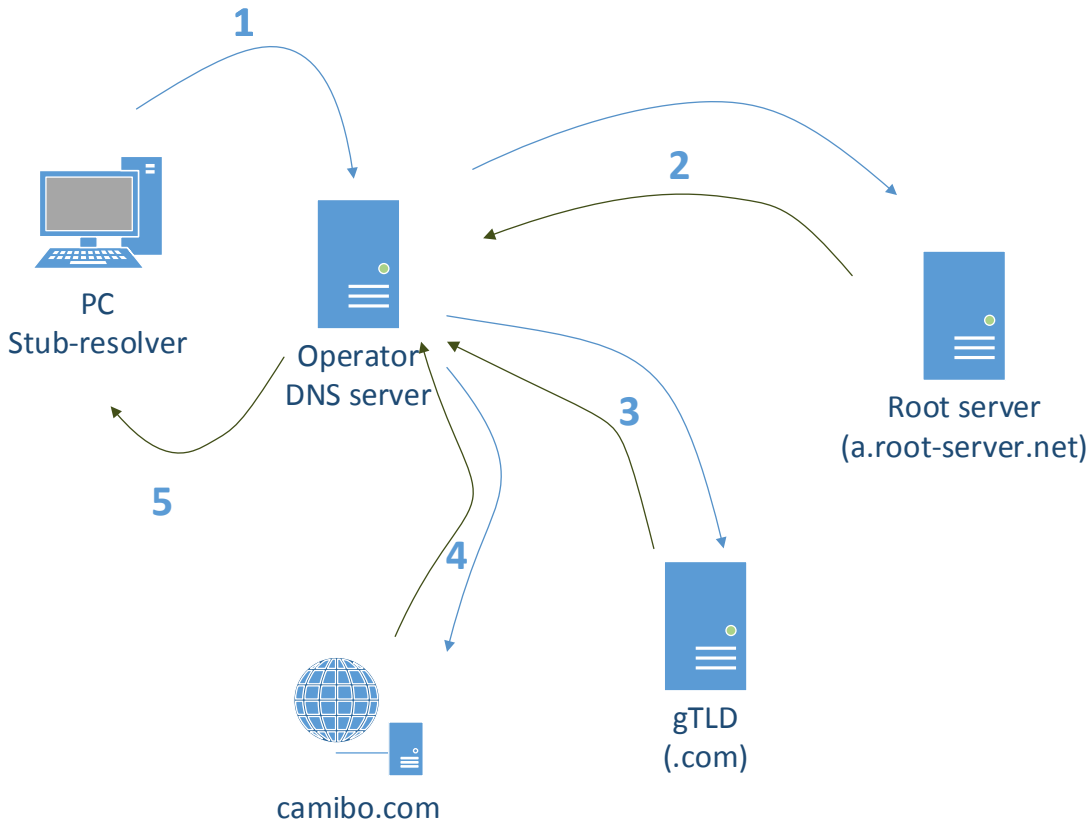
1. Käyttäjän etsiessä tiettyä osoitetta esim. www.camibo.com, selain tarkastaa ensin välimuistista löytyykö kyseistä DNS nimeä vastaava IP osoite paikallisesti. Jos kyseistä dataa ei löydy, kutsutaan prosessia nimeltä selvittäjä (resolver). (Aitchison, R. 2011, chapter 1)

2. DNS selvittäjä (resolver) on hyvin monimutkainen ohjelma ja siksi nykyään monet järjestelmät käyttävät standardin mukaista, mutta pienempää versiota resolverista, jota kutsutaan ”caching stub-resolveriksi” tai ”stub-resolveriksi”. Tämä pienennetty ohjelma osaa käyttää myös välimuistia ja jos vastaus löytyy välimuistista, palautetaan se selaimelle. Jos vastausta ei löydy välimuistista, kysely lähetetään verkossa eteenpäin koneelle konfiguroidun DNS-palvelin parametrin mukaan. Huom! Monissa kotiverkoissa kyseinen DNS-palvelin toimii käyttäjän reitittimessä, jolloin prosessi on hiukan erilainen kolmoskohdan suhteen. (Aitchison, R. 2011, chapter 1)

3. Käyttäjän pyyntö saapuu lopulta DNS palvelimelle (resolver), joka on täysin toiminnallinen selvittäjä. Kyseisellä palvelimella on usein paljon resursseja käytössään ja se toimii selvittäjänä monille käyttäjille. Tämäkin palvelin etsii vastauksia ensin omasta välimuististaan. Tätä palvelinta kutsutaan usein myös nimillä ”caching name server” tai ”recursive name server”, sillä kyseinen palvelin tarjoaa palvelua hyvin monelle käyttäjälle, jolloin myös sen välimuistista löytyy paljon tavaraa. Erikoisuutena tässä palvelimessa on se, että

> jos käyttäjän hakuun ei löydy vastausta, niin kysely lähetetään eteenpäin DNS autoritaarisessa hierarkiassa. Kyseiseen kyselyyn saadaan (authorative answer) autoritaarinen vastaus, joka lähetetään alkuperäiselle käyttäjälle ja välimuistitetaan paikallisesti, jotta tulevat kyselytkin saadaan ohjattua perille nopeammin. (Aitchison, R. 2011, chapter 1)

Kuviossa 7 on kuvallisesti aiemmin selitetty teoria. Käyttäjän halutessa sivuston ”www.camibo.com”, tutkii hän ensin välimuistinsa kuten teorian kohdassa yksi ja kaksi, kun dataa ei löydy siirrytään kuvassa kohtaan numero yksi. Seuraavaksi PC:n ”Stub-resolver” lähettää pyynnön eteenpäin operaattorin DNS-palvelimelle, eli siirrymme teorian kohdassa kohtaan numero kolme. Operaattorin DNS-palvelin etsii vastausta osoitteelle ”www.camibo.com” omasta välimuististaan ja kun sitä ei löydy, siirrytään kuvion kohtaan kaksi, eli pyyntö lähetetään eteenpäin juurinimipalvelimille ”root server”. Juurinimipalvelin puolestaan vastaa algoritmeihinsa perustuen jonkun osoitteen .com autoritääriselle palvelimelle. Kuvan kohdassa kolme operaattorin DNS-palvelin lähettää nyt selvityspyynnön ”camibo.com”-osoitteesta gTLD autoritääriselle palvelimelle .com, johon kyseinen autoritäärinen palvelin vastaa kertomalla osoitteen ”camibo.com” DNS-palvelimelle. Nyt viimein kuvion kohdassa neljä operaattorin palvelin lähettää pyynnön selvittääkseen ”asiakasosoitteen” nimeltä ”www.camibo.com” camibo.com:n DNS-palvelimelle, johon kyseinen DNS-palvelin vastaa antamalla IP-osoitteen ”asiakaskoneelle” www.camibo.com. Tämän vastauksen operaattorin DNS-palvelin tallentaa omaan välimuistiin ajanjaksoksi, jonka se on itse pakottanut välimuistien vanhenemisajaksi tai kuinka pitkän TTL-arvon camibo.com on kyseiselle osoitteelle määrittänyt. Kohdassa viisi lopulta kyseinen IP-osoitetieto lähetetään myös asiakkaalle, jolloin asiakas voi ottaa yhteyttä osoitteeseen ”www.camibo.com”. Tämä kyseinen nimenselvitysprosessi on paljon nopeampi muiden käyttäjien kohdalla, sillä muut asiakkaat saavat vastauksen suoraan operaattorin DNS-nimipalvelimen välimuistista, eikä kohtia 2-4 tarvitse käydä uudestaan vasta kuin DNS-tietueen poistuttua palvelimen välimuistista.



Kuvio 7 DNS nimenselvitysprosessi ja välimuistitus

2.7.6 Zonet

Nimipalvelimet tukevat ominaisuutta, jota kutsutaan ”zoneksi”. Zone on käytännössä ”zone tiedosto”, joka kääntää domain nimet operatiivisiksi kokonaisuuksiksi, kuten yksittäisiksi asiakkaiksi, mail palvelimiksi yms.. Zone tiedosto käyttää RR:iä (resource records) tietyn domain nimen kuvaukseen.

Zone tiedosto sisältää usein seuraavan tyyppisiä RR:iä:

- Dataa, joka selittää zonen ominaisuuksia. Tätä kutsutaan myös SOA (Start of Authority) tallenteeksi. Tämä osuus on pakollinen kaikissa zone tiedostoissa
- Kaikki ”käyttäjät” zonessa. Näitä kuvataan usein Address Resource Recordein (A record) IPv4:ssä. IPv6 käyttää AAAA recordeja.
- Dataa joka selittää globaalia tietoa zonesta. Usein nämä ovat MX resource recordeja, jotka selittävät domainin mail palvelimia ja NS resource recordeja, jotka selittävät autoritaarisia nimipalvelimia domainissa.

(Aitchison, R. 2011, chapter 1)

Zone tiedostoformaatti

RFC 1035 mukaan zone-tiedostot ovat tekstitiedostoja, joita voidaan muokata millä vain standardin mukaisilla tiedostomuokkaimilla, ja nämä zone tiedostot voivat sisältää kolmen tyyppisiä merkintöjä.

- Kommentit (comments) alkavat puolipisteellä (;) ja jatkuvat rivin loppuun asti. Kyseinen merkki terminoi lopun rivin.
- Direktiivit (directives) alkavat dollarimerkillä ($) ja niitä käytetään zone tiedostojen prosessoimiseen.
- RR (resource record) käytetään määrittämään erilaisia määreitä ja arvoja domainissa. RR:t ovat yhden rivin mittaisia, ellei kyseistä riviä laiteta sulkujen sisään.

(Aitchison, R. 2011, chapter 2)

Zone tiedostosisältö

Zone tiedosto sisältää usein seuraavan tyyppisiä RR:iä ja direktiivejä:

$TTL- Määrittää (TTL) arvon zonelle tai domainille. Tämä arvo määrittää sen, kuinka pitkäksi aikaa yksi RR voidaan välimuistittaa toiselle DNS palvelimelle. Tämä osuus on pakollinen.

Jokainen Resource Record voi käyttää vapaaehtoista TTL-arvoa, joka määritetään sekunneissa. RFC2038 on määrittänyt standardin arvon TTL arvoksi RR:lle, jos sitä ei erikseen määritetä. TTL:n syntaksi standardissa on muotoa ”$TTL aika-sekunteina”. Nollaksi määritetty aika tarkoittaa, että kyseistä tietoa ei välimuistiteta. Maksimaalinen arvo on 2147483647, joka tarkoittaa yli 68 vuotta. Standardi RFC1912 tosin suosittelee, että kaikille tietueille annettaisiin vähintään päivän mittainen TTL arvo ja RR:lle jotka harvoin muuttuvat annettaisiin jopa viikkoja vastaava arvo. TTL määrittää myös kaksi DNS operaatiota. Ensimmäinen on ACCESS LOAD, joka määrittää kuinka nopeasti tallenne häviää resolverin välimuistista; kuinka usein DNS kyselyitä täytyy tehdä. Toinen on CHANGE PROPAGATION, joka määrittää kuinka kauan täytyy maksimissaan odotella, jotta jokin muutos leviää zonen palvelimelta kaikille käyttäjille. (Aitchison, R. 2011, chapter 2)

$ORIGIN- Domain nimi zonelle jota määritetään. Tämä on vapaaehtoinen. Kyseinen määritys on standardoitu standardissa RFC 1035. Kyseinen määritelmä kertoo mikä

domain nimi on lisättävä RR:ään, jonka nimi on puutteellinen. Kyseinen prosessi on käyttäjälle täysin näkymätön. Tässä vaiheessa on hyvä huomioida, että jos domain ei pääty pisteeseen, ORIGIN määritelmän mukainen domain nimi lisätään perään. Täydellinen domain nimi päättyy pisteeseen (FQDN). Syntaksi Origin attribuutille on ”$ORIGIN domain-nimi”. HUOM! Kyseisen muuttujan ”domain-nimi” täytyy päättyä pisteeseen, jotta kyseessä on FQDN. Vaikka kyseessä onkin vaihtoehtoinen attribuutti, on se hyvä määrittää sillä esimerkiksi BIND9 ohjelmisto tulee tämän arvon puuttuessa korvaamaan tämän itsenäisesti. (Aitchison, R. 2011, chapter 2)

SOA RR- Tämän täytyy esiintyä ensimmäisenä RR:nä zone tiedostossa. Tämä esittää globaaleja määreitä zonesta tai domainista. Zone tiedostossa voi olla vain yksi SOA RR ja tämä osuus on pakollinen. SOA on määritetty standardissa RFC 1035. Kyseinen RR on tärkein zone tiedostossa, joka sisältää monia parametreja. SOA:n syntaksi on seuraavanlainen: ”name ttl class rr name-server e-mail sn refresh retry expiry min”. (Aitchison, R. 2011, chapter 2)

NS RR- Määrittää autoritaariset nimipalvelimet kyseiselle zonelle tai domainille. Näitä merkintöjä täytyy olla kaksi tai useampi. Tämä osuus on pakollinen. NS on määritetty standardissa rfc 1035. NS RR:n syntaksi on muotoa ”name ttl class rr name” (Aitchison, R. 2011, chapter 2)

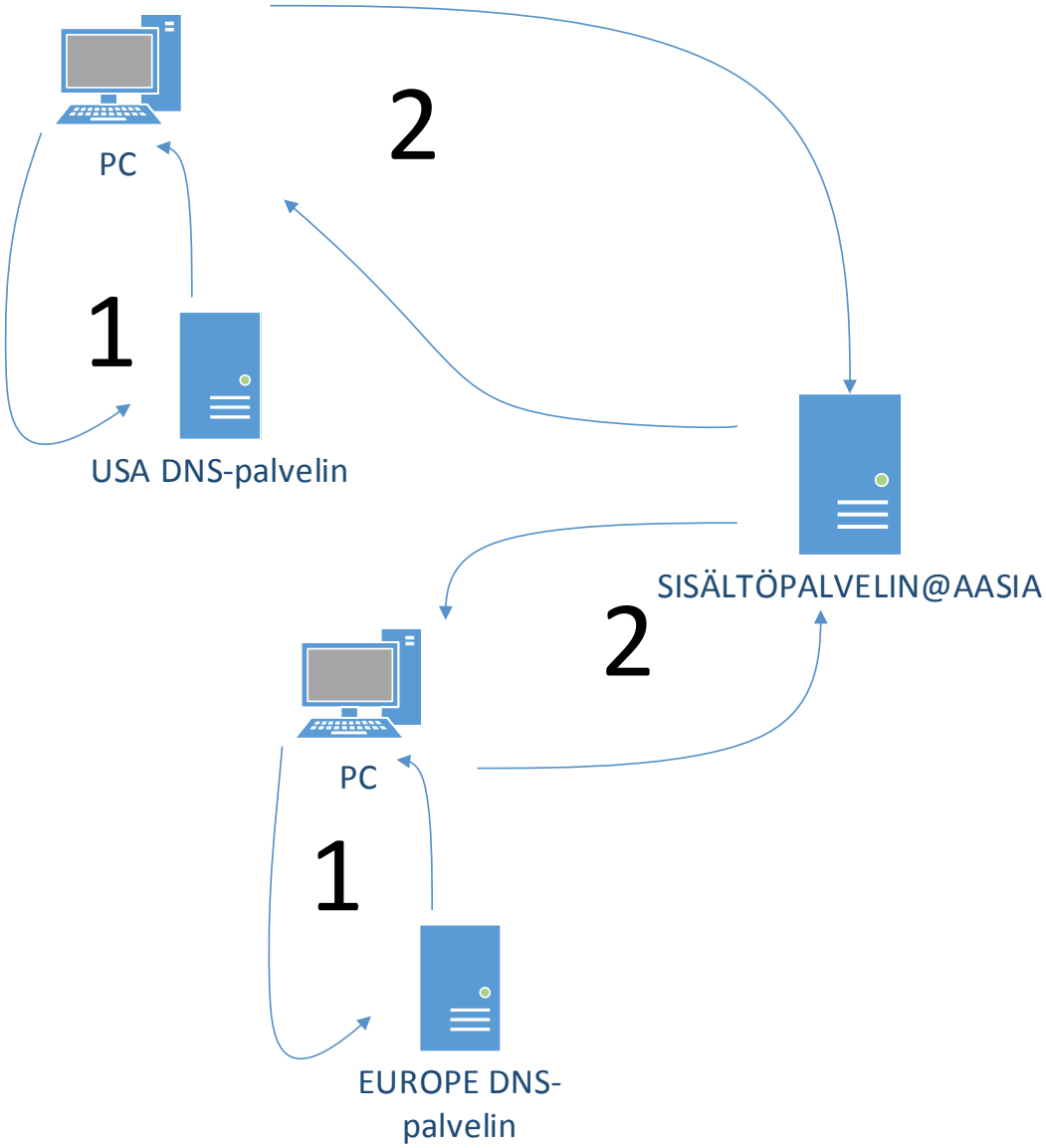
MX RR (mail exchanger)- Määrittää sähköpostipalvelimet zonelle. Tämä voi osoittaa myös viereisiin domaineihin. Tämä osuus on vapaaehtoinen. (Aitchison, R. 2011, chapter 2)

A RR (address)- Käytetään määrittämään IPv4 osoite palveluille kyseisessä domainissa. Tämä osuus on vapaaehtoinen. (Aitchison, R. 2011, chapter 2)

CNAME RR- Määrittää Alias RR:n, joka kertoo ”nimen” tietyllä palvelulle. Tämä osuus on vapaaehtoinen. (Aitchison, R. 2011, chapter 2)

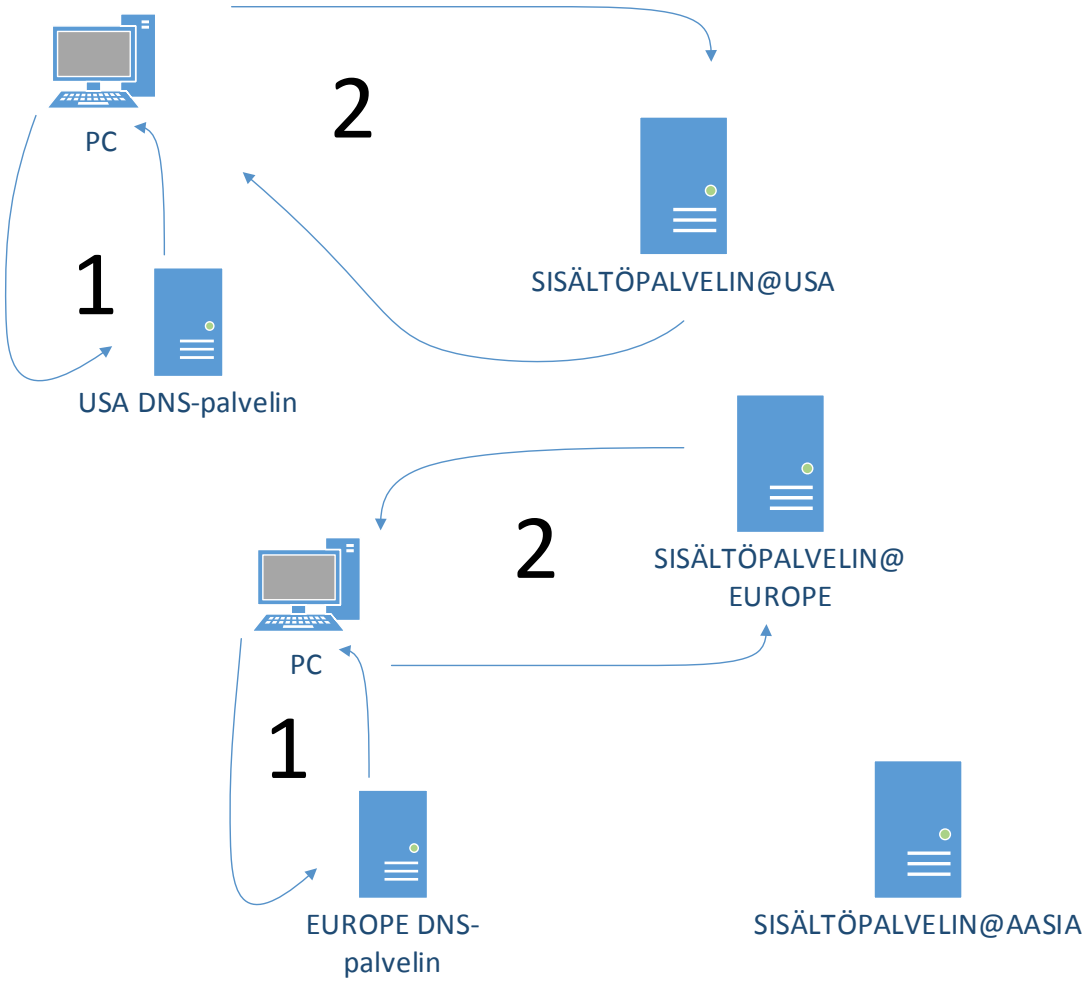
2.8 CDN ja perinteinen DNS-malli, käytännön ero

Perinteisessä DNS mallissa on yksi DNS-nimi yhtä IP-osoitetta vastaan, kuten aiemmissa teorioissa tämä onkin selitetty. CDN eroaa tästä siten, että yhtä DNS-nimeä kohtaan voidaankin osoittaa montaa eri IP-osoitetta riippuen halutuista parametreista, joita voivat olla mahdollisesti mm. ”hyppyjen määrä”, kuorman määrä tai käyttäjän geolokaatio. Kuviossa 8 on piirrettynä perinteinen DNS-malli. Kyseisessä mallissa jokainen käyttäjä saa 1. kohdan DNS-nimikyselyynsä saman vastauksen, oli käyttäjä mistä päin maailmaa tahansa. Kyseisen vastauksen jälkeen kaikki käyttäjät ottavat yhteyden samaan palvelimeen, kuten kohdassa 2 on nuolin osoitettu. Perinteinen malli on täysin toimiva, kun palvelun toiminta on kohdistettu tiettyyn käyttäjäyhteisöön, joka on maantieteellisesti hyvin keskittynyttä. Maantieteellisesti keskittynyt käyttäjäkunta voi olla esimerkiksi jokin suomalaisille kohdistettu palvelu, jota ei ole alun perin tarkoitettu edes ulkomaalaisten käytettäväksi. Tästä johtuen kaukaisemmilla käyttäjillä palvelussa voi olla helposti häiriötä ja saavutettavuusongelmia johtuen huonoista yhteyksistä.



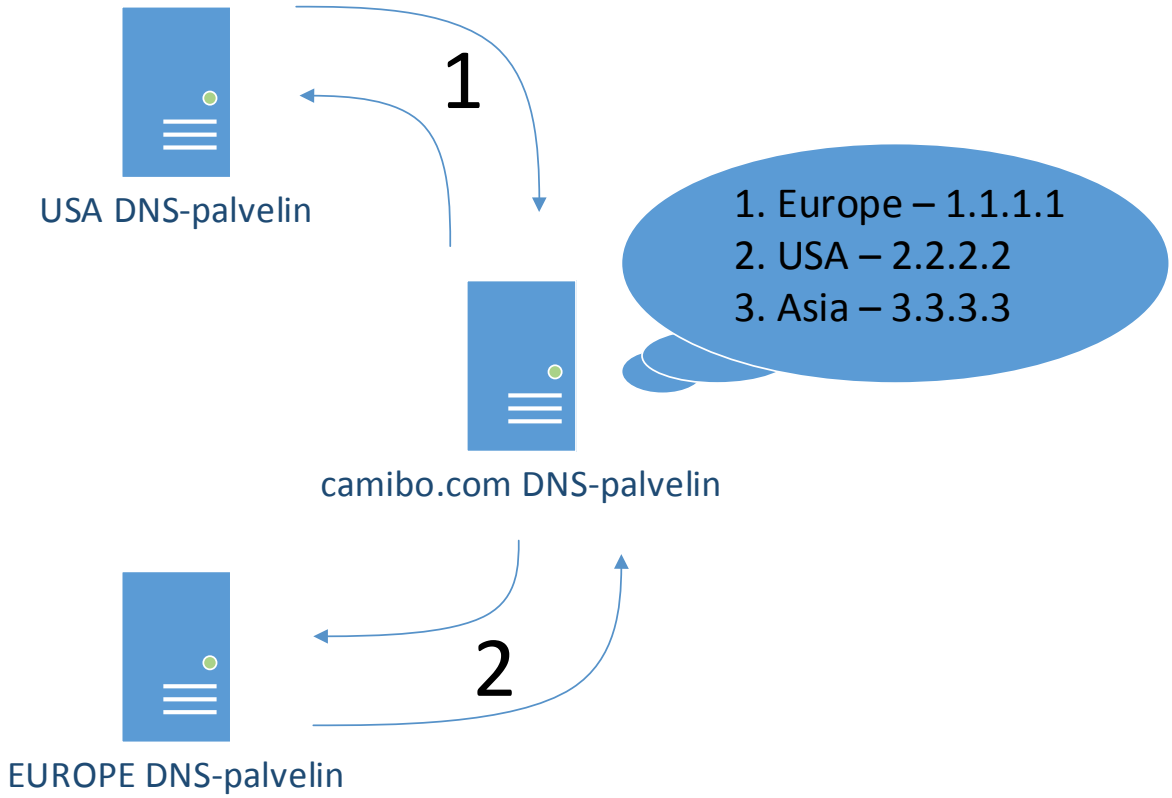
Kuvio 8 perinteinen DNS-malli

Perinteisen DNS-mallin ongelmia voi korjata suhteellisen helposti ottamalla käyttöön DNS-pohjaisen kuormantasauksen perustuen käyttäjän geolokaatioon. Kyseisessä mallissa loppukäyttäjä saa läheisimmän sisältöpalvelimen haluamalleen sisällölle perustuen hänen maantieteelliseen sijaintiinsa. Tämä prosessi on havainnollistettuna kuviossa 9. Nyt aiemmin käytetty Aasian sisältöpalvelin ei palvele enää USA:n ja EU:n käyttäjiä. CDN geolokaatioon perustuvassa kuormantasauksessa lähin sisältöpalvelin tuodaan loppukäyttäjille perustuen heidän käyttämänsä DNS-palvelimen käyttämään IP-osoitteeseen.



Kuvio 9 CDN DNS-malli

Kuviossa 9 esitetty malli toimii käyttäjälle täysin näkymättömästi. Kuviossa 10 on puolestaan havainnollistettuna, miten tämä prosessi toimii taustalla. Kun USA:ssa sijaitsevat DNS-palvelimet pyytävät osoitetietoa camibo.com resursseille, palautetaan heille arvo perustuen USA:n IP-osoitealueeseen, eli tässä tapauksessa camibo.com autoritäärinen palvelin palauttaa IP-osoitteen 2.2.2.2. Prosessi toimii samalla tavalla kaikille DNS-kyselyille, eli kohdassa 2. Euroopan DNS-palvelimelle palautetaan arvo 1.1.1.1.



Kuvio 10 CDN taustatoiminta

2.9 Esimerkki nykypäivän CDN-verkosta

EdgeCast Networks on Yhdysvaltalainen CDN-palveluntarjoaja, joka on perustettu vuonna 2006. Yrityksellä on henkilökuntaa noin 300 henkilöä ja sen päätoimipiste sijaitsee Kaliforniassa. (Edgecast. 2014, The Company)

Aiemmissa CDN-ratkaisuissa asennettiin pieniä määriä palvelimia ympäri maapalloa, jotta palvelu saataisiin mahdollisimman lähelle käyttäjää. Edgecast on tosin katsonut tämän mallin vanhanaikaiseksi ja tehottomaksi ja sijoittanut pienemmän määrän suuria datakeskuksia Internet Exchange Pointien (IXP) yhteyteen. Näitä pisteitä Edgecast Networks kutsuu SuperPOP nimityksellä. SuperPOP on käytännössä palvelinsali, josta löytyy massiivinen määrä tiedon prosessointinopeutta ja välimuistitusmahdollisuuksia. Nämä SuperPOP-pisteet ovat kaikenlisäksi kiinni suurimpien runkoverkkojen yhteyksissä mahdollistaen nopean tiedonsiirron. Kuviossa 11 on Edgecast Networksin ”SuperPOP”-palvelinkeskusten sijainnit kartalla. (Edgecast. 2014, network overview)

Kuvio 11 Edgecast Networks SuperPOP lokaatiot (Edgecastmap)

Edgecast Networks on valinnut käyttäjälle parhaan datakeskuksen määritykseen Anycast-ratkaisun. (Edgecast. 2014, Application Delivery)

3 Käytetyt järjestelmät

3.1 Bind9

Berkeley Internet Name Domain, tai lyhyesti BIND on avoimen lähdekoodin DNS-ohjelmisto, jota kehittää tällä hetkellä ISC inc. (Internet Systems Consortium). BIND on ehkäpä käytetyin DNS-sovellus, joskaan ei ainoa. Kun puhutaan IETF:n (Internet Engineering Task Force) RFC-dokumenteista, BIND:iä on aina pidetty ohjelmistona, joka toteuttaa tämän dokumentin mukaisia säädöksiä ja ominaisuuksia hyvin tarkasti. Useimmissa tapauksissa tarkka dokumentoinnin mukainen toteutus tarkoittaa myös ohjelmiston hitautta suuremmalla määrällä ominaisuuksia, BIND ei ole tässä tapauksessa poikkeus. (Aitchison, R. 2011, chapter 1)

BIND on tällä hetkellä kaikista suosituin ja käytetyin DNS-ohjelmisto Internetmaailmassa. Järjestelmää pidetään yleisesti ottaen vakaana ja on siksi valittu useimpien organisaatioiden DNS-järjestelmäksi. (Bind, 2014)

BIND-ohjelmisto koostuu kolmesta osasta:

1. DNS-palvelinjärjestelmästä. Järjestelmä käyttää ”named”-ohjelmaa vastaanottamaan ja vastaamaan kaikkiin DNS-nimenselvityspyyntöihin.
2. Selvittäjästä. Selvittäjän tehtävä on selvittää DNS-nimiä vastaavat IP-osoitteet. Tämän selvittäjä toteuttaa lähettämällä kyselyt oikeille palvelimille internetissä. Tähän pakettiin kuuluu myös ns. kirjastot, joita ohjelmoijat voivat lisätä omiin ohjelmiinsa, jos tuki DNS-nimenselvityksille tarvitaan.
3. Testausohjelmistoista. Nämä ohjelmistot ovat diagnostiikkaohjelmia, joilla vikoja saadaan paikallistettua. (Bind, 2014)

3.2 Windows Server IIS7.5 ARR

Internet Information Services (IIS) on Microsoftin Web-palvelintuote. IIS7 ja uudemmat versiot kyseisestä ohjelmistotuotoksesta tarjoavat uuden pyyntöprosessointi arkkitehtuurin, joka sisältää mm:

1. Windows Process Activation Servicen (WAS), joka sallii muidenkin kuin HTTP ja HTTPS-protokollien käytön.
2. Web-palvelinmoottorin, jota voidaan kustomoida lisäämällä erilaisia moduuleja.
3. Integroitujen pyyntöprosessien pipelinet IIS:stä ja ASP.NET:stä. (Introduction to IIS Architectures, 2007)

IIS7:n arkkitehtuuri on muuttunut versiosta IIS6 merkittävästi, sillä kyseessä on ohjelmiston uudelleenkirjoitus. Tästä huolimatta moni vanha arkkitehtuurillinen asia on pysynyt samana. Näistä tärkeimmät ovat mm. työskentelijäprosessit, application poolit ja ISAPI. Yksi suurimmista muutoksista on web-palvelimen integroituminen applikaatiopalvelimen kanssa. Kun IIS6-järjestelmä puolestaan tuki monia eri palveluita, kuten ASP.NET:iä tai SharePointia, on IIS7 integroitu tiiviimmin näiden palveluiden yhteyteen. (Schaefer K. 2008, chapter 2)

Application Request Routing (ARR) on lisäosa IIS-palvelinjärjestelmään. ARR-lisäosa tuo mahdollisuuden ylläpitäjille lisätä ominaisuuksia verkkoonsa. Lisäominaisuuksista hyötyvät eritoten Content Delivery Networkien (CDN) ylläpitäjät. ARR tuo IIS:ään uusina ominaisuuksina mm. sääntöpohjaisen reitityksen, keksit, kuormantasauksen HTTP-pyynnöille ja hajautetun välimuistituksen. (IIS.Net. 2014)

3.3 F5 BIG-IP GTM

F5 BIG-IP Global Traffic Manager (GTM) on järjestelmä, joka tarkkailee verkon saatavuutta ja suorituskykyä perustuen muista järjestelmistä saatuun dataan, ja skaalaa verkon toimintaa sen mukaan. BIG-IP GTM-järjestelmällä voidaan suorittaa kuormantasausta, topologiaan perustuvaa reititystä ja sääntökontrollia käyttäen omaa iRules skriptausta. (BIG-IP_GTMC, s.14)

BIG-IP GTM:llä voidaan suorittaa tasoittaista globaalia kuormantasausta (GSLB). Laitteisto jakelee DNS nimenselvitys pyyntöjä hierarkisesti. Ensimmäisenä pyyntö selvitetään parhaiten sopivaan ”altaaseen” Wide IP-ympäristössä, minkä jälkeen altaan sisältä valitaan paras virtuaaliserveri, jonka IP toimitetaan siten alun perin sitä pyytävällä järjestelmälle. Parhaan vaihtoehdon GTM valitsee perustuen käyttäjän asettamiin parametreihin. Kuormantasaus voi olla joko staattista tai dynaamista. Staattisessa metodissa kuormantasaus hoidetaan käyttäen valmiiksi aseteltuja kuvioita. Dynaamisessa kuormantasauksessa haluttu resurssi valitaankin käyttäen suorituskykyä mittaavia parametreja, joita big3d agentit keräävät ympäri verkkoa.
(BIG-IP_GTMC, s.14)

BIG-IP GTM suorittaa DNS pyyntöjen kuormantasausta perustuen saatavuuteen. Pyydetty resurssi voidaan luokitella saatavaksi, kun se vastaa yhteen tai useampaan vaatimukseen, jotka sille on asetettu. BIG-IP GTM:ssä saatavuus voidaan hoitaa kolmella eri tavalla: resurssin riippuvuus toisesta resurssista, saatavuus rajat tai arvot, jotka kyseinen resurssi palauttaa BIG-IP GTM monitorille. Kun yksi resurssi määritellään poissaolevaksi tai saavuttamattomaksi, BIG-IP GTM valitsee seuraavaksi parhaan resurssin perustuen olemassa olevaan kuormantasausmetodiin. (BIG-IP_GTMC, s.17)

BIG-IP GTM:n toiminta perustuu pääsääntöisesti DNS palvelujen ympärille. F5 on rakentanut DNS palveluja varten oman DNS palvelimensa nimeltä DNS Express. DNS Express on kevyt DNS palvelin, joka voi vastata DNS pyyntöihin, sekä tehdä ”zone transfer requesteja” tietyille asiakkaille. Tämän lisäksi DNS Express tarjoaa myös mahdollisuuden kommunikaation suojaukseen käyttäen TSIG avaimia. (F5 DNS-implementation, s.22)

4 Toteutus

4.1 GeoDNS kuormantasauspalvelin

GeoDNS kuormantasaus-palvelin tulee toimimaan yhdyspisteenä, jonka kautta liikenne välitetään ympäri maailmaa perustuen loppukäyttäjän geolokaatioon perustuen joko käyttäjän käyttämään DNS-nimipalvelimeen tai käyttäjän IP-osoitteeseen. Ainoastaan DNS-kyselyt tulevat käymään GeoDNS kuormantasaajalla, jonka jälkeen liikenne siirtyy muiden palvelimien hoidettavaksi. Työssä käytetty CDN toimintamalli on aiemmin teoriassa selitetty Peering/Private CND-malli, joita monet suuret CDN-palveluntarjoajatkin käyttävät.

Mahdollisia kuormantasausratkaisuja oli useita, joista yksi vaihtoehto oli myös KEMP-pohjainen ratkaisu, mutta lisenssikysymysten vuoksi päädyin tekemään toteutuksen käyttäen F5 Global Traffic Manageria, että Bind9-ohjelmistoa. Bind9 on täysin puhdas DNS-palvelin, johon joudutaan asentamaan kolmannen osapuolen päivitys, ennen kuin DNS kuormantasaus saadaan toimintaan. F5 GTM on puolestaan täysiverinen kuormantasaaja, joka käyttää omia tehokkaampia algoritmejä DNS-tietueiden levitykseen, että kuormantasaukseen.

Kuormantasausta tutkittiin myös Windows Server järjestelmillä, joiden päällä toimi Application Request Routing. Microsoft ARR pystyy tosin vain paikalliseen kuormantasaukseen, eikä se sovellu DNS-pohjaiseen globaaliin kuormantasaukseen. Windows ARR-ratkaisuja tulin tosin käyttämään itse palvelinkeskusten luonnissa.

4.2 Bind9

Bind9 GeoDNS palvelin asennettiin CentOS 6.4 järjestelmän päälle. Bind9:stä käytettiin versiota 9.4.1-P1, sillä kyseinen GeoDNS päivitys on luotu kyseiselle Bind-versiolle. Kyseisen GeoDNS-päivityksen on tehnyt caraytech, joka käyttää Maxmind geotietokantoja hyväkseen ja kyseisen päivityksen avulla voidaankin jakaa kokonaisia valtioita alueiksi ilman, että täytyisi tietää kyseisten alueiden IP-osoitteita.

Ensimmäisenä luotuna alueena on Amerikka, pohjoinen ja eteläinen. Maat saadaan jaoteltua yksinkertaisesti luomalla oma view. Kaikki valtiot tietokannassa käyttävät ISO-3166-2 mukaisia maakoodeja.

Alla oleva konfiguraatio toimi pohjana kaikille view-lausekkeille. Kyseiset view-lausekkeet luotiin aivan normaalisti named.conf konfiguraatiotiedostoon. Koska kyseinen palvelin on Autoritäärinen DNS-palvelin, recursion asetus on asetettu ”no” arvolle.

```
view ”name” {
        match-clients { country_US; }
        recursion no;
        zone ”camibo.com” {
                type master;
                file “/var/named/camibo.com.db”;
};
```

Kuviossa 12 on “America” palvelimelle tarkoitetut konfiguraatiot. America-alueeseen on tarkoitettu pohjoinen, että eteläinen Amerikka.

```
view "America" {
        match-clients { country_AR; country_BR; country_CA; country_CU; country$
        recursion no;
        zone "camibo.com" {
                type master;
                file "/var/named/camibo-us.com.db";
        };
```

Kuvio 12 Bind9 america view

Kuviossa 13 on samallalailla luotu Aasian view-konfiguraatio.

```
view "asia" {
        match-clients { country_IN; country_CN; country_KR; country_HK; country$
        recursion no;
        zone "camibo.com" {
                type master;
                file "/var/named/camibo-asia.com.db";
        };
};
```

Kuvio 13 Bind9 asia view

Kun aiemmassa kohdassa luotiin koko Aasiaa koskeva view, Euroopan kohdalla pääseekin jo helpommalla. Eurooppa on työssä oletus view ja siksi Eurooppaan ei tarvitse ”match-clients” lehdelle yhtään valtiokoodia. Ainoat alueet, joille tarvitsee luoda maakohtaisilla koodaukset ovat Amerikka ja Aasia.

```
view "other" {
        match-clients { any; };
        recursion no;
        zone "camibo.com" {
                type master;
                file "/var/named/camibo.com.db";_
        };
```

Kuvio 14 Bind9 other zone

Kuviossa 15 on yksinkertainen konfiguraatio camibo:n Amerikan zone-tiedostolle. Eroavaisuus alueellisten konfiguraatioiden välillä tulee lähinnä IP-osoitteistuksessa käyttäjien ”x.camibo.com” suhteen, kuten tässä kohdassa www.

```
;
$TTL    10
camibo.com.     86400   IN      SOA     ns1.camibo.com. mail.camibo.com.       $
                5               ;serial number
                86400           ;refresh
                10              ;retry
                3600000         ;expire
                10              )
ns1.camibo.com.         IN      A       81.52.190.150
camibo.com.             IN      NS      ns1.camibo.com.
www.camibo.com.         IN      A       67.201.66.153
```

Kuvio 15 Bind9 Amerikan zone

Kuviossa 16 on puolestaan konfiguraatio camibo:n ”default zonelle”, eli Euroopan zonelle. Suurin eroavaisuus on siis www-client IP-osoitteistuksessa.

```
;
$TTL    10
camibo.com.     86400   IN      SOA     ns1.camibo.com. mail.camibo.com.       $
                4               ;serial number
                10              ;refresh
                7200            ;retry
                3600000         ;expire
                10              )
www.camibo.com. 300     IN      A       81.52.190.117
ns1.camibo.com.         IN      A       81.52.190.150
camibo.com.             IN      NS      ns1.camibo.com.
```

Kuvio 16 Bind9 Euroopan zone

Viimeisenä kuviossa 17 on Aasian käyttäjille määritetty zone-tiedosto. Eroavaisuus on edelleenkin vain www-asiakkaan IP-osoitetiedoissa.

```
;
$TTL    10
camibo.com.     86400   IN      SOA     ns1.camibo.com. mail.camibo.com.        $
                4               ;serial number
                10              ;refresh
                7200            ;retry
                3600000         ;expire
                10              )
www.camibo.com. 300     IN      A       192.88.171.186
ns1.camibo.com.         IN      A       81.52.190.150
camibo.com.             IN      NS      ns1.camibo.com.
```

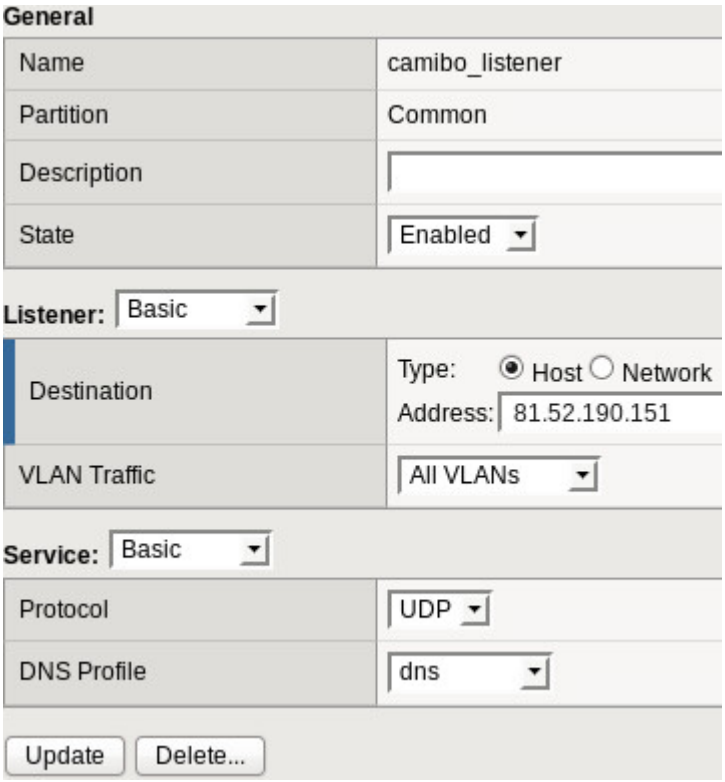
Kuvio 17 Bind9 Aasian zone

4.3 F5 Global Traffic Manager

Pohjimmiltaan F5 GTM toimii samalla tavalla kuin Bind9-järjestelmä. Kaikki toiminnallisuus perustuu DNS:ään, jonka perusteella käyttäjät sijoitetaan eri puolilla maailmaa sijaitseville palvelimille. Eroavaisuus tulee konfiguraatiossa, joka on täysin erilainen valmistajasta toiseen, vaikka F5 GTM taustalla toimiikin Bind-järjestelmä.

Kuviossa 18 ensimmäisenä luotiin DNS pyyntöjen kuuntelijan F5 BIG-IP GTM:lle, sillä laite tarvitsee kuuntelijan ottaakseen vastaan ja vastatakseen tuleviin DNS- pyyntöihin. Toimenpide voidaan suorittaa välilehdillä:

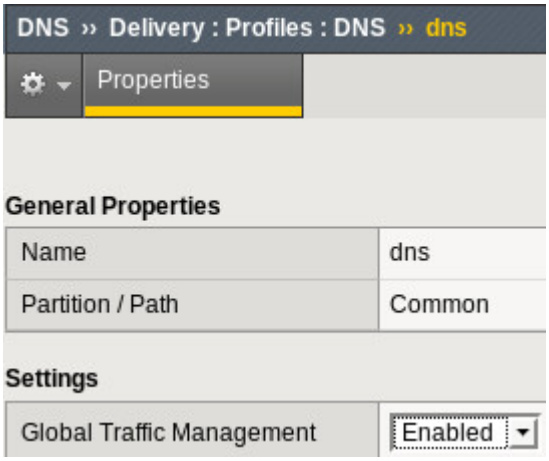
DNS-> Delivery-> Listeners



Kuvio 18 Camibo Listener

Uuden kuuntelijan voi luoda tabista ”Create…”. Tärkeimmät parametrit tällä sivulla olivat seuraavat: Name, Destination, service port. Vaikka vetovalikkojen alta olisi saanut vaihdettua ”Basic” lehden ”Advanced” lehdeksi, edistyneitä asetuksia kuuntelijalle ei tarvita. ”DNS Profile” säilytetään oletusarvoissaan tässä, että myös kuvion 19 tapauksessa, joka löytyy välilehdiltä:

DNS-> Delivery-> Profiles-> DNS



Kuvio 19 DNS profiili

On kuitenkin tärkeää huomioida, että “Global Traffic Management” on “Enabled”, sillä tämä on hyvin oleellinen GSLB:n toiminnan kannalta. Tällä saadaan globaalin tason kuormantasaus toimimaan DNS:n päällä.

Toimenpiteiden jälkeen F5 laitteen täytyisi pystyä kuuntelemaan tulevia DNS kyselyitä. Seuraavaksi asetetaan toimintaan GSLB:n alueprofiilit, jotka löytyvät

DNS-> GSLB-> Topology-> Regions

Kyseisellä toimenpiteellä luotiin alueet, johon käyttäjät luokitellaan heille asetetun DNS-palvelimen mukaan.
Topologiapohjainen kuormantasaus jakaa DNS nimenselvityspyynnöt perustuen käyttäjän geolokaatioon. Sijainnistaan riippuen käyttäjät siirretään mahdollisimman läheiselle palvelinkeskukselle. Kun topologiapohjainen kuormantasaus asetetaan päälle, BIG-IP GTM-järjestelmä käyttää topologia asetuksia kuormantasaukseen (BIG-IP_GTMLB, s.12)

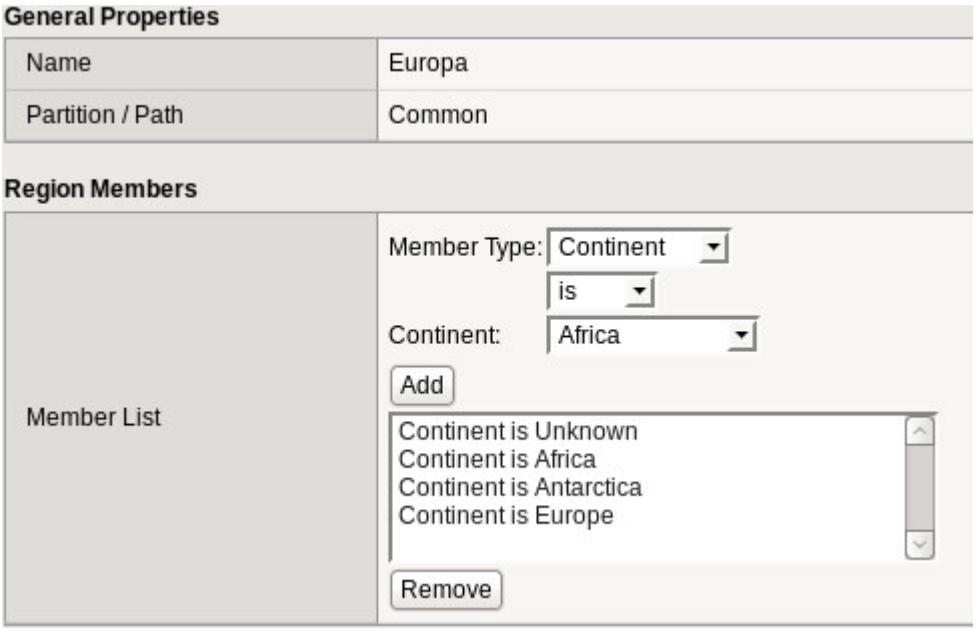
BIG-IP Global Traffic Manager järjestelmälle kuormantasauksen alueelliset merkinnät luodaan välilehdiltä:

DNS-> GLSB-> Topology-> Regions

Alueet jaettiin seuraavanlaisesti:

- ***Europa: ”Continents is: Unknown, Africa, Antarctica, Europa”***
- ***Asia: “Continent is: Asia, Oceania”***
- ***America: “Continent is: NA, SA”***

Kuviossa 20 on visuaalisesti vielä sama, joka on jo selitettynä aiemmin.

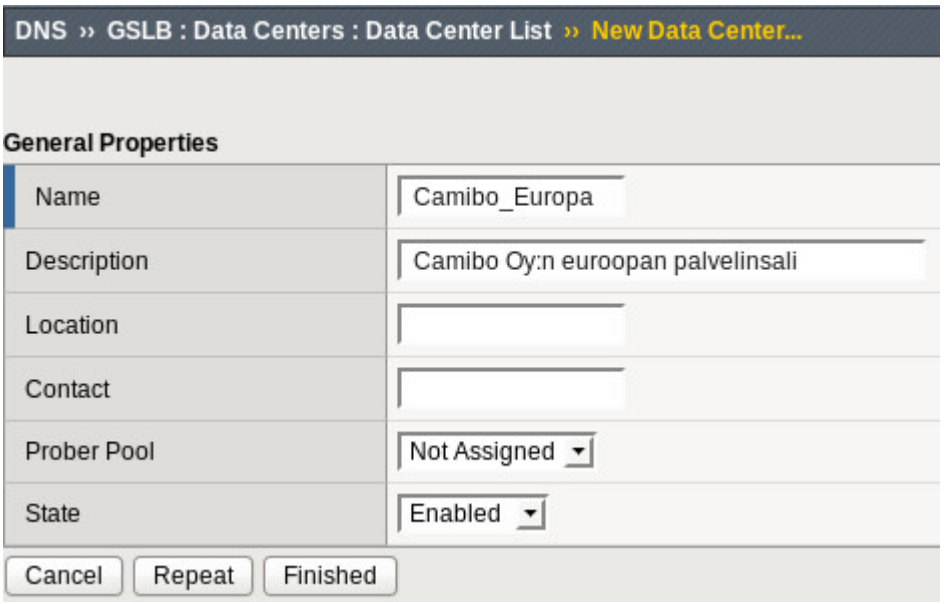


Kuvio 20 Regions välilehti

Kyseisten geolokaatiotietojen jälkeen on hyvä luoda Datakeskukset, virtuaalipalvelimet ja ”poolit”, joihin liikenne kuormatasataan käyttäen aiemmin määriteltyjä alueita.

Ensimmäisenä täytyy luoda palvelinkeskukset (Data Center), johon virtuaalipalvelimet luodaan. Tämä toimenpide on esitettynä kuviossa 21 ja voidaan konfiguroida seuraavien välilehtien alta:

DNS-> GSLB-> Data Centers

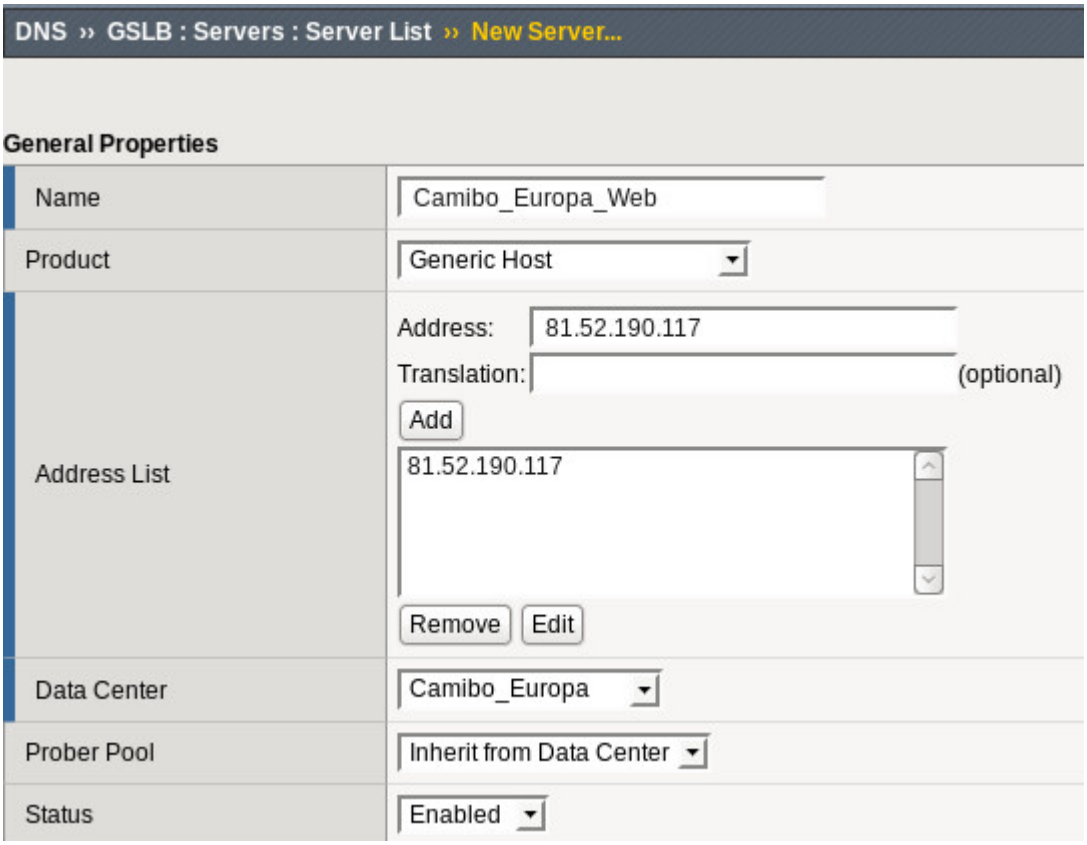


Kuvio 21 F5 Datakeskuksen asennus

Kyseinen toimenpide on suhteellisen yksinkertainen, sillä palvelinkeskuksille ainoa pakollinen tietue on nimi (Name). Tähän on tosin hyvä antaa lisätietojakin, sillä monimutkaisissa ratkaisuissa pelkkä nimitieto voi jättää paljon arvelun varaan. Kuvailu (Description) ei vaikuta järjestelmän toimintaan millään tapaa, vaan on oiva lisäapu antamaan lisätietoja kyseisestä palvelinkeskusmerkinnästä.

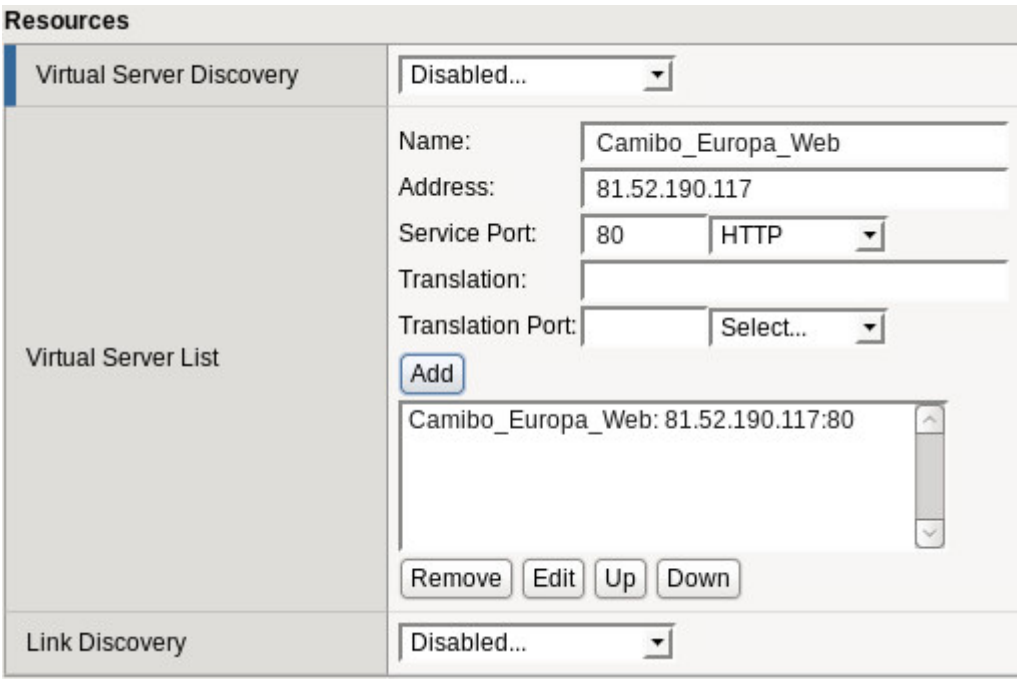
Kun halutut datakeskukset on luotuna, voidaan luoda itse virtuaalipalvelimet. Virtuaalipalvelimet eivät tässä kontekstissa ole varsinaisia palvelimia, vaan enemmänkin tietueita kyseisten palvelimien olemassaolosta. Kuviossa 22 on visualisoituna kyseinen prosessi ja kyseiset virtuaalipalvelimet voidaan luoda välilehdiltä:

DNS-> GSLB-> Servers



Kuvio 22 F5 Palvelin datakeskukseen

Ensimmäinen pakollinen tietue on kuten aiemminkin; nimitieto, joka on taas käyttäjän vapaasti valittavissa. Product-kentän merkintä ei ole oleellinen, mutta se on asetettu arvoksi ”Generic Host” omia merkintöjä varten. Kuviossa 23 on puolestaan saman välilehden toinen tärkeä osuus visualisoituna.

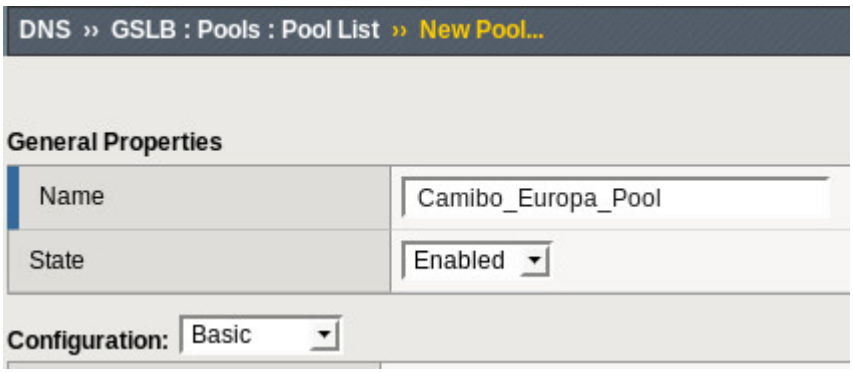


Kuvio 23 F5 Palvelin datakeskukseen 2

Address listaan on puolestaan asetettava osoitteet, joita kyseinen palvelin käyttää, sekä niiden mahdolliset NAT-tietueet. Data Center vetovalikosta valitaan kyseisiä palvelimia koskeva palvelinsali. Resources osion ”Virtual Server List” listaukseen on hyvä luoda kaikki palvelut, joita kyseinen palvelin tarjoaa loppukäyttäjille. Tässä vaiheessa on hyvä huomioida, että jos käytössä on useampia F5-laitteita ja ”health checkit” halutaan käyttöön, on serveriksi asetettava F5 LTM-laite, joka tekee tarvittavat osoitemuunnokset ja vastaa GTM:n tekemiin ”health checkeihin”.

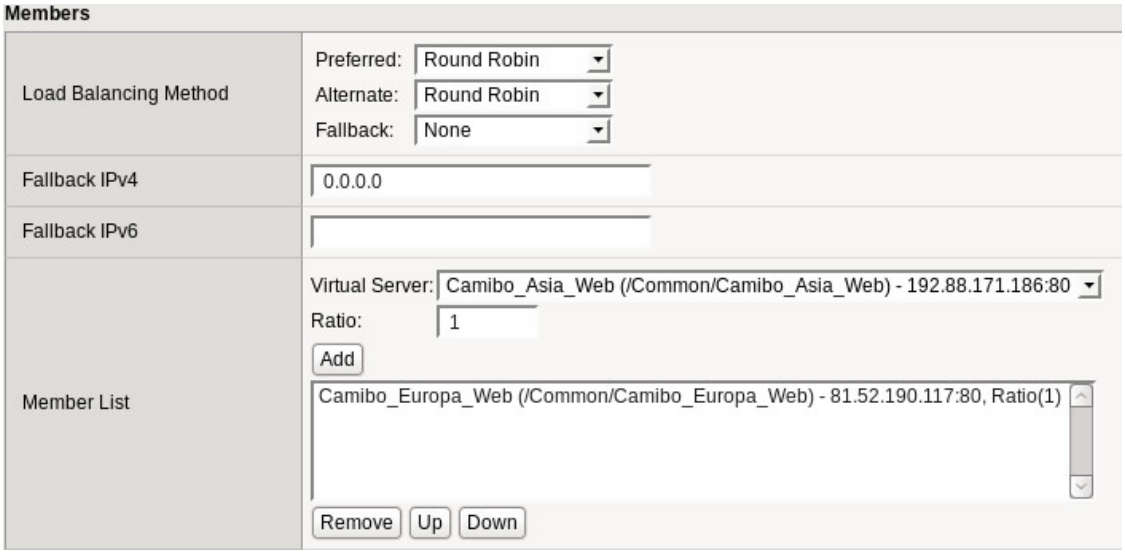
Kuviossa 24 on esitettynä Poolin luontiprosessi, joka voidaan suorittaa välilehdiltä:

DNS-> GSLB-> Pools



Kuvio 24 F5 Uusi pool

Pool on yrityksen tai organisaation toimipiste, jonka alla voi olla useita osia datakeskusta ja liikenne voidaan reitittää myös datakeskuksen sisällä. Poolin ”Camibo_Europa_Pool” konfiguraatio on esitettynä kuviossa 25.

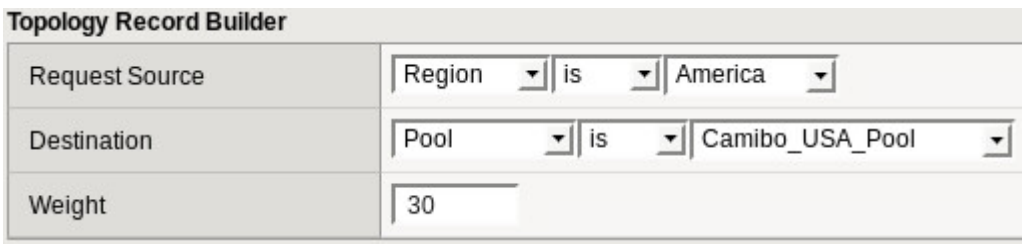


Kuvio 25 F5 Uusi pool 2

Koska referenssiratkaisussa keskityttiin itse liikenteen jakamiseen ympäri maapalloa, datakeskusten sisäinen reititys ei ollut oleellista työn kannalta. Tässä on tosin mahdollista reitittää liikenne käyttäen monia erilaisia kuormantasausmetodeja, joista päälle jätettiin ”Round Robin”, sillä palvelimia on tässä tapauksessa vain yksi poolia kohden. On tosin tärkeää huomioida ”Fallback Load Balancing Method”, sillä sen ollessa ”None” arvo, palvelinten ollessa alhaalla siirrytään takaisin ylemmälle tasolle kuormantasauksen hierarkiassa. Pool tarvitsee toimiakseen myös virtuaalikoneita, joista tässä tapauksessa on asetettu toimintaan ”Camibo_Europa_Web”.

Kuvion 20 geolokaatioiden jälkeen täytyy luoda myös painotukset eri palvelimille, kuinka liikenne jaetaan ympäri maapalloa käyttäen DNS:ää. Tämän on havainnollistettuna kuviossa 26 ja suoritetaan välilehdiltä:

DNS-> GLSB-> Topology-> Records



Kuvio 26 F5 Topologiset reititykset

Ensimmäiselle riville ”Request Source” laitetaan aiemmin luotu ”Region” alue ja ”Destination” lehdelle puolestaan Pool, eli haluttu ”datakeskus”. Kuviossa 27 on esitettynä luotuja säännöksiä geolokationaaliselle ”reititykselle”.

LDNS Request Source	Destination	Weight
Region is Asia	Pool is Camibo_Asia_Pool	30
Region is Europa	Pool is Camibo_Europa_Pool	20
Region is America	Pool is Camibo_USA_Pool	10

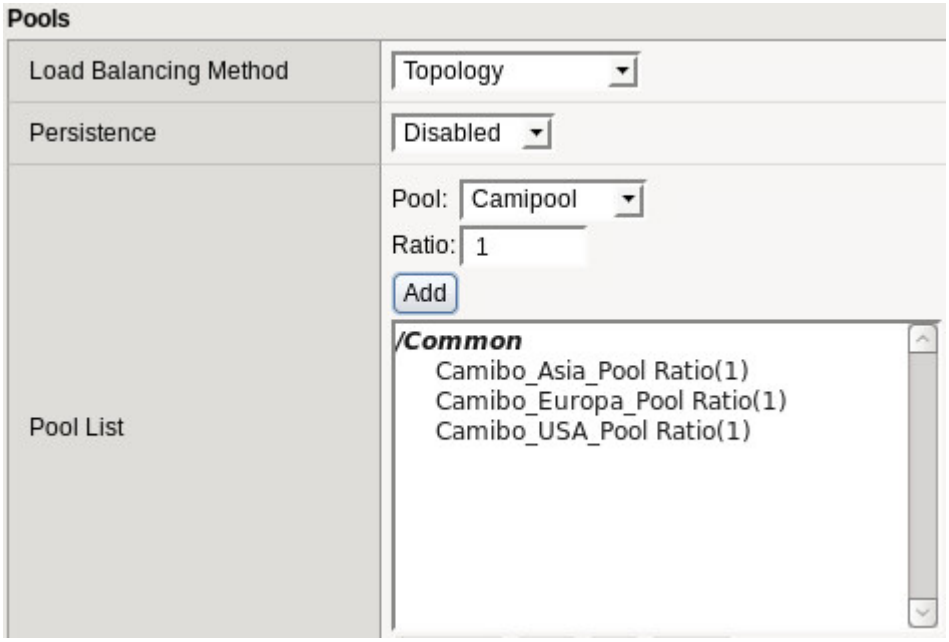
Kuvio 27 F5 Näkymä topologiareitityksistä

Seuraavaksi luotu Wide IP on tavallaan yrityksen koko verkko, johon liikenne reititetään käyttäen DNS:ää. Tässä tapauksessa liikenne sivustolle ”www.camibo.com” reititetään eri parametrein eri pooleille, jotka ovat tavallaan eri alueille ripoteltuja datakeskuksia. Kuviossa 28 esitetyn Wide IP:n voi luoda välilehdiltä:

DNS-> GSLB-> Wide IP

Kuvio 28 F5 Wide IP FQDN

Nimeksi täytyy antaa haluaman resurssin FQDN, joka on tässä tapauksessa ”www.camibo.com”. Kuviossa 29 on saman välilehden alempaa osiota, jotka on konfiguroitu ohessa.



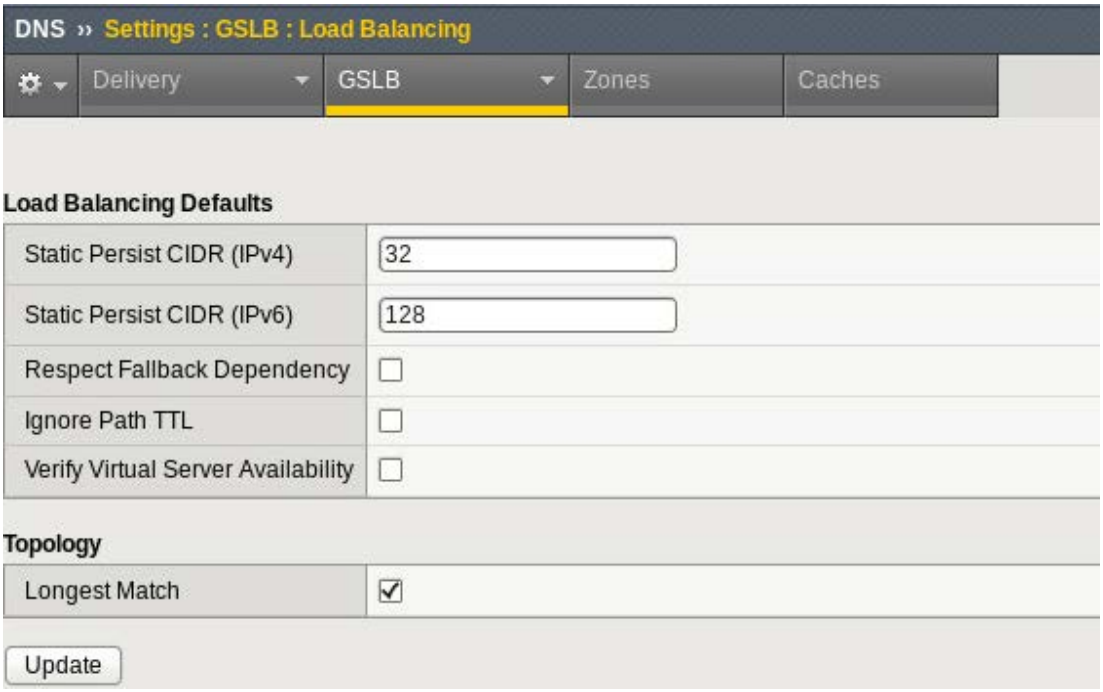
Kuvio 29 F5 Wide IP poolit

Pools asetuksiin täytyy asettaa ”Load Balancing Method”, joka täytyy asettaa arvoksi ”Topology”, jotta liikenne balansoidaan käyttäen aiemmin luotuja Topology recorde-ja. Pool-vetovalikosta on syytä lisätä kaikki datakeskukset, eli ”poolit”. Ratio arvon voi jättää oletukseksi, sillä kyseisellä arvolla on merkitystä ainoastaan, jos Load Balancing Method asetetaan arvolle ”ratio”.
Lopuksi hyväksymällä asetykset täytyisi näkyä kuvion 30 mukainen näkymä, jossa nä-kyy kaikki luodut tallenteet.

Kuvio 30 F5 Näkymä valmiista FQDN tallenteesta

Tässä vaiheessa toiminnan voi testata poistamalla tarkitukset käytöstä. Kuviossa 31 esitetyt tarkistukset voi poistaa välilehdeltä:

DNS-> Settings-> GSLB-> Load Balancing



Kuvio 31 F5 Verify Availability

Tarkistuksia käytetään varmistamaan palvelinten toiminta, sillä liikenne on turha reitittää palvelimille maailmalla, jos WEB-palvelimet ovat alhaalla. Tarkistukset saa pois käytöstä ottamalla täpän ”Verify Virtual Server Availability” pois päältä. Tämä asetus on oleellinen, jos verkossa ei ole muita F5-laitteita, sillä F5 käyttää big3d agentteja virtuaalipalvelinten ”health check” tarkastuksiin. F5 GTM ei itsessään pysty

näitä tekemään palvelimille, vaan F5 GTM tarvitsee F5 LTM järjestelmiä näiden tekemiseen ja raportoimiseen.

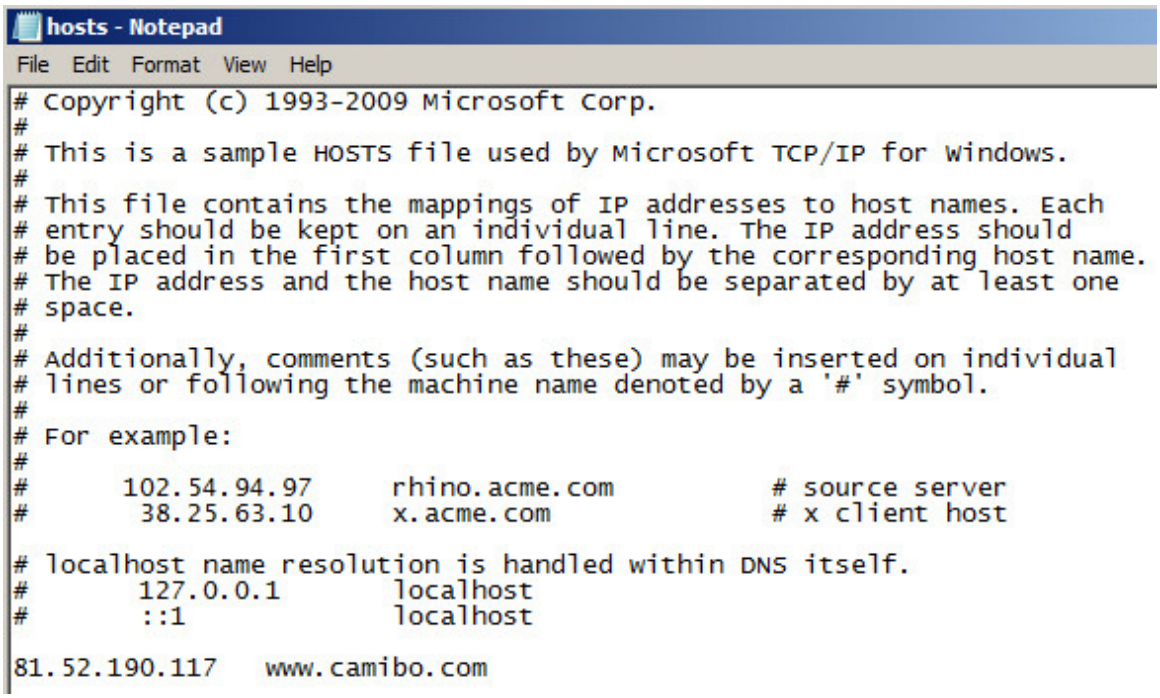
4.4 Microsoft Windows CDN palvelimet

4.4.1 Reunawebpalvelimet (edge)

Työn toimintaa testataan käyttämällä Content Destribution Network:n reunalle asetettuja web-palvelimia, jotka tarjoavat identtisen nettisivuston loppukäyttäjille. Reunawebpalvelimien käyttöjärjestelmänä toimii Windows Server 2008R2, jonka päälle on asennettuna rooliksi Web Server (IIS). IIS-roolin päälle on asennettu myös ”Application Request Routing version 3” (AAR3)-ohjelmisto, joka tuo monia hyödyllisiä ominaisuuksia IIS-ohjelmistoon.
Application Request Routing tarjoaa työn kannalta tärkeitä ominaisuuksia, kuten mahdollisuuden toimia kuormantasaajana muille sisäverkon palvelimille. Muita tärkeitä ominaisuuksia ovat mm. hajautettu välimuistitus, sekä keksien käyttö välityspalvelimien käytön suhteen. (IIS.net. 2014)

Ensimmäisenä tehtävänä on hyvä asettaa reunawebpalvelimet osoittamaan domain ”www.camibo.com” itseensä. Hosts tiedoston sisältö on esiteltynä kuviossa 32. Kyseinen toimenpide suoritetaan windowsin ”hosts”-tiedostoon, joka löytyy kansiorakenteesta:

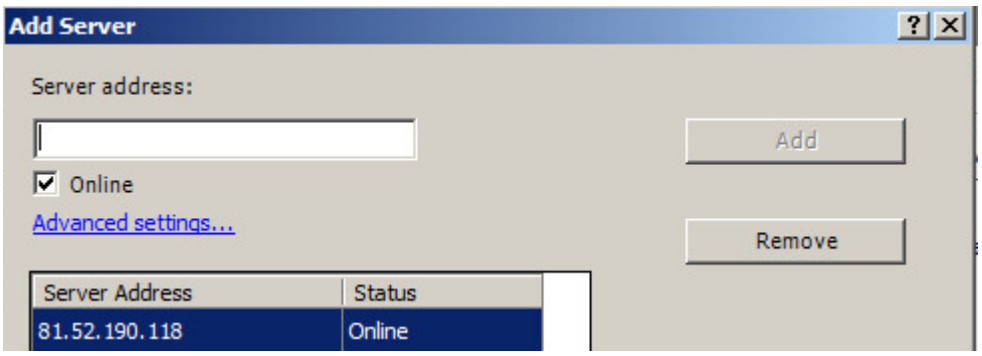
C:\Windows\System32\drivers\etc



Kuvio 32 Windows hosts-tiedosto

Toimenpide voidaan osoittaa joko omaan julkiseen tai sisäverkon IP-osoitteeseen, tai osoitteeseen 127.0.0.1 (localhost).

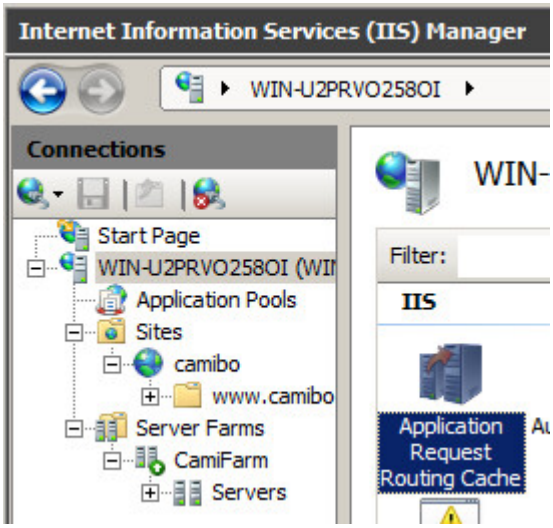
Seuraavaksi palvelimille asetetaan palvelinfarmi ”server farm” asetukset, jotta palvelimet pystyvät löytämään seuraavan tason palvelimet, joilta data mahdollisesti haetaan, jos sitä ei omasta välimuistista löydy. Kuviossa 33 on havainnollistettuna palvelimen asettaminen.



Kuvio 33 Windows edge palvelinfarmi

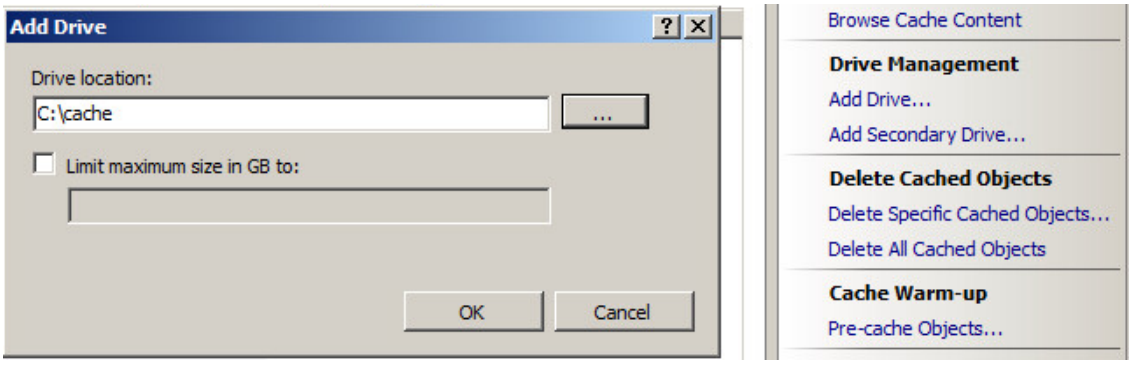
Palvelinfarmin voi luoda ”Server Farms” lehden alta, jota klikkaamalla ”Add Servers…” voidaan asettaa seuraavan tason palvelimet, joita reunapalvelin tulee käyttämään.

Reunawebpalvelimet tarvitsevat myös välimuistit käyttöönsä, sillä haluttu data täytyy ladata Ranskassa sijaitsevalta pääpalvelimelta, eli ns. origin palvelimelta. Tämä data on siksi hyvä tallentaa paikallisesti välimuistiin, jotta kyseistä dataa ei tarvitse aina ladata kaukaisesta pisteestä, vaan se voitaisiin tarjota suoraan reunapisteeltä.
Ensin on hyvä asettaa ensisijaisen (cache) välimuistin lähde. Kuvion 34 tilanteessa klikataan ensin ”Application Request Routing cache”.



Kuvio 34 ARR edge cache

Kun kyseinen kuvake on valittuna, ”Actions” lehdeltä valitaan ”Drive Management” ja tämän alta ”Add Drive…”. Kuviossa 35 on esitettynä välimuistin lisääminen.



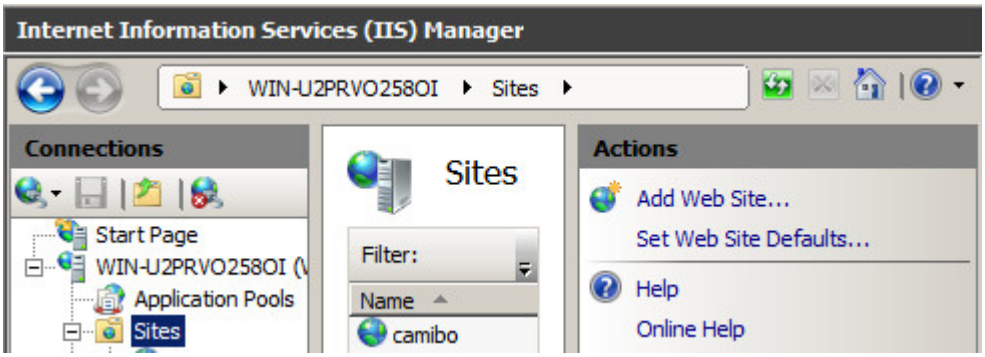
Kuvio 35 Välimuistin lisääminen

Tämä kyseinen välimuisti on hyvä asettaa kyseenomaiselle koneelle, sillä paikalliselta koneelta kyseinen välimuisti on nopein saavuttaa.

On myös hyvä huomioida, että välimuistiominaisuudet sallivat myös toissijaisen välimuistin käytön. Koneella voi olla aiemmin määritetty ensisijainen ”primary” cache ja

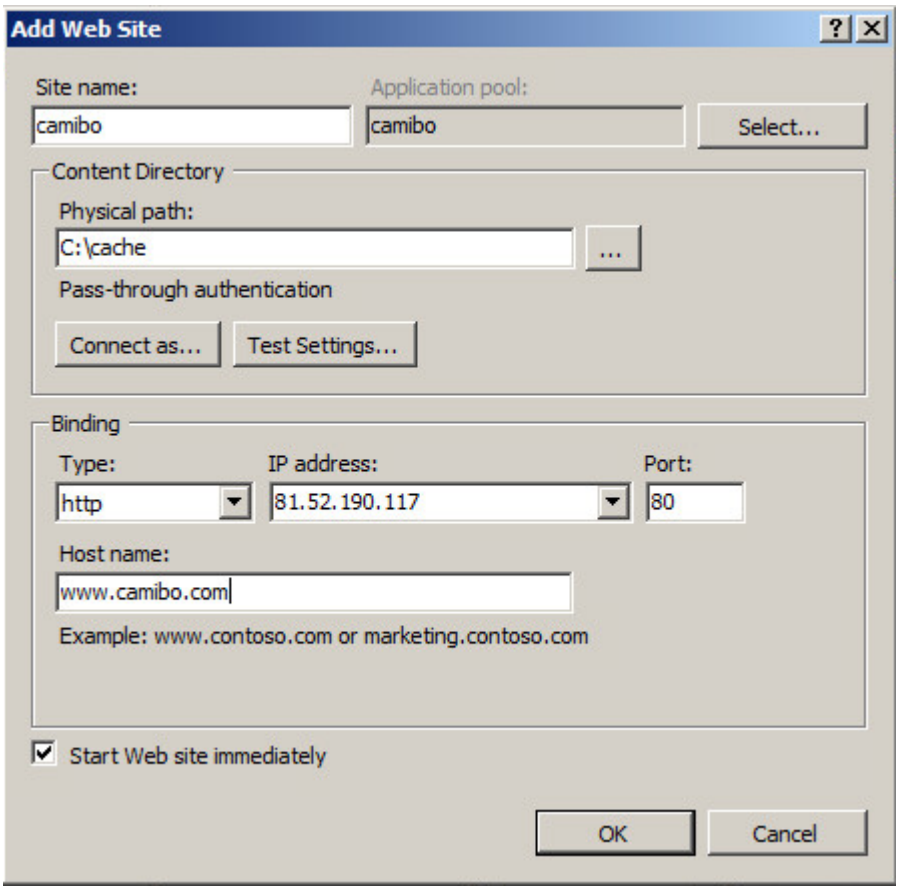
toissijainen ”secondary” cache. Toissijainen välimuisti on usein erillinen SAN-asema, jonka useampi edge-piste jakaa keskenään.

Itse web-sivusto asetettiin toimintaan kuvioiden 36 ja 37 mukaisesti.



Kuvio 36 Web-sivuston lisääminen

WEB-sivusto saadaan lisättyä normaaliin tapaan IIS-palvelimella. Eli ensimmäisenä valitaan ”Sites” välilehti, jonka alta valitaan ”Add Web Site...”. Tämän alta paljastuu kuvion 37 mukainen näkymä.



Kuvio 37 Web-sivuston asetukset

Kyseessä on Ranskan toimipisteen WEB-konfiguraatio, joskin muillakin sivupistellä parametrit IP-osoitetta lukuunottamatta ovat täysin samat.

4.4.2 (Origin server)

Microsoft IIS Origin-palvelin asennetaan hyvin samankaltaisesti kuin reunapalvelimet, joskin hiukan yksinkertaistetummin. Origin-palvelin ei tarvitse tietoa seuraavan tason palvelimista, sillä Origin-palvelin on se palvelin, jolta data viime kädessä haetaan. Ensimmäisenä asetetaan ”hosts”-tiedosto toimintakuntoon kuvion 38 mukaisesti. Hosts-tiedosto löytyy seuraavasta kansiorakenteesta:

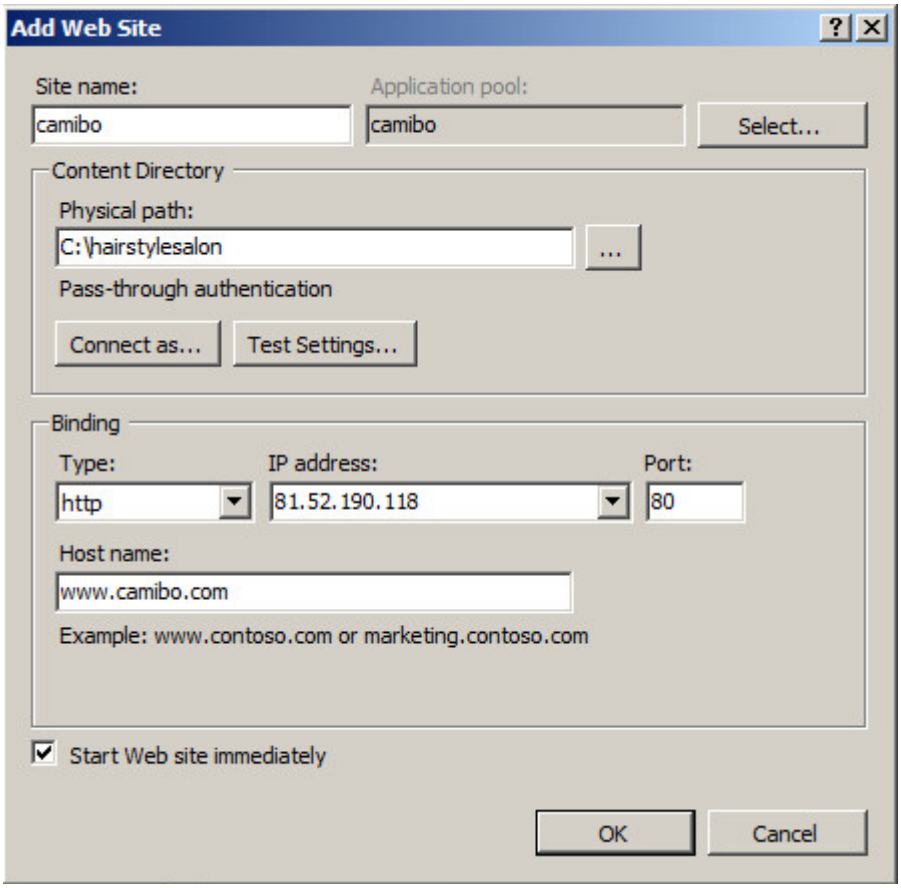
C:\Windows\System32\drivers\etc

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#       127.0.0.1       localhost
#       ::1             localhost

81.52.190.118    www.camibo.com
```

Kuvio 38 Origin-palvelimen hosts-tiedosto

Haettaessa osoitetta DNS-nimelle ”www.camibo.com”, Origin-palvelin osoittaa it-seensä.

Origin-palvelin tarvitsee ainoastaan toimivan web-sivuston, joka toimii alkuperäisenä mallina sivustolle, jonka reunapalvelimet lataavat. Tämän sivuston saa toimintaan IIS hallintapaneelista ”Sites”-lehden alta: ”Add Web Site…”. Sivusto on asetettu toimimaan kuvion 39 parametrein.



Kuvio 39 Origin-palvelimen web-sivuston lisäys

Origin-palvelimen WEB-konfiguraatio ei eroa mitenkään oleellisesti reunapisteiden konfiguraatiosta, ainoastaan IP-osoite on vaihtunut.

5 Varmentaminen

5.1 Komentoja

Sivuston toiminta on ympäri maapalloa näennäisesti sama: käyttäjä avaa selaimen ja kirjoittaa osoiteriville osoitteen ”www.camibo.com”, jolloin samannäköinen sivusto aukeaa, oli käyttäjä missä vain. Loppukäyttäjälle kuormantasaajan toiminta on käytännössä näkymätöntä, mutta esimerkiksi komennoilla

nslookup

dig @IP-osoite www.camibo.com

voidaan nähdä, kuinka saadun palvelimen IP-osoitteistus muuttuu riippuen DNS-palvelimen maantieteellisestä lokaatiosta.

5.1.1 NSLOOKUP komento

Nslookup-komennon parhaana puolena voidaan pitää yleistä saatavuutta. Komento on käytettävissä lähes joka järjestelmässä. Vaikka dig-komento on paljon monipuolisempi, on se harvinainen Microsoft Windows-ympäristöissä. Nslookup-komento saattaa vaikuttaa hyvin yksinkertaiselle, mutta etenkin ”-all” option avulla saadaan näkyviin paljon lisäparametreja kyseiselle komennolle. (Aitchison, R. 2011, chapter 9)

Kuviossa 40 käytetty nslookup-komento palauttaa vastauksen kyselylle

nslookup www.camibo.com

Kyseinen DNS-kysely on lähetetty ensin omalle DNS-palvelimelle, jolta vastaus on lopulta saapunut.

```
[sulo@aasia ~]$ nslookup www.camibo.com
Server:         202.236.167.5
Address:        202.236.167.5#53

Non-authoritative answer:
Name:   www.camibo.com
Address: 192.88.171.186
```

Kuvio 40 nslookup-komennon esittäminen

5.1.2 DIG komento

Dig-komento on tällä hetkellä DNS-diagnostiikassa käytetyin työkalu. Yleisestikin ottaen dig-komenton on paljon tehokkaampi työkalu kuin nslookup, joskin saatavuus

etenkin Microsoft Windows-järjestelmillä voi olla heikohkoa. (Aitchison, R. 2011, chapter 9)

Kuvion 41 tapauksessa dig-komento palauttaa vastauksen kyselyyn

dig @202.236.167.5

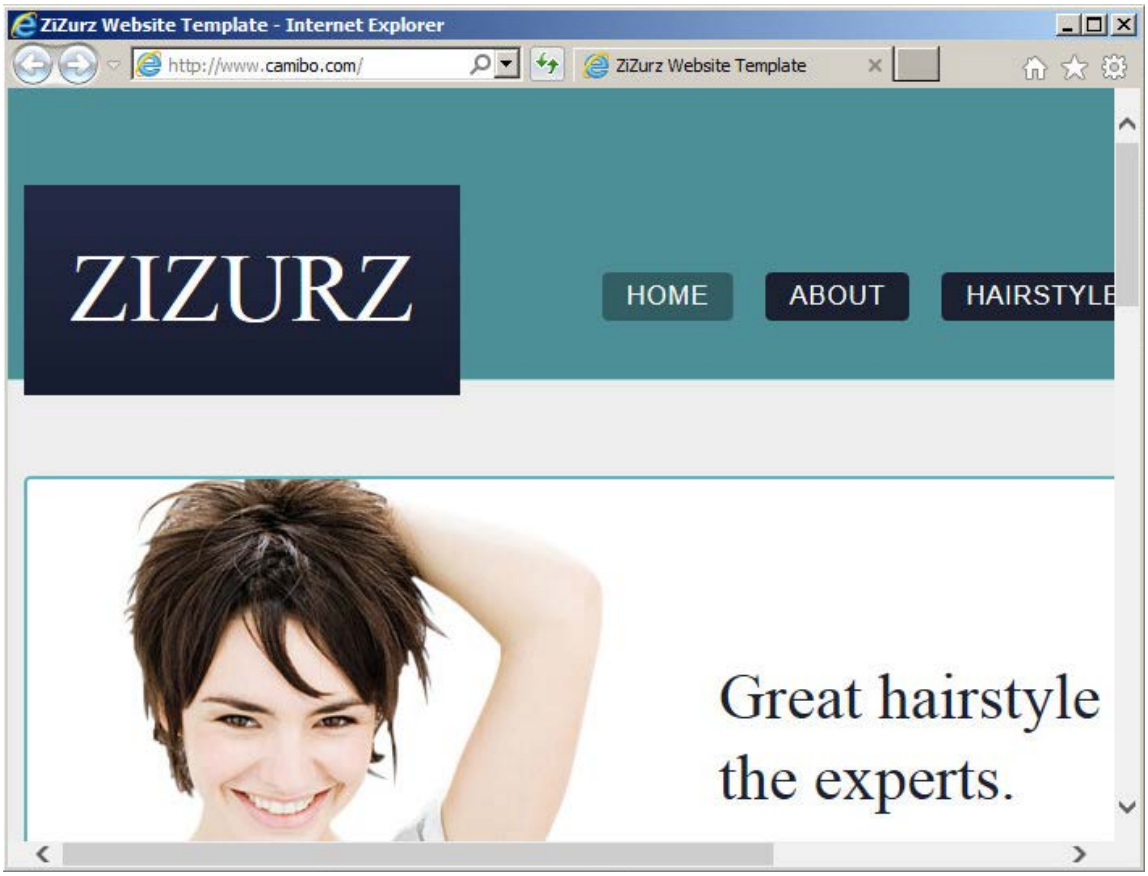
Palautetussa vastauksessa on saatu tietoon käytössä olevat root-palvelimet.

```
[sulo@aasia ~]$ dig @202.236.167.5

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @202.236.167.5
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32431
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;.                              IN      NS

;; ANSWER SECTION:
.                       214     IN      NS      b.root-servers.jst.
.                       214     IN      NS      c.root-servers.jst.
.                       214     IN      NS      a.root-servers.jst.

;; Query time: 1 msec
;; SERVER: 202.236.167.5#53(202.236.167.5)
;; WHEN: Tue Apr  8 11:07:59 2014
;; MSG SIZE  rcvd: 78
```

Kuvio 41 dig-komento

5.2 Bind9 toiminnan todentaminen

Loppukäyttäjälle toiminnallisuus näkyy vain kuvion 42 näköisenä sivustona. Sivusto on täysin identtinen Aasian, Euroopan ja Yhdysvaltojen palvelimien käyttäjille, joten toiminnallisuutta täytyy tutkia käyttäen DNS-kyselyitä. Kyseisen www-sivuston pohja on ladattu sivustolta http://www.freewebsitetemplates.com/ ja löytyy nimellä ”Hais-tyle Salon”.

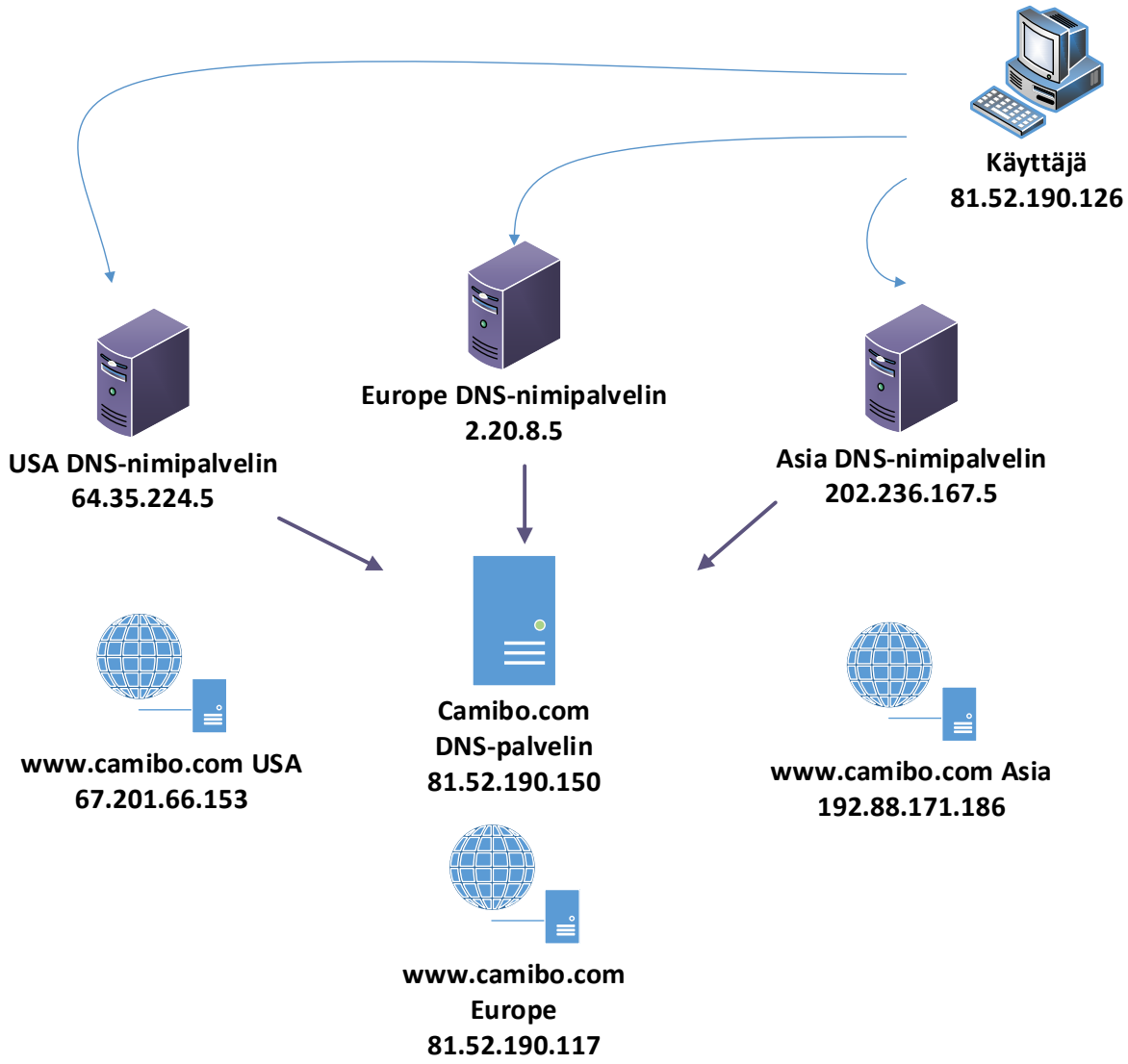


Kuvio 42 Sivustonäkymä loppukäyttäjälle

BIND-järjestelmän toimintaa todensin käyttäen dig-komentoa seuraavin parametrein:

dig @"alueenDNSpalvelimenIP" www.camibo.com

Kyseisellä komennolla täytyisi saada eri vastaus riippuen, miltä DNS-palvelimelta vastausta haetaan. Testissä kyselijäkone ei siis vaihtanut paikkaansa, vaan ainoastaan halutun DNS-palvelimen osoite muuttui. Kuviossa 43 on esitettynä kyseinen prosessi.



Kuvio 43 Bind9 toiminnan testaus

Ensimmäisen testin suoritin tekemällä kyselyn Japanissa sijaitsevalle DNS-palvelimelle. Aiemmin esitetty komento antoi kuvion 44 mukaisen vastauksen. Tärkein tieto kyseisestä tulosteesta on ”;; ANSWER SECTION:”, josta selviää että kysytty osoite ”www.camibo.com” sijaitsee IP-osoitteessa 192.88.171.186. Tämä tulos on hyväksyttävä, sillä kyseinen IP-osoite kuuluu camibo:n Aasian web-palvelimelle. Aasian ja Oseanian käyttäjät on asetettu kyseisen palvelimen käyttäjäkuntaan.

```
[sulo@pilotcentos ~]$ dig @202.236.167.5 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @202.236.167.5 www.camibo
.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60352
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         300     IN      A       192.88.171.186

;; AUTHORITY SECTION:
camibo.com.             10      IN      NS      ns1.camibo.com.

;; ADDITIONAL SECTION:
ns1.camibo.com.         477     IN      A       81.52.190.150

;; Query time: 12 msec
;; SERVER: 202.236.167.5#53(202.236.167.5)
;; WHEN: Tue Apr 15 13:25:11 2014
;; MSG SIZE  rcvd: 82
```

Kuvio 44 Bind, Dig Aasian DNS palvelimelta

Kuviossa 45 on varmennettuna vielä, että kyselyt saapuvat itse BIND-palvelimelle. Palvelimella oli suoritettu komento:

tcpdump-i eth0-vvv-n port 53

jolloin tulevat DNS-kyselyt saadaan tulostettua reaaliaikaisesti. Riviltä 3 näkee, että kyselyt saapuvat rekursiiviselta palvelimelta osoitteessa ”202.236.167.5” ja kolmanneksi alimmalla rivillä annetaan vastaus ”192.88.171.186”.

```
13:24:11.384597 IP (tos 0x0, ttl 61, id 44310, offset 0, flags [none], proto UDP
 (17), length 71)
    202.236.167.5.37853 > 81.52.190.150.domain: [udp sum ok] 26625% [1au] A? www
.camibo.com. ar: . OPT UDPsize=4096 OK (43)
13:24:11.391273 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto UDP (17),
 length 121)
    81.52.190.150.domain > 202.236.167.5.37853: [bad udp cksum a5d1!] 26625*- q:
 A? www.camibo.com. 1/1/2 www.camibo.com. [5m] A 192.88.171.186 ns: camibo.com.
[10s] NS ns1.camibo.com. ar: ns1.camibo.com. [10s] A 81.52.190.150, . OPT UDPsiz
e=4096 OK (93)
```

Kuvio 45 Bind, tcpdump sisältö Aasian DNS kyselylle

Seuraava kysely suoritettiin Euroopan rekursiiviselle DNS-palvelimelle osoitteessa 2.20.8.5. Vastaukseksi saatiin oletettu arvo ”81.52.190.117”, joka on Euroopan WEB-palvelimen IP-osoite. Tulosteen kyselylle voikin nähdä kuviosta 46.

```
[sulo@pilotcentos ~]$ dig @2.20.8.5 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @2.20.8.5 www.camibo.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9294
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         300     IN      A       81.52.190.117

;; AUTHORITY SECTION:
camibo.com.             10      IN      NS      ns1.camibo.com.

;; ADDITIONAL SECTION:
ns1.camibo.com.         900     IN      A       81.52.190.150

;; Query time: 17 msec
;; SERVER: 2.20.8.5#53(2.20.8.5)
;; WHEN: Tue Apr 15 13:15:23 2014
;; MSG SIZE  rcvd: 82
```

Kuvio 46 Bind, Dig Euroopan DNS palvelimelta

Kuviossa 47 on nähtävillä kuviota 45 vastaavaa tulostetta. Kolmannen rivin DNS-palvelimen osoite vastaa Euroopan rekursiivisen DNS-palvelimen osoitetta ja kolmanneksi alimmalla rivillä saadaankin täysin oikea IP-osoite ”81.52.190.117”.

```
13:14:24.024039 IP (tos 0x0, ttl 63, id 4007, offset 0, flags [none], proto UDP
(17), length 71)
    2.20.8.5.10897 > 81.52.190.150.domain: [udp sum ok] 35693% [1au] A? www.cami
bo.com. ar: . OPT UDPsize=4096 OK (43)
13:14:24.034896 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto UDP (17),
 length 121)
    81.52.190.150.domain > 2.20.8.5.10897: [bad udp cksum a243!] 35693*- q: A? w
ww.camibo.com. 1/1/2 www.camibo.com. [5m] A 81.52.190.117 ns: camibo.com. [10s]
NS ns1.camibo.com. ar: ns1.camibo.com. [10s] A 81.52.190.150, . OPT UDPsize=4096
 OK (93)
```

Kuvio 47 Bind, tcpdump sisältö Euroopan DNS kyselylle

Viimeisessä BIND-testauksessa tehtiin dig-komento Yhdysvalloissa sijaitsevalle rekursiiviselle DNS-palvelimelle osoitteessa 64.35.224.5, jonka tulosteen voikin nähdä kuviosta 48. Tuloste on hyväksyttävä, sillä Yhdysvaltojen DNS-palvelimen käyttäjät siirretään Yhdysvalloissa sijaitsevalle web-palvelimelle IP-osoitteessa 67.201.66.153.

```
[sulo@pilotcentos ~]$ dig @64.35.224.5 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @64.35.224.5 www.camibo.c
om
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5346
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         10      IN      A       67.201.66.153

;; AUTHORITY SECTION:
camibo.com.             10      IN      NS      ns1.camibo.com.

;; ADDITIONAL SECTION:
ns1.camibo.com.         900     IN      A       81.52.190.150

;; Query time: 4 msec
;; SERVER: 64.35.224.5#53(64.35.224.5)
;; WHEN: Tue Apr 15 13:26:35 2014
;; MSG SIZE  rcvd: 82
```

Kuvio 48 Bind, Dig Yhdysvaltain DNS palvelimelta

Kuviossa 49 on todennettuna, että kyselyt saapuvat Amerikan rekursiiviselta DNS-palvelimelta camibo:n autoritääriselle DNS-palvelimelle. IP-osoitteet vastaavat jälleen kolmannella, että kolmanneksi viimeisellä rivillä.

```
13:26:21.513291 IP (tos 0x0, ttl 62, id 16720, offset 0, flags [none], proto UDP
 (17), length 71)
    64.35.224.5.hermes > 81.52.190.150.domain: [udp sum ok] 55600% [1au] A? www.
camibo.com. ar: . OPT UDPsize=4096 OK (43)
13:26:21.513542 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto UDP (17),
 length 121)
    81.52.190.150.domain > 64.35.224.5.hermes: [bad udp cksum d979!] 55600*- q:
A? www.camibo.com. 1/1/2 www.camibo.com. [10s] A 67.201.66.153 ns: camibo.com. [
10s] NS ns1.camibo.com. ar: ns1.camibo.com. [10s] A 81.52.190.150, . OPT UDPsize
=4096 OK (93)
```

Kuvio 49 Bind, tcpdump sisältö Amerikan DNS kyselylle

5.3 F5 BIG-IP GTM

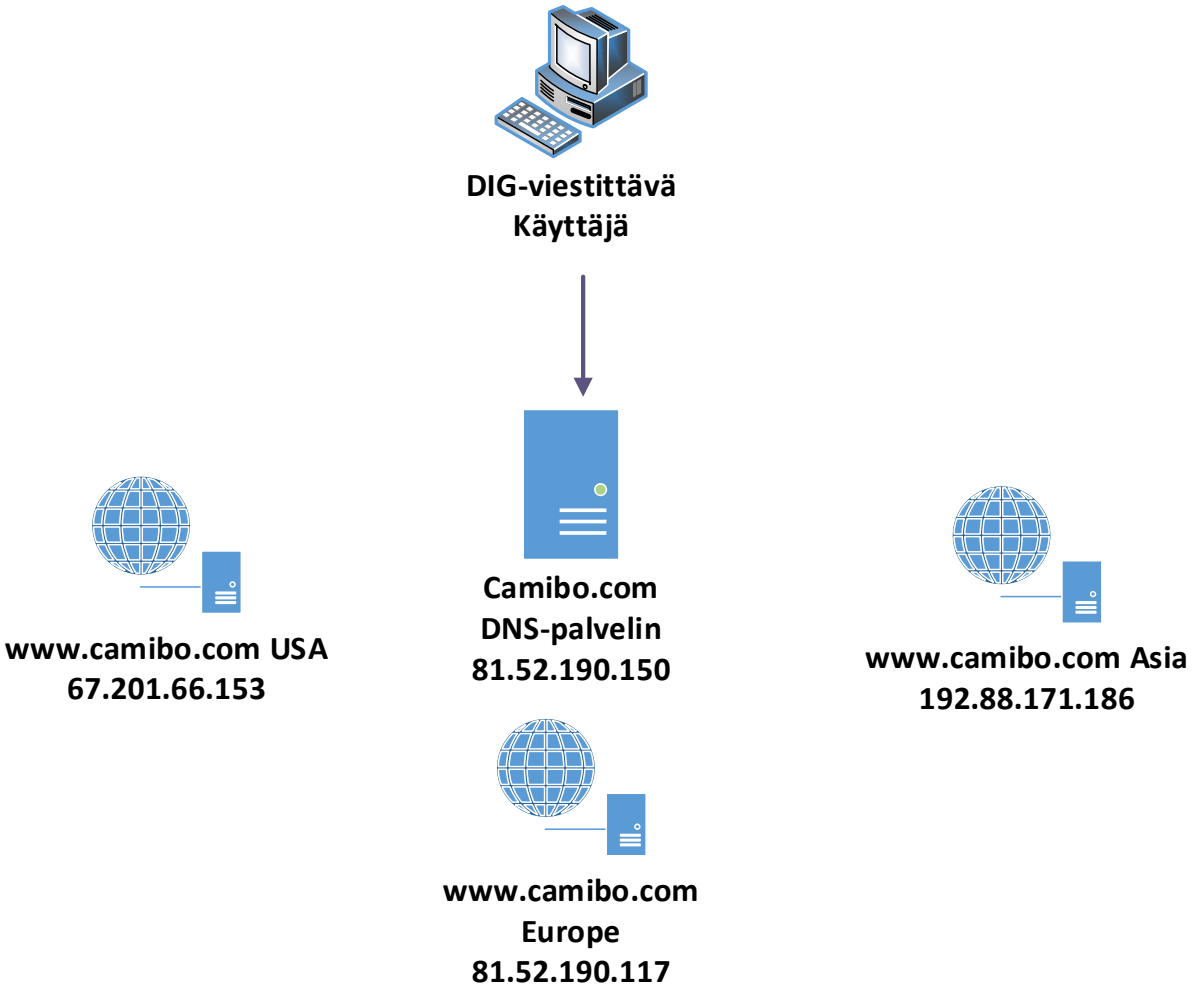
5.3.1 Todennus DIG-viestein

F5 BIG-IP GTM-järjestelmän toimintaa todensin käyttäen dig-komentoa seuraavin parametrein:

dig @81.52.190.150 www.camibo.com

Kyseisellä komennolla täytyisi saada eri vastaus riippuen, missä päin maailmaa kyselijäkone sijaitsee. Testeissä kysyin vastausta suoraan camibo.com:n autoritääriseltä palvelimelta. Vaikka IP-osoitteet ovat erilaiset, kuin BIND testauksissa, vastausten täytyisi olla samat.

Kuvion 50 DIG-viestittävä käyttäjä vaihtaa paikkaansa maailmalla jokaisessa testausvaiheessa. Koneelta suoritetaan DIG-komento osoitteeseen 81.52.190.150, jolloin riippuen käyttäjän maantieteellisestä sijainnista camibo.com autoritäärinen DNS-palvelin tarjoaa eri IP-osoitteen.



Kuvio 50 DIG-viestit suoraan camibo.com autoritääriselle palvelimelle

Ensimmäisen testin suoritin siirtämällä testikoneen Japanin verkkoon. Aiemmin esitetty komento antoi kuvion 51 mukaisen vastauksen. Tärkein tieto kyseisestä tulosteesta on ”;; ANSWER SECTION:”, josta selviää että kysytty osoite ”www.camibo.com” sijaitsee IP-osoitteessa 192.88.171.186. Tämä tulos on hyväksyt-

tävä, sillä kyseinen IP-osoite kuuluu camibo:n Aasian web-palvelimelle. Aasian ja Oseanian käyttäjät on asetettu kyseisen palvelimen käyttäjäkuntaan.

```
[sulo@aasia ~]$ dig @81.52.190.150 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @81.52.190.150 www.camibo
.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25897
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         30      IN      A       192.88.171.186

;; Query time: 6 msec
;; SERVER: 81.52.190.150#53(81.52.190.150)
;; WHEN: Fri Apr 11 13:47:34 2014
;; MSG SIZE  rcvd: 48
```

Kuvio 51 F5, Dig Japanista F5 DNS:lle.

Seuraavassa testissä asetin samaisen koneen Ranskan verkkoon. Aiemmin esitetty kysely palauttaa kuvion 52 tulosteen. Tämäkin tulos on hyväksyttävä, sillä Euroopan käyttäjät on asetettu käyttämään palvelinta IP-osoitteessa 81.52.190.117.

```
[sulo@aasia ~]$ dig @81.52.190.150 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @81.52.190.150 www.camibo
.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36662
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         30      IN      A       81.52.190.117

;; Query time: 1 msec
;; SERVER: 81.52.190.150#53(81.52.190.150)
;; WHEN: Fri Apr 11 15:28:02 2014
;; MSG SIZE  rcvd: 48
```

Kuvio 52 F5, Dig Ranskasta F5 DNS:lle

Viimeisessä dig-komennon testauksessa asetin testikoneen vielä Yhdysvaltojen verkkoon. Tästä sain kuvion 53 mukaista tulostetta, joka on taaskin hyväksyttävää, sillä Amerikan käyttäjät on osoitettu IP-osoitteeseen 67.201.66.153.

```
[sulo@aasia ~]$ dig @81.52.190.150 www.camibo.com

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 <<>> @81.52.190.150 www.camibo
.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16602
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.camibo.com.                        IN      A

;; ANSWER SECTION:
www.camibo.com.         30      IN      A       67.201.66.153

;; Query time: 3 msec
;; SERVER: 81.52.190.150#53(81.52.190.150)
;; WHEN: Fri Apr 11 13:37:02 2014
;; MSG SIZE  rcvd: 48
```

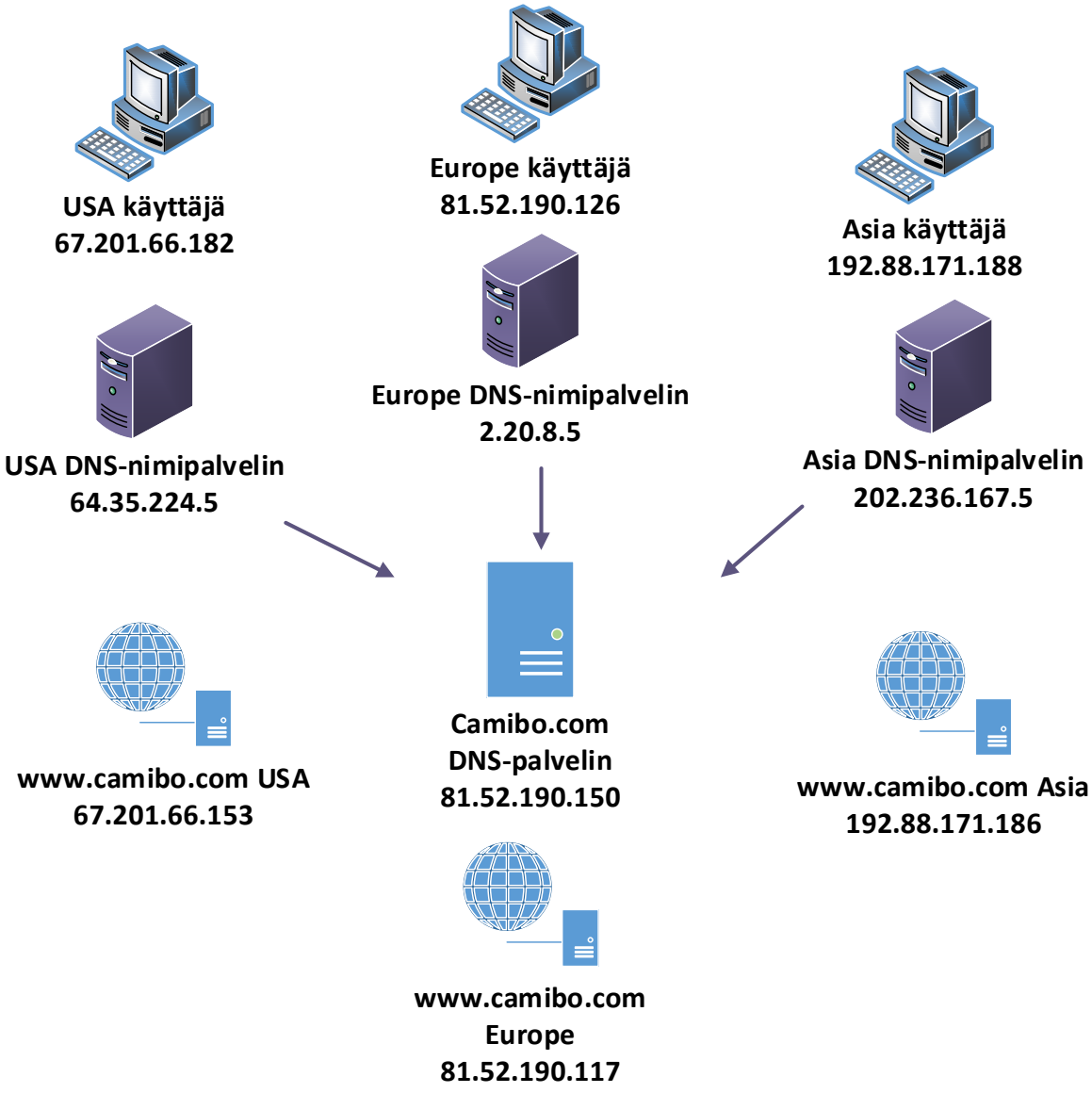
Kuvio 53 F5, Dig Yhdysvalloista F5 DNS:lle

5.3.2 Todennus F5 & WserverCache @USA

Ennen jokaista maantieteellistä testiä, tyhjensin F5 BIG-IP GTM-järjestelmän statistiikkatiedot, jotta kyselyt pystytään varmentamaan ilman turhaa infoa ja vain tärkeä informaatio jää näkyviin. Tyhjennys suoritettiin siis ennen testejä kaikissa kolmessa todennusvaiheessa.

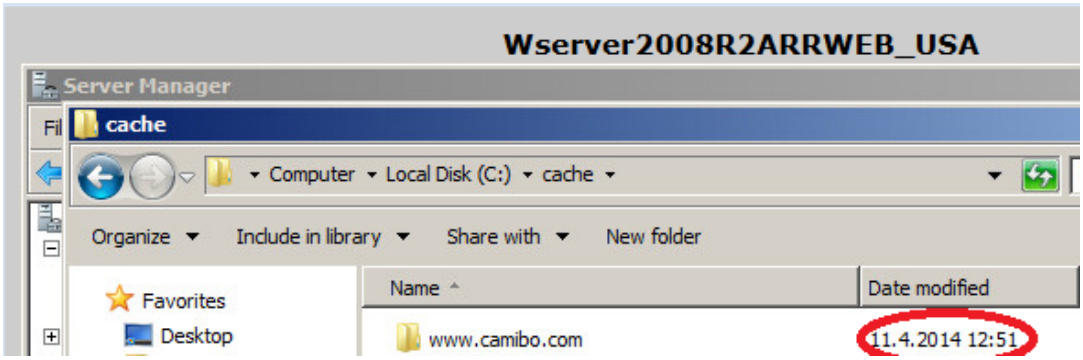
Kuviossa 54 on topologiakuva suoritetuille testeille. Samalla linjalla pystysuunnassa olevat koneet sijaitsevat samalla maantieteellisellä sijainnilla. Käyttäjän DNS-nimenselvityspyyntö siirtyy ensin paikalliselle DNS-nimipalvelimelle. Paikallinen DNS-nimipalvelin suorittaa tämän jälkeen DNS-nimenselvityspyynnön eteenpäin camibo.com domainin autoritääriselle DNS-palvelimelle. camibo.comin autoritäärinen DNS-nimipalvelin vastaa DNS-nimenselvityspyyntöön perustuen saapuvan paketin lähettäjän IP-osoitteeseen ja antaa vastauksen perustuen IP-osoitteen oletettuun maantieteelliseen sijaintiin. Vastaus lähetetään takaisin paikalliselle DNS-

nimipalvelimelle ja siitä takaisin käyttäjälle, joka voi lopulta ottaa yhteyden lähimmälle HTTP-palvelimelle.



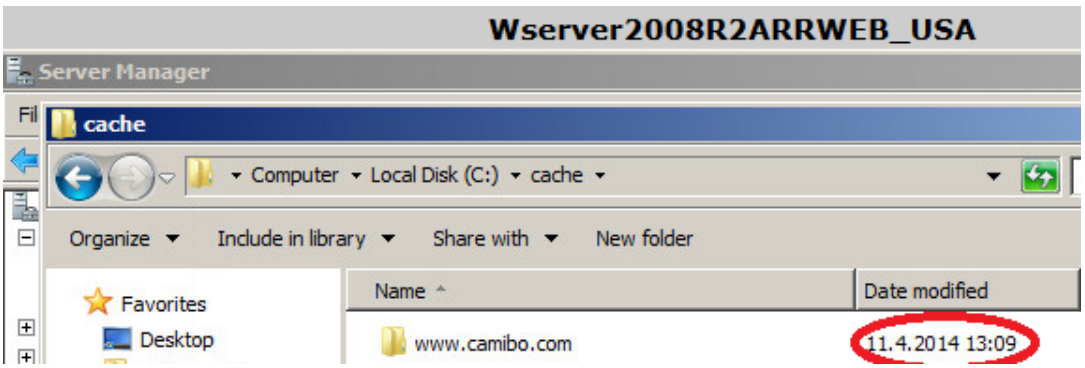
Kuvio 54 Topologiakuva F5 ja ARR todennuksille

Windows Server 2008R2 ARR IIS-palvelinten toiminnan testauksen tein F5-järjestelmien testauksen ohessa. Testauksissa todensin välimuistin toimintaa. Ensimmäinen testaus palveluittain tehtiin asettamalla käyttäjäkone Yhdysvaltoihin (IP), josta otettiin selaimella yhteys osoitteeseen ”www.camibo.com”. Välimuistin vanhenemisajaksi oli asetettu 10 minuuttia, joten uuden yhteyden oton jälkeen ”Date Modified” arvo tulee muuttumaan, kun yhteys otetaan uudestaan 10 minuutin tai myöhemmän ajan kohden jälkeen. Kuviosta 55 näkee, että ensimmäisen kerran sivusto on ladattu kyseiseltä palvelimelta kello 12:51.



Kuvio 55 ARRWEB_USA camibo-sivusto

Kun sivustoon otetaan uudestaan yhteys 18 minuutin päästä uudelleen, huomaakin että ”Date modified” arvo on muuttunut, kuten kuviosta 56 näkeekin. Muutos johtuu siitä, että caching time on nollaantunut ja sivusto on ladattu uudestaan Origin toimipisteeltä.



Kuvio 56 ARRWEB_USA camibo caching timen loputtua

Kuviossa 57 kone on asetettu Yhdysvalloissa sijaitsevaan verkkoon. IP-osoite on 67.201.66.182, joka on Yhdysvaltain IP-osoitealueella.

```
[sulo@aasia ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:56:01:0A:20
          inet addr:67.201.66.182  Bcast:67.201.67.255  Mask:255.255.254.0
```

Kuvio 57 testikoneen ifconfig tuloste Amerikassa

Seuraavaksi kyseiseltä tietokoneelta on tehty komento ”nslookup www.camibo.com”, jolla on saatu yksinkertaisesti vastaus 67.201.66.153, kuten kuviossa 58 on esitetty. Tämä on se osoite, jonka Yhdysvalloissa sijaitsevan käyttäjän kuuluisikin saada.

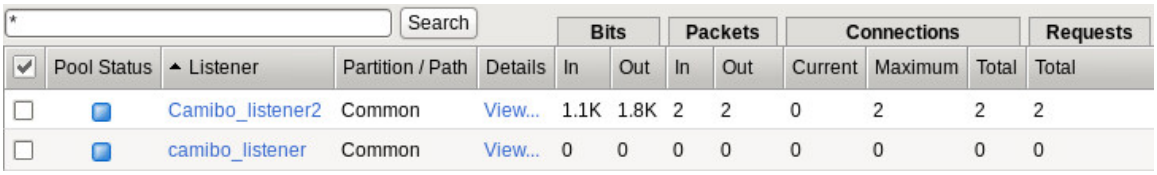
```
[sulo@aasia ~]$ nslookup www.camibo.com
Server:         64.35.224.5
Address:        64.35.224.5#53

Non-authoritative answer:
Name:   www.camibo.com
Address: 67.201.66.153
```

Kuvio 58 testikoneen nslookup tuloste Amerikassa

F5 BIG-IP GTM DNS-kuuntelijan statistiikkaa saa esille välilehdeltä:

DNS-> DELIVERY-> Listeners-> Statistics

Kyseisestä statistiikasta käy ilmi, että nslookup-komennon jälkeen kone on saanut pyyntöjä (Requests) kaksi kappaletta. Tämä on esitettynä kuviossa 59.



* Search				Bits		Packets		Connections			Requests	
☑	Pool Status	▲ Listener	Partition / Path	Details	In	Out	In	Out	Current	Maximum	Total	Total
☐	■	Camibo_listener2	Common	View...	1.1K	1.8K	2	2	0	2	2	2
☐	■	camibo_listener	Common	View...	0	0	0	0	0	0	0	0

Kuvio 59 F5 BIG-IP GTM listener statistiikka Amerikan kyselyille

F5 BIG-IP GTM GSLB-statistiikkaa saa välilehdeltä:

DNS-> GSLB-> Wide IP-> Statistics

Kyseiseltä välilehdeltä aukeaa kuvion 60 kaltainen näkymä. Tällä voidaan varmentaa, että kyseiselle FQDN:lle on tullut pyyntöjä kaksi kappaletta, joista yksi on selvitetty (resolved).

* Search						Requests	
☑	Status	▲ Wide IP	Partition / Path	Details	Pools	Total	Resolved
☐	■	www.camibo.com	Common	View...	View...	2	1

Kuvio 60 F5 BIG-IP GTM Wide-IP statistiikkaa Amerikan kyselyille

Kuviosta 61 käy ilmi, että toinen pyynnöistä on AAAA tyyppinen, eli IPv6 osoitteen-selvityspyyntö. Tätä ei ole selvitetty, vaan ainoastaan IPv4 pyyntöön on vastattu. ”Load Balancing”, eli kuormantasauspalkista näkee myös, että yksi pyynnöistä on kuormantasattu perustuen aiemmin säädettyihin parametreihin.

Requests	
Total	2
Persisted	0
A	1
AAAA	1
Resolved	1
Dropped	0
Load Balancing	
Preferred	1

Kuvio 61 F5 BIG-IP GTM Wide-IP statistiikkaa Details välilehdeltä

Kuviossa 62 on Pools tason tilastoja, jotka saa näkyviin välilehdiltä:

DNS-> GSLB-> Pools-> Statistics

Kuormantasaus on toteutettu Pools tasolle, joten kuviosta 62 näkeekin, että Yhdysvalloissa sijaitsevan hostin tehdessä ”nslookup www.camibo.com” komennon, tasataan se ”Camibo_USA_Pool” pooliin, joka nimensä mukaisesti kuormantasaa Yhdysvaltoihin.

*	Search						Load Balancing		
	Status	Pool	Partition / Path	Members	Preferred	Alternate	Fallback	Returned to DNS	
		Camibo_Asia_Pool	Common	View...	0	0	0	0	
		Camibo_Europa_Pool	Common	View...	0	0	0	0	
		Camibo_USA_Pool	Common	View...	1	0	0	0	
		Camipool	Common	View...	0	0	0	0	

Kuvio 62 F5 BIG-IP GTM Pool statistiikkaa

Kuviossa 63 on menty astetta alemmaksi kuormantasaushierarkiassa. Poolien alta löytyvät itse ”palvelimet”, joihin voidaan suorittaa kuormantasaus vielä erilleen. Tässä tapauksessa jokaisen Poolin alla on vain yksi palvelin, joten valinta on helppo: palvelimeksi on valittu ”Camibo_USA_Web”, kuten ”Pics” osion alta voi nähdä. Nämä statistiikat voi nähdä välilehdeltä:

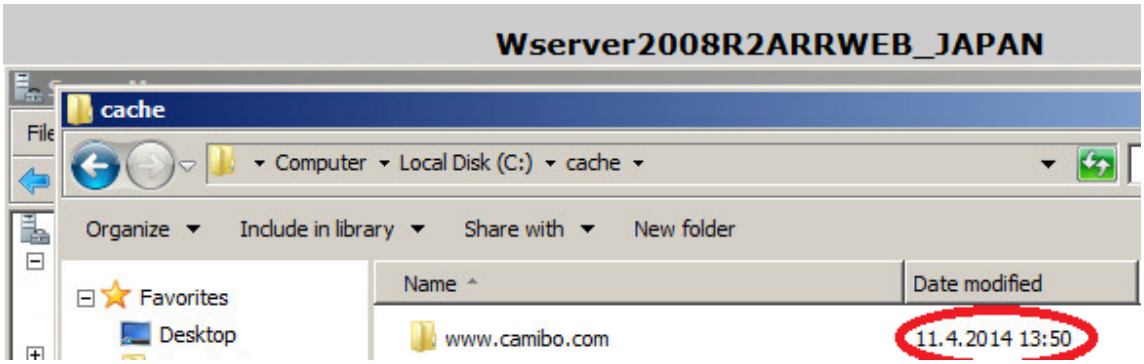
DNS-> GSLB-> Servers-> Statistics

*	Search							Throughput (bits/sec)	
	Status	Server	Address	Partition / Path	Virtual Servers	Picks	Connections	In	Out
		Camibo_Asia_Web	192.88.171.186	Common	View...	0	0	0	0
		Camibo_Europa_Web	81.52.190.117	Common	View...	0	0	0	0
		Camibo_USA_Web	67.201.66.153	Common	View...	1	0	0	0

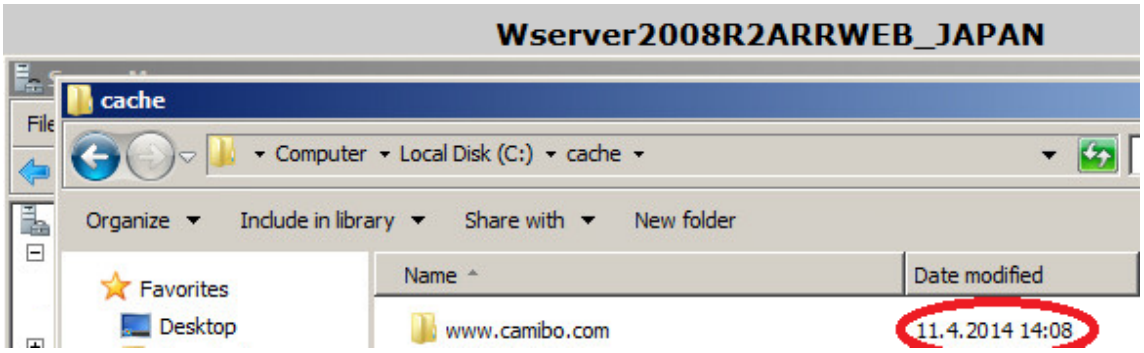
Kuvio 63 F5 BIG-IP GTM Palvelin statistiikkaa Amerikan kyselyille

5.3.3 Todennus F5 & WserverCache @Asia

Kuten USA:n todennusvaiheessa totesin, Microsoft Windows Server 2008R2-todennukset teen F5-laitteiden todennusten ohessa. Myös Japanin palvelimen väli-muistin sisältö uusitaan sen vanhetessa, kuten kuvioista 64 ja 65 ”Date modified” ajankohdista voidaan todentaa.



Kuvio 64 ARRWEB_JAPAN camibo-sivusto



Kuvio 65 ARRWEB_JAPAN camibo caching timen loputtua

Kuviossa 66 on testikoneen IP-osoitetiedot koneen kytkeytyessä Aasian. IP-osoite on 192.88.171.188, joka on Aasian IP-osoitealueelta.

```
[sulo@aasia ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:56:01:0A:20
          inet addr:192.88.171.188  Bcast:192.88.171.255  Mask:255.255.255.0
          inet6 addr: fe80::250:56ff:fe01:a20/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:78 errors:0 dropped:0 overruns:0 frame:0
          TX packets:19 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9642 (9.4 KiB)  TX bytes:2454 (2.3 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:12 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:720 (720.0 b)  TX bytes:720 (720.0 b)
```

Kuvio 66 testikoneen ifconfig tuloste Aasiassa

Seuraavaksi kyseiseltä tietokoneelta on tehty komento ”nslookup www.camibo.com”, jolla on saatu vastaus 192.88.171.186. Kyseinen osoite kuuluu Camibon Aasian reunapalvelimelle. Tämä on varmistettavissa kuviosta 67.

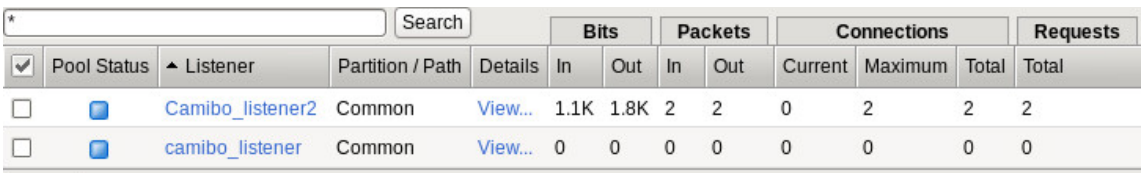
```
[sulo@aasia ~]$ nslookup www.camibo.com
Server:         202.236.167.5
Address:        202.236.167.5#53

Non-authoritative answer:
Name:   www.camibo.com
Address: 192.88.171.186
```

Kuvio 67 testikoneen nslookup tuloste Aasiassa

F5 BIG-IP GTM DNS-kuuntelijan statistiikkaa saa esille välilehdeltä:

DNS-> DELIVERY-> Listeners-> Statistics

Kyseisestä statistiikasta käy ilmi, että nslookup-komennon jälkeen kone on saanut pyyntöjä (Requests) kaksi kappaletta. Tämä on esitettynä kuviossa 68.



Kuvio 68 F5 BIG-IP GTM listener statistiikka Aasian kyselyille

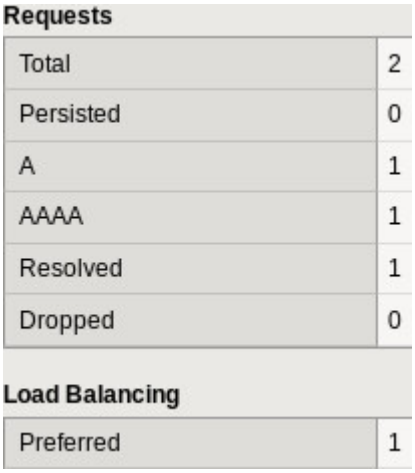
F5 BIG-IP GTM GSLB-statistiikkaa saa välilehdeltä:

DNS-> GSLB-> Wide IP-> Statistics

Kyseiseltä välilehdeltä aukeaa kuvion 69 kaltainen näkymä. Tällä voidaan varmentaa, että kyseiselle FQDN:lle on tullut pyyntöjä kaksi kappaletta, joista yksi on selvitetty (resolved).

Kuvio 69 F5 BIG-IP GTM Wide-IP statistiikkaa Aasian kyselyille

Kuten Amerikan kyselyissä, vain IPv4-pyyntöön on vastattu ja kyseiselle vastaukselle on toteutettu kuormantasausprosessi. Kuvio 70 on täysin samanlainen kuin Amerikan vastaava kuvio 61.

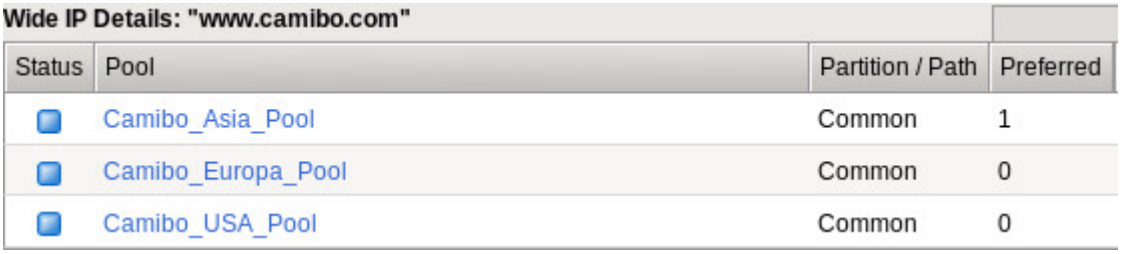


Kuvio 70 F5 BIG-IP GTM Wide-IP statistiikkaa Details välilehdeltä

Kuviossa 71 on Pools tason tilastoja, jotka saa näkyviin välilehdiltä:

DNS-> GSLB-> Pools-> Statistics

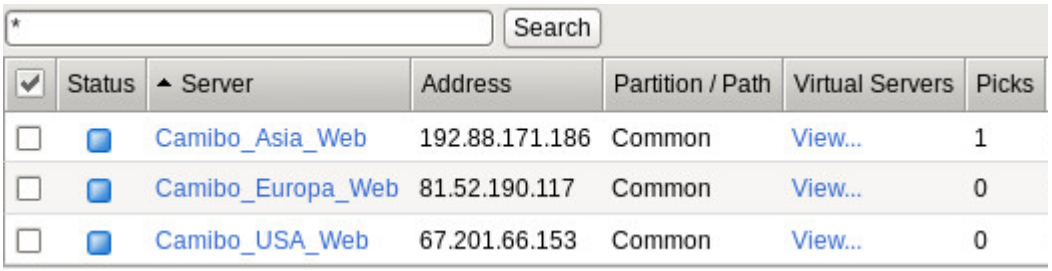
Kuten Amerikan tapauksessa, ensimmäinen kuormantasaus tehdään Pools tasolla. Tässä tapauksessa kuormantasaus on osunut kohteeseen ”Camibo_Asia_Pool”, joka on täysin oikea valinta, sillä kysely saapuu Aasiasta.



Kuvio 71 F5 BIG-IP GTM Palvelin statistiikkaa Aasian kyselyille

Seuraavaksi on menty alemmaksi kuormantasaushierarkiassa kuten Amerikan kohdallakin. Ja kuten aiemmin, järjestelmiä on vain yksi jokaisen Poolin alla, joten valinta on helppo. Tässä tapauksessa on valittu ”Camibo_Asia_Web” ja kyseinen valinta on oikea. Kuvion 72 statistiikkanäkymään pääsee välilehdiltä:

DNS-> GSLB-> Servers-> Statistics

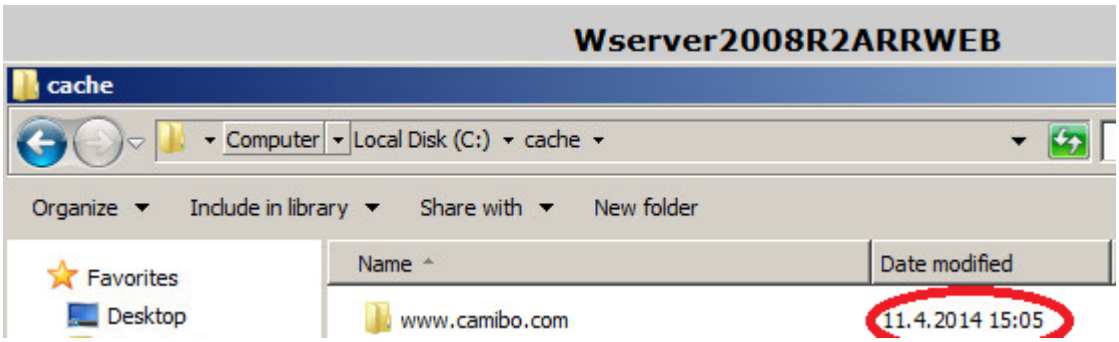


Kuvio 72 F5 BIG-IP GTM Palvelin statistiikkaa Aasian kyselyille

5.3.4 Todennus F5 & WserverCache @Europe

Viimeiset testaukset suoritin Euroopassa Ranskan palvelimelle. Kuten aiemmissa todennuksissa myös Ranskan palvelimen välimuistin sisältö uusitaan sen vanhetessa, kuten kuvioista 73 ja 74 ”Date modified” ajankohdista voidaan todentaa.

Kuvio 73 ARRWEB camibo-sivusto (Eurooppa)



Kuvio 74 ARRWEB camibo caching timen loputtua (Eurooppa)

Kuviossa 75 on testikoneen IP-osoitetiedot koneen kytkeytyessä Eurooppaan. IP-osoite on 81.52.190.126, joka on Euroopan IP-osoitealueelta.

```
[sulo@aasia ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:56:01:0A:20
          inet addr:81.52.190.126  Bcast:81.52.190.255  Mask:255.255.255.0
          inet6 addr: fe80::250:56ff:fe01:a20/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:21 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2313 (2.2 KiB)  TX bytes:2094 (2.0 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:12 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:720 (720.0 b)  TX bytes:720 (720.0 b)
```

Kuvio 75 testikoneen ifconfig tuloste Euroopassa

Seuraavaksi kyseiseltä tietokoneelta on tehty komento ”nslookup www.camibo.com”, jolla on saatu vastaus 81.52.190.117, kuten kuviossa 76 on esitetty. Kyseinen osoite kuuluu Camibon Euroopan reunapalvelimelle.

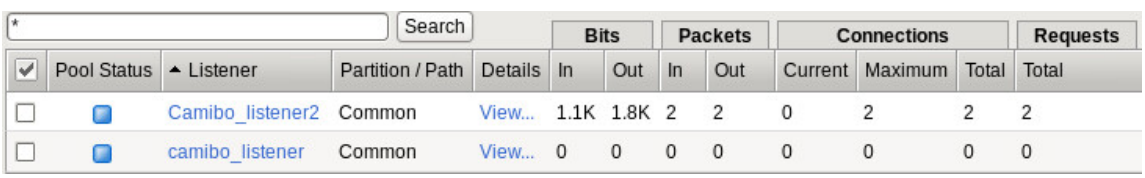
```
[sulo@aasia ~]$ nslookup www.camibo.com
Server:         2.20.8.5
Address:        2.20.8.5#53

Non-authoritative answer:
Name:   www.camibo.com
Address: 81.52.190.117
```

Kuvio 76 testikoneen nslookup tuloste Euroopassa

F5 BIG-IP GTM DNS-kuuntelijan statistiikkaa saa esille välilehdeltä:

DNS-> DELIVERY-> Listeners-> Statistics

Kyseinen statistiikka on nähtävissä kuviossa 77. Tämä tuloste ei eroa kyselyiden osalta aiemmista todennuksista.



Kuvio 77 F5 BIG-IP GTM listener statistiikka Euroopan kyselyille

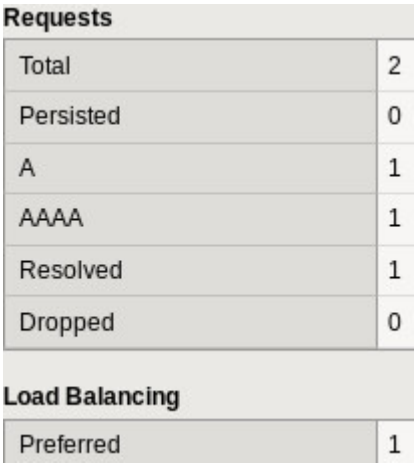
F5 BIG-IP GTM GSLB-statistiikkaa saa välilehdeltä:

DNS-> GSLB-> Wide IP-> Statistics

Kyseiseltä välilehdeltä aukeaa kuvion 78 kaltainen näkymä. Kuten aiemmissakin todennuksissa, pyyntöjä on saapunut kaksi, joista toinen on selvitetty.

Kuvio 78 F5 BIG-IP GTM Wide-IP statistiikkaa Euroopan kyselyille

Kuten Amerikan kyselyissä, vain IPv4-pyyntöön on vastattu ja kyseiselle vastaukselle on toteutettu kuormantasausprosessi. Kuvio 79 on täysin samanlainen kuin Amerikan vastaava kuvio 61.



Kuvio 79 F5 BIG-IP GTM Wide-IP statistiikkaa Details välilehdeltä

Kuviossa 80 on Pools tason tilastoja, jotka saa näkyviin välilehdiltä:

DNS-> GSLB-> Pools-> Statistics

Kuten aiemmissa tapauksissa, ensimmäinen kuormantasaus tehdään Pools tasolla. Tässä tapauksessa kuormantasaus on osunut kohteeseen ”Camibo_Europa_Pool”, joka on täysin oikea valinta, sillä kysely saapuu Euroopan IP-osoitealueelta.

Kuvio 80 F5 BIG-IP GTM Palvelin statistiikkaa Euroopan kyselyille

Kuormantasauksessa on edetty hierarkiassa alaspäin. Ja kuten aiemmin, järjestelmiä on vain yksi jokaisen Poolin alla, joten valinta on helppo. Tässä tapauksessa on valittu ”Camibo_Europa_Web”. Kuvion 81 statistiikkanäkymään pääsee välilehdiltä:

DNS-> GSLB-> Servers-> Statistics

Kuvio 81 F5 BIG-IP GTM Palvelin statistiikkaa Euroopan kyselyille

6 Pohdinta

6.1 Yleistä

Työssä saavutettin sille asetetut tavoitteet jotakuinkin. Pystytetyt järjestelmät vastaavat melkolailla tämänhetkisiä ”mainstream” ratkaisuja toteuttaa kuormantasausta CDN:lle. Ratkaisuissa käyttäjän geolokaatio määriteltiin perustuen käyttäjän käyttämään DNS-palvelimeen. Tämä ratkaisu on sinänsä toimiva, mutta tapauksissa, joissa käyttäjä on itse määrittänyt oman DNS-palvelimensa, geolokaatio saattaa mennä pahastikin pieleen. Tällä hetkellä tähän ongelmaan ei ole saatavilla DNS-pohjaista ratkaisua, joskin kehitteillä saattaa olla muutoksia itse DNS-protokollaan, jolloin DNS-kyselyihin lisättäisiin osa tai koko käyttäjän IP. Esimerkiksi googlella ja OpenDNS:llä on omat ratkaisunsa, joista lisäinfoa löytää mm. hakusanoin "The Global Internet Speedup".

6.2 F5 jatkokehitys

F5 BIG-IP GTM-järjestelmää ei ollut mahdollista ottaa täydellisesti käyttöön johtuen lisenssisyistä. GTM-järjestelmällä pystyy nykyisen konfiguraation turvin tekemään DNS-pohjaista reititystä tai kuormantasausta, mutta minkäänlaisia ”healthcheckejä” ei nykyisellään pysty tekemään. GTM-järjestelmä vaatisi jokaiseen ”datacenter”-osioon oman F5 BIG-IP LTM-järjestelmän, mutta lisensointisyistä tätä en pystynyt toteuttamaan. GTM käyttää big3d-agentteja ”healthcheck” arvioihinsa. Nämä big3d-agentit ovat F5 yksityisomistukseisia ohjelmia, jotka toimivat F5 LTM-järjestelmien päällä. Tästä johtuen vertailua järjestelmien kesken oli huonohkoa suorittaa.

6.3 Windows Server CDN-palvelimet

Työn loputtua taustalla on täysin toimiva CDN-järjestelmä. Kyseistä järjestelmää voisi tosin täydentää ja jatkokehittää mm. lisäämällä välityspalvelimia, että suuremman määrän välimuistia. Alkupäässä työtä rajattiin myös siten, että painopistettä poistettiin itse datakeskusten sisällä tapahtuvasta liikennöinnistä ja keskitytään itse globaaliin kuormantasaukseen.

Proxy-palvelimien avulla voitaisiin mm. reitittää asiakkaan liikennettä vielä datakeskuksen sisäisesti eri kuormantasausmetodein tai perustuen käyttäjän selaimen omi-

naisuuksiin. Esimerkiksi käyttäjät, jotka saapuvat Euroopasta, joiden selaimena on Mozilla Firefox ja lisäosana .net framework voitaisiin reitittää eri palvelimelle kuin käyttäjät, joiden selaimena on Internet Explorer. Vaihtoehtoja on monia. Yksi vaihtoehto olisi myös lisätä palvelimia edgen taakse ja suorittaa kuormantasausta keskusten sisällä perustuen tiettyihin parametreihin ja tehdä pysyvyyteen perustuvia ratkaisuja (cookies, url rewrite).

6.4 Bind

Bind on ominaisuuksiltaan nykyisellään hyvin yksinkertainen ja vaatisi suuremmassa mittakaavassa ominaisuuksia, jotta pystyisi kilpailemaan suurempien valmistajien kanssa. Tärkein ja räikein ominaisuus, jonka Bind tarvitsisi on terveystarkastukset (health checks), joilla reunapalvelimia pystytään tutkimaan ovatko ne pystyssä. Toinen mahdollinen lisäys on ”edns-client-subnet”-kokeellinen patchi bindille. Kyseinen patchi lisää mahdollisuuden sisällyttää osan käyttäjän IP-osoitteesta DNS-kyselyihin. Tämä tosin vaatisi jonkin asteista räätälöintiä ohjelmakoodin osalta, sillä IP-osoitealue patchi täytyisi saada yhteensopivaksi ”edns-client-subnet”-patchin kanssa.

Lähteet

Aitchison, R. 2011. Pro DNS and BIND

Aulds, C. 2001. Linux Apache Web Server Administration, Second Edition

BIG-IP_GTMC. BIG-IP Global Traffic Manager: Concepts version 11.5

BIG-IP_GTMLB. BIG-IP Global Traffic Manager: Load Balancing version 11.5

Bind. Viitattu 05.05.2014. http://www.isc.org/downloads/bind/

Easttom, C., Palladino, S. 2012. Essential Linux Administration: A Comprehensive Guide for Beginners

Edgecast. Viitattu 07.05.2014. http://www.edgecast.com/

Edgecastmap. Viitattu 07.05.2014. http://www.edgecast.com/network/map/

F5 DNS-implementation. BIG-IP dns services: implementations version 11.5

Held, G. 2010. A Practical Guide to Content Delivery Networks, Second Edition

IIS.net. Nettisivustolta IIS.net viitattu 31.03.2014
http://www.iis.net/downloads/microsoft/application-request-routing

Introduction to IIS Architectures. 2007. nettisivustolta IIS.net viitattu 05.05.2014.
http://www.iis.net/learn/get-started/introduction-to-iis/introduction-to-iis-architecture

Jyväskylä Security Technology viitattu 10.02.2014. http://jyvsectec.fi/

Li, J. On peer-to-peer (P2P) content delivery. Viitattu 04.02.2014. http://www.land.ufrj.br/~classes/coppe-redes-2008/biblio/P2P-content-delivery.pdf

owasp-cookie. Viitattu 17.02.2014.

https://www.owasp.org/images/5/5a/TestingGuide-LogoutTest-fig1.png

Schaefer K. 2008. Professional IIS 7.0

Stutzbach, D., Zappala, D., Rejaie, R., The Scalability of Swarming Peer-to-Peer Content Delivery. Viitattu 04.02.2014. http://ix.cs.uoregon.edu/~reza/PUB/networking05.pdf

Tutustu JAMKiin. Viitattu 10.02.2014. http://www.jamk.fi/fi/Tietoa-JAMKista/Tutustu-JAMKiin/

Verma, D. 2002. Content Distribution Networks: An Engineering Approach

What is CDN?. Nettisivustolta CDN-Advisor viitattu 04.02.2014. http://www.cdn-advisor.com/what-is-cdn/

Liitteet

Liite 1. Bind9 named.conf

```
include "/usr/local/bind/etc/rndc.key";

controls {
        inet 127.0.0.1 allow { localhost; } keys { "rndc-key"; };
};

options {
        directory       "/var/named";
        pid-file        "/var/named/named.pid";
        dump-file       "data/cache_dump.db";
        statistics-file "data/named_stats.txt";
                        allow-transfer {"any";};
};

logging {

        channel default_debug {
                        file "data/named.run";
                        severity dynamic;
        };
};

view "north_america" {
        match-clients { country_AR; country_BR; country_CL; country_EC;
country_US; country_CA; country_MX; };
        recursion no;
```

```
        zone "camibo.com" {
                type master;
                file "/var/named/camibo-us.com.db";
        };
};

view "asia" {
        match-clients { country_IN; country_CN; country_KR; country_HK;
country_PH; country_SG; country_TW; country_TH; country_JP; };
        recursion no;
        zone "camibo.com" {
                type master;
                file "/var/named/camibo-asia.com.db";
        };
};

view "other" {
        match-clients { any; };
        recursion no;
        zone "camibo.com" {
                type master;
                file "/var/named/camibo.com.db";
        };
};
```

Liite 2. Camibo-us.com.db

```
;
$TTL    10
camibo.com. 86400   IN      SOA     ns1.camibo.com.
        mail.camibo.com.    (
                5                   ;serial number
```

```
                    86400               ;refresh
                    10                  ;retry
                    3600000             ;expire
                    10                  )
ns1.camibo.com.             IN      A       81.52.190.150
camibo.com.         IN      NS      ns1.camibo.com.
www.camibo.com.             IN      A       67.201.66.153
```

Liite 3. Camibo-asia.com.db

```
;
$TTL        10
camibo.com. 86400       IN      SOA     ns1.camibo.com.
            mail.camibo.com.    (
                    4               ;serial number
                    10              ;refresh
                    7200            ;retry
                    3600000         ;expire
                    10              )
www.camibo.com.     300     IN      A       192.88.171.186
ns1.camibo.com.             IN      A       81.52.190.150
camibo.com.         IN      NS      ns1.camibo.com.
```

Liite 4. Camibo.com.db

```
;
$TTL        10
camibo.com. 86400       IN      SOA     ns1.camibo.com.
            mail.camibo.com.    (
                    4               ;serial number
                    10              ;refresh
```

	7200		;retry	
	3600000		;expire	
	10		)	
www.camibo.com.	300	IN	A	81.52.190.117
ns1.camibo.com.		IN	A	81.52.190.150
camibo.com.	IN	NS	ns1.camibo.com.	

Liite 5. F5 konfiguraatio (SCF)

```
#TMSH-VERSION: 11.5.0

cli admin-partitions {
   update-partition Common
}
apm client-packaging /Common/client-packaging { }
apm resource remote-desktop citrix-client-bundle /Common/default-citrix-client-
bundle { }
auth user admin {
   description "Admin User"
   encrypted-password "$1$salt$IEd.dPRrJY41NYnnaBENQ1"
   partition-access all
   role admin
   shell tmsh
}
auth user root {
   description none
   encrypted-password "$1$tAiUgKE3$t27ER2uyRzoplfh91./94."
   shell bash
}
cm cert /Common/dtca-bundle.crt {
```

```
    cache-path
/config/filestore/files_d/Common_d/trust_certificate_d/:Common:dtca-
bundle.crt_32406_1
    checksum SHA1:1289:3bf71d1956c775801a971d7bb568df286dc8fcfa
    revision 1
}
cm cert /Common/dtca.crt {
    cache-path
/config/filestore/files_d/Common_d/trust_certificate_d/:Common:dtca.crt_32402_1
    checksum SHA1:1289:3bf71d1956c775801a971d7bb568df286dc8fcfa
    revision 1
}
cm cert /Common/dtdi.crt {
    cache-path
/config/filestore/files_d/Common_d/trust_certificate_d/:Common:dtdi.crt_32398_1
    checksum SHA1:1220:137c6b9e2689c828dd64ef1995f4f33645f849d4
    revision 1
}
cm device /Common/bigip1 {
    active-modules { "BIG-IP, LAB (LTM,APM,ASM,AM, GTM), VE|ISQYRWS-
WHZFQUU|IPV6 Gateway|Rate Shaping|Ram Cache|50 MBPS COMPRESSION|SSL,
500 TPS Per Core|Client Authentication|ASM, VE|AFM, VE|DNS Services|WBA,
VE|PSM, VE|Acceleration Manager, VE|WOM, VE|DNSSEC|Anti-Virus Checks|Base
Endpoint Security Checks|Firewall Checks|Network Access|Secure Virtual Key-
board|APM, Web Application|Machine Certificate Checks|Protected Work-
space|Max Compression, VE|Remote Desktop|App Tunnel|SSL, Max TPS,
VE|Routing Bundle, VE|DNS Rate Fallback, 50|GTM Rate Fallback, 8|DNS Licensed
Objects, 0|GTM Rate, 8|DNS Rate Limit, 50 QPS|GTM Licensed Objects, 0|DNS Rate
Fallback, Unlimited|DNS Licensed Objects, Unlimited|DNS Rate Limit, Unlimited
QPS|AAM, Upgrade from WAM, (v11.4 & later)" "DNSSEC|YEEPPGW-OUCBVOV"
"GTM, VE|OSPVMMT-TITFFCP|IPV6 Gateway|Ram Cache|STP|DNS Express|Routing
Bundle, VE|GTM Licensed Objects, Unlimited|DNS Rate Fallback, Unlimited|DNS Li-
```

```
censed Objects, Unlimited|GTM Rate Fallback, (UNLIMITED)|DNS Rate Limit, Unlim-
ited QPS|GTM Rate, Unlimited" }
    base-mac 00:50:56:84:6f:6b
    build 0.0.221
    cert /Common/dtdi.crt
    chassis-id 420430d2-ef16-d19c-4fee8e2e9ec2
    edition Final
    hostname bigip1
    key /Common/dtdi.key
    management-ip 192.168.1.100
    marketing-name "BIG-IP Virtual Edition"
    optional-modules { "Advanced LTM Protocols" "App Mode (TMSH Only, No
Root/Bash)" "Carrier Grade NAT, BIG-IP" "External Interface and Network HSM" "Ex-
ternal Interface and Network HSM, VE" "SDN Services, VE" "SSL, Forward Proxy" "SSL,
Forward Proxy, VE" "SWG Subscription, 1Yr, VE" "SWG Subscription, 3Yr, VE" "URL
Filtering Subscription, 1Yr, VE" "URL Filtering Subscription, 3Yr, VE" }
    platform-id Z100
    product BIG-IP
    self-device true
    time-zone PDT
    version 11.5.0
}
cm device-group /Common/device_trust_group {
    auto-sync enabled
    devices {
        /Common/bigip1 { }
    }
    hidden true
    network-failover disabled
}
cm device-group /Common/gtm {
    devices {
```

```
      /Common/bigip1 { }
   }
   hidden true
   network-failover disabled
}
cm key /Common/dtca.key {
   cache-path
/config/filestore/files_d/Common_d/trust_certificate_key_d/:Common:dtca.key_324
04_1
   checksum SHA1:1704:d8c0a93bb4cf569831c66aae820510961f94c415
   revision 1
}
cm key /Common/dtdi.key {
   cache-path
/config/filestore/files_d/Common_d/trust_certificate_key_d/:Common:dtdi.key_324
00_1
   checksum SHA1:1704:558c2cb667fe293d365b6a214a4893b9eb4fecb5
   revision 1
}
cm traffic-group /Common/traffic-group-1 {
   unit-id 1
}
cm traffic-group /Common/traffic-group-local-only { }
cm trust-domain /Common/Root {
   ca-cert /Common/dtca.crt
   ca-cert-bundle /Common/dtca-bundle.crt
   ca-devices { /Common/bigip1 }
   ca-key /Common/dtca.key
   guid 47532604-31d5-4f16-b1e1005056846f6b
   status standalone
   trust-group /Common/device_trust_group
}
```

```
gtm datacenter /Common/Camibo_Asia {
   description "Camibo Oy:n Aasian palvelinkeskus"
}
gtm datacenter /Common/Camibo_Europa {
   description "Camibo Oy:n euroopan palvelinsali"
}
gtm datacenter /Common/Camibo_USA {
   description "Camibo Oy:n yhdysvaltain palvelin"
}
gtm pool /Common/Camibo_Asia_Pool {
   fallback-mode none
   members {
      /Common/Camibo_Asia_Web:Camibo_Asia_Web {
         order 0
      }
   }
}
gtm pool /Common/Camibo_Europa_Pool {
   fallback-mode none
   members {
      /Common/Camibo_Europa_Web:Camibo_Europa_Web {
         order 0
      }
   }
}
gtm pool /Common/Camibo_USA_Pool {
   fallback-mode none
   members {
      /Common/Camibo_USA_Web:Camibo_USA_Web {
         order 0
      }
   }
```

```
}
gtm pool /Common/Camipool {
   fallback-mode none
   load-balancing-mode topology
}
gtm region /Common/America {
   region-members {
      continent NA { }
      continent SA { }
   }
}
gtm region /Common/Asia {
   region-members {
      continent AS { }
      continent OC { }
   }
}
gtm region /Common/Europa {
   region-members {
      continent -- { }
      continent AF { }
      continent AN { }
      continent EU { }
   }
}
gtm server /Common/Camibo_Asia_Web {
   addresses {
      192.88.171.186 {
         device-name Camibo_Asia_Web
      }
   }
   datacenter /Common/Camibo_Asia
```

```
    product generic-host
    virtual-servers {
        Camibo_Asia_Web {
            destination 192.88.171.186:80
        }
    }
}
gtm server /Common/Camibo_Europa_Web {
    addresses {
        81.52.190.117 {
            device-name Camibo_Europa_Web
        }
    }
    datacenter /Common/Camibo_Europa
    product generic-host
    virtual-servers {
        Camibo_Europa_Web {
            destination 81.52.190.117:80
        }
    }
}
gtm server /Common/Camibo_USA_Web {
    addresses {
        67.201.66.153 {
            device-name Camibo_USA_Web
        }
    }
    datacenter /Common/Camibo_USA
    product generic-host
    virtual-servers {
        Camibo_USA_Web {
            destination 67.201.66.153:80
```

```
      }
   }
}
gtm topology  ldns: region /Common/Asia  server: pool /Common/Camibo_Asia_Pool
{
   order 4
   score 30
}
gtm topology  ldns: region /Common/Europa  server: pool
/Common/Camibo_Europa_Pool {
   order 5
   score 20
}
gtm topology  ldns: region /Common/America  server: pool
/Common/Camibo_USA_Pool {
   order 6
   score 10
}
gtm wideip /Common/www.camibo.com {
   pool-lb-mode topology
   pools {
      /Common/Camibo_Asia_Pool {
         order 0
      }
      /Common/Camibo_Europa_Pool {
         order 1
      }
      /Common/Camibo_USA_Pool {
         order 2
      }
   }
}
```

```
gtm global-settings general {
   send-wildcard-rrs enabled
}
gtm global-settings load-balancing {
   verify-vs-availability no
}
gtm global-settings metrics {
   metrics-collection-protocols { icmp }
}
gtm global-settings metrics-exclusions {
   addresses none
}
gtm monitor http /Common/HTTP_monitori {
   defaults-from /Common/http
   destination *:*
   interval 30
   probe-timeout 5
   send "GET /index.html HTTP/1.1"
   timeout 120
}
ltm default-node-monitor {
   rule none
}
ltm node /Common/67.201.66.153 {
   address 67.201.66.153
}
ltm node /Common/81.52.190.117 {
   address 81.52.190.117
}
ltm node /Common/192.88.171.186 {
   address 192.88.171.186
}
```

```
ltm virtual /Common/Camibo_listener2 {
    destination /Common/81.52.190.150:53
    ip-protocol udp
    mask 255.255.255.255
    profiles {
        /Common/dns { }
        /Common/udp_gtm_dns { }
    }
    source 0.0.0.0/0
    translate-address disabled
    translate-port disabled
}
ltm virtual /Common/camibo_listener {
    destination /Common/81.52.190.151:53
    disabled
    ip-protocol udp
    mask 255.255.255.255
    profiles {
        /Common/dns { }
        /Common/udp_gtm_dns { }
    }
    source 0.0.0.0/0
    translate-address disabled
    translate-port disabled
}
ltm virtual-address /Common/81.52.190.150 {
    address 81.52.190.150
    arp enabled
    icmp-echo enabled
    mask 255.255.255.255
    traffic-group /Common/traffic-group-local-only
}
```

```
ltm virtual-address /Common/81.52.190.151 {
    address 81.52.190.151
    arp enabled
    icmp-echo enabled
    mask 255.255.255.255
    traffic-group /Common/traffic-group-local-only
}
ltm dns nameserver /Common/France1 {
    route-domain /Common/0
}
ltm dns zone /Common/Basiczone {
    dns-express-server /Common/France1
    transfer-clients {
        /Common/France1
    }
}
ltm dns cache resolver /Common/camibo.com.resolver.2 {
    route-domain /Common/0
}
ltm monitor dns /Common/DNS_monitori {
    accept-rcode no-error
    answer-contains query-type
    defaults-from /Common/dns
    destination *:*
    interval 5
    qname www.camibo.com
    qtype a
    time-until-up 0
    timeout 16
}
net dns-resolver /Common/France1 {
    route-domain /Common/0
```

```
}
net interface 1.1 {
   media-fixed 10000T-FD
}
net interface 1.2 {
   media-fixed 10000T-FD
}
net interface 1.3 {
   media-fixed 10000T-FD
}
net interface 1.4 {
   media-fixed 10000T-FD
}
net interface 1.5 {
   media-fixed 10000T-FD
}
net interface 1.6 {
   media-fixed 10000T-FD
}
net route /Common/France1 {
   gw 81.52.190.1
   network default
}
net route-domain /Common/0 {
   id 0
   vlans {
      /Common/http-tunnel
      /Common/socks-tunnel
      /Common/France1
   }
}
net self /Common/France1 {
```

```
    address 81.52.190.150/24
    traffic-group /Common/traffic-group-local-only
    vlan /Common/France1
}
net self-allow {
    defaults {
        ospf:0
        tcp:161
        tcp:22
        tcp:4353
        tcp:443
        tcp:53
        udp:1026
        udp:161
        udp:4353
        udp:520
        udp:53
    }
}
net stp /Common/cist {
    interfaces {
        1.1 {
            external-path-cost 2000
            internal-path-cost 2000
        }
    }
    vlans {
        /Common/France1
    }
}
net vlan /Common/France1 {
    interfaces {
```

```
      1.1 { }
    }
    tag 1
}
net fdb tunnel /Common/http-tunnel { }
net fdb tunnel /Common/socks-tunnel { }
net fdb vlan /Common/France1 { }
net ipsec ike-daemon /Common/ikedaemon { }
net tunnels tunnel /Common/http-tunnel {
    description "Tunnel for http-explicit profile"
    profile /Common/tcp-forward
}
net tunnels tunnel /Common/socks-tunnel {
    description "Tunnel for socks profile"
    profile /Common/tcp-forward
}
security firewall port-list /Common/_sys_self_allow_tcp_defaults {
    ports {
      22 { }
      53 { }
      161 { }
      443 { }
      1029-1043 { }
      4353 { }
    }
}
security firewall port-list /Common/_sys_self_allow_udp_defaults {
    ports {
      53 { }
      161 { }
      520 { }
      1026 { }
```

```
        4353 { }
    }
}
security firewall rule-list /Common/_sys_self_allow_all {
    rules {
        _sys_allow_all {
            action accept
        }
    }
}
security firewall rule-list /Common/_sys_self_allow_defaults {
    rules {
        _sys_allow_tcp_defaults {
            action accept
            ip-protocol tcp
            destination {
                port-lists {
                    /Common/_sys_self_allow_tcp_defaults
                }
            }
        }
        _sys_allow_udp_defaults {
            action accept
            ip-protocol udp
            destination {
                port-lists {
                    /Common/_sys_self_allow_udp_defaults
                }
            }
        }
        _sys_allow_ospf_defaults {
            action accept
```

```
            ip-protocol ospf
        }
    }
}
security firewall rule-list /Common/_sys_self_allow_management {
    rules {
        _sys_allow_ssh {
            action accept
            ip-protocol tcp
            destination {
                ports {
                    22 { }
                }
            }
        }
        _sys_allow_web {
            action accept
            ip-protocol tcp
            destination {
                ports {
                    443 { }
                }
            }
        }
    }
}
security ip-intelligence policy /Common/ip-intelligence { }
sys db avr.subnetprefixlength.ipv4 {
    value "24"
}
sys db avr.subnetprefixlength.ipv6 {
    value "64"
```

```
}
sys db dhclient.mgmt {
   value "disable"
}
sys db gtm.peerinfototalgtms {
   value "0"
}
sys db provision.extramb {
   value "0"
}
sys db provision.tomcat.extramb {
   value "0"
}
sys db rule.validation {
   value "strict"
}
sys db tm.allowmulticastl2destinationtraffic {
   value "disable"
}
sys db tm.tcpallowinsecurerst {
   value "disable"
}
sys db tmm.classallocatemetadata {
   value "enable"
}
sys db tmm.coredump {
   value "enable"
}
sys db tmm.gradualfileloadadjust {
   value "enable"
}
sys db tmm.lb.wlcoffset {
```

```
    value "disable"
}
sys db tmm.verbose {
    value "disable"
}
sys db tmm.verbosecmp {
    value "disable"
}
sys dns {
    description configured-by-dhcp
    name-servers { 195.148.26.4 195.148.26.8 }
}
sys feature-module cgnat {
    disabled
}
sys folder / {
    device-group none
    hidden false
    inherited-devicegroup false
    inherited-traffic-group false
    traffic-group /Common/traffic-group-1
}
sys folder /Common {
    device-group none
    hidden false
    inherited-devicegroup true
    inherited-traffic-group true
    traffic-group /Common/traffic-group-1
}
sys global-settings {
    gui-setup disabled
    mgmt-dhcp disabled
```

```
}
sys management-dhcp /Common/sys-mgmt-dhcp-config {
   request-options { subnet-mask broadcast-address routers domain-name domain-
name-servers host-name ntp-servers }
}
sys management-ip 192.168.1.100/24 {
   description configured-statically
}
sys management-route /Common/default {
   description configured-statically
   gateway 192.168.1.1
   network default
}
sys provision gtm {
   level nominal
}
sys provision ltm {
   level nominal
}
sys snmp {
   agent-addresses { tcp6:161 udp6:161 }
   communities {
      /Common/comm-public {
         community-name public
         source default
      }
   }
   disk-monitors {
      /Common/root {
         minspace 2000
         path /
      }
```

```
        /Common/var {
            minspace 10000
            path /var
        }
    }
    process-monitors {
        /Common/bigd {
            process bigd
        }
        /Common/chmand {
            process chmand
        }
        /Common/httpd {
            max-processes infinity
            process httpd
        }
        /Common/mcpd {
            process mcpd
        }
        /Common/sod {
            process sod
        }
        /Common/tmm {
            max-processes infinity
            process tmm
        }
    }
}
sys sflow global-settings http { }
sys sflow global-settings vlan { }
sys software update {
    auto-check enabled
```

```
    frequency weekly
}
wom endpoint-discovery { }
```

Liite 6. edns-client-subnet

http://www.afasterinternet.com/

http://tools.ietf.org/html/draft-vandergaast-edns-client-subnet-02

Liite 7. GeoIP

https://code.google.com/p/bind-geoip/