Max Serlo

Kuormantasausjärjestelmän toiminta

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikan koulutusohjelma

Insinöörityö

17.5.2018

Metropolia

Tekijä Otsikko Sivumäärä Aika	Max Serlo Kuormantasausjärjestelmän toiminta 33 sivua 17.5.2018
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tietotekniikka
Ammatillinen pääaine	Tietoverkot ja tietoliikenne
Ohjaajat	Lehtori Marko Uusitalo

Insinöörityön tarkoituksena oli implementoida kuormantasausjärjestelmä sekä perehtyä palvelinympäristössä tehtävään kuormantasaukseen ja sen avulla saavutettaviin hyötyihin. Työssä tutkittiin kuormantasauksessa käytettävää arkkitehtuuria sekä kuormantasausalgoritmeja ja niiden soveltuvuutta eri käyttötarkoituksiin.

Työssä tutustuttiin Citrix Systems, Inc -laitevalmistajan kuormantasauslaitteiden ominaisuuksiin ja niiden käyttötarkoituksiin. Käytännön osuudessa implementoitiin Citrix NetScaler VPX -kuormantasausjärjestelmä sekä perehdyttiin syvällisemmin järjestelmän toimintaan.

NetScaler VPX:n asennus toteutettiin virtualisoidusti käyttäen Microsoft Hyper-V -virtualisointialustaa. Lisenssinä käytettiin NetScaler Platinum -testilisenssiä, jonka avulla laitteen kaikki ominaisuudet olivat käytettävissä. NetScaler VPX -kuormantasausjärjestelmän käyttöönotossa käytettiin valmista virtuaalikonetta, joka liitettiin virtualisointialustaan.

Järjestelmän konfigurointi aloitettiin perusasetuksista, jonka jälkeen määriteltiin web-palvelimen kuormantasaus. Kuormantasaukseen lisättiin sisällönsuodatus sekä erilaisia liikenteen optimointiominaisuuksia. Kaikki konfiguroinnit pyrittiin tekemään ensisijaisesti komentokehotteen avulla. Määrittelyt testattiin lisäksi käyttäen web-käyttöliittymää.

Lopputuloksena kuormantasausjärjestelmän käyttöönotto toteutui onnistuneesti, ja järjestelmän ominaisuudet saatiin määriteltyä suunnitelman mukaisesti.

Avainsanat	Kuormantasaus, kuormanjako, kuormantasausjärjestelmä, NetScaler

Author Title Number of Pages Date	Max Serlo Operation of the Load Balancing System 33 pages 17 May 2018
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Professional Major	Information and Communications Technology
Instructors	Marko Uusitalo Senior Lecturer

The purpose of this thesis was to implement a load balancing system and to examine how load balancing works, and what kind of benefit it provides in server environment. In the beginning of the thesis, load balancing architecture and different kind of load balancing algorithms and their suitability for different uses was examined.

Load balancing product's details of the Citrix Systems, Inc manufacturer were introduced. In the practical part of the thesis Citrix NetScaler VPX load balancer was implemented and the features of the device were further examined.

NetScaler VPX installation was done virtually using Microsoft Hyper-V hypervisor. The device was activated with NetScaler Platinum license, which provides all system features. NetScaler virtual appliance initialization was performed with virtual machine.

System configuration was started with basic setup. After defining basic settings, the load balancing feature and configuration for the web server was done. Content switching feature was added for the load balancing setup and different kind of optimization was also added to the system. Command-line was preferred when configuration was performed. All configurations were also tested with web user interface.

The result of the load balancer implementation was done successfully, and the system features were configured as planned.

Keywords	Load balancer, Load balancing, Load balancing system, NetScaler

Sisällys

Lyhenteet

1 Johdanto 1

2 Kuormantasaus 2

2.1 Toiminta palvelinympäristössä 2
2.2 Kuormantasaus OSI-mallin kerroksissa 4
2.3 Layer 4 kuormantasaus 5
2.3.1 NAT mode 5
2.3.2 DSR mode (Direct Server Return) 6
2.3.3 Tunnel mode 7
2.4 Layer 7 kuormantasaus 8
2.4.1 Proxy mode 8
2.4.2 Transparent proxy mode 9
2.5 Algoritmit 9
2.5.1 Round Robin 10
2.5.2 Weighted Round Robin 10
2.5.3 Least connection 10
2.5.4 Fastest connection 11
2.5.5 Hash-algoritmit 11

3 Citrix NetScaler ADC 11

3.1 Fyysiset alustat NetScaler MPX & SDX 12
3.2 Virtualisoidut alustat VPX & CPX 13
3.3 NetScaler verkkotopologia 14

4 NetScaler VPX ominaisuudet ja implementointi 16

4.1 Virtualisointi ja alkuasetukset 17
4.2 Web-käyttöliittymä 20
4.3 Web-palvelimen kuormantasaus 21
4.4 Content switching 23
4.5 SSL-salauksen käyttö 25
4.6 HTTP-kompressointi 25
4.7 Integrated Caching 26
4.8 High Availability 28

5 Yhteenveto 29

Lähteet 30

Lyhenteet

ADC	Application Delivery Controller. Kuormantasausjärjestelmä, jossa on kattavat kuormantasaus-, liikenteen optimointi- ja tietoturvaominaisuudet.
ARP	Address Resolution Protocol. Tietoliikenneprotokolla, jonka tehtävä on muuntaa IP-osoitteet fyysisiksi osoitteiksi.
CS	Content Switching. Kuormantasausominaisuus, jonka avulla voidaan esittää erilaista web-sivuston sisältöä eri käyttäjille.
CSV	Comma-separated values. Tiedostomuoto, jolla tallennetaan taulukkomuotoista dataa tekstitiedostoon.
DNS	Domain Name System. Nimipalvelujärjestelmä, jonka tehtävänä on muuntaa verkkotunnukset IP-osoitteiksi.
DSR	Direct Server Return. Kuormantasaustapa, jossa serveriltä tuleva paluuliikenne ei kulje kuormantasaajan läpi.
HA	High Availability. Korkea saatavuus
HTTP	Hypertext Transfer Protocol. Tiedonsiirtoprokolla, jota käytetään selainten ja web-palvelimien välisessä tiedonsiirrossa.
IP	Internet Protocol. TCP/IP-mallin verkkokerroksessa toimiva protokolla, joka välittää tietoliikennepaketteja.
LB	Load Balancing. Kuormantasaus.
MAC	Media Access Control. Verkkosovittimen yksilöivä koodi.
NAT	Network Address Translation. Osoitteenmuunnos on tekniikka, jonka avulla säästetään julkisia IP-osoitteita.
OSI	Open Systems Interconnection Reference Model. Tietoliikennearkkitehtuurimalli, OSI-malli koostuu seitsemästä kerroksesta.

SNIP	Subnet IP Address. Aliverkon osoite. Aliverkkojen avulla voidaan parantaa verkon suorituskykyä ja hallintaa.
SPOF	Single point of failure. Yksittäinen vikaantumispiste.
SSH	Secure Shell. Salattu tietoliikenneprotokolla, jonka yleisin käyttötarkoitus on suojatun etäyhteyden muodostaminen laitteiden välille.
SSL	Secure Socket Layer. Salausprotokolla, jolla suojataan tietoliikennettä.
TCP	Transmission Control Protocol. Tietoliikenneprotkolla, jolla luodaan yhteyksiä laitteiden välille.
UDP	User Datagram Protocol. Yhteydetön protokolla, joka ei vaadi yhteyttä laitteiden välille.
VIP	Virtual IP. Virtuaali-IP
Vserver	Virtual Server. Virtuaalipalvelin

1 Johdanto

Verkkopalveluiden tulee jatkuvasti palvella kasvavaa käyttäjämäärää. Pilvipalveluiden yleistymisen myötä palveluiden ja sovellusten käyttö ei enää ole sijainti- tai laiteriippuvaista. Palveluiden jatkuva saatavuus onkin yksi keskeisimmistä asioista nykypäivänä. Usein palvelunkäyttäjän ja palveluntarjoajan välillä tehdään palvelutasosopimuksia. Sopimuksissa yleensä määritellään palveluille tietty saatavuus, jonka alittamisesta voi seurata sanktioita.

Kuormantasaustekniikan avulla voidaan tehostaa palveluiden toimintaa ja toteuttaa vikasietoisia palvelinympäristöjä, joilla taataan palveluiden tehokas ja jatkuva saatavuus. Kasvavat tietoturvauhat asettavat omat vaatimuksensa nykypäivän palveluille. Modernit kuormantasausjärjestelmät mahdollistavat suojautumisen nykyaikaisilta sovelluksiin kohdistuvilta hyökkäyksiltä.

Insinöörityössä on tarkoituksena perehtyä kuormantasaajien toimintaan palvelinympäristössä. Alussa perehdytään kuormantasausarkkitehtuuriin sekä tutkitaan kuormantasauksessa käytettäviä tekniikoita ja niiden soveltuvuutta eri käyttötarkoituksiin.

Työssä perehdytään tarkemmin Citrix Systems, Inc. valmistajan tuotteisiin ja niiden teknisiin ominaisuuksiin. Lopuksi työssä implementoidaan Citrix NetScaler VPX Application Delivery Controller, joka on monipuolinen kuormantasausjärjestelmä.

Implementointi toteutetaan virtualisoidusti käyttäen Microsoft Hyper-V -virtualisointialustaa, johon asennetaan NetScaler VPX -virtuaalilaite. Tavoitteena on konfiguroida järjestelmään toimiva kuormantasaus ja perehtyä syvällisemmin NetScaler-kuormantasausjärjestelmän kuormanjaon ja liikenteen optimointiominaisuuksiin.

2 Kuormantasaus

Kuormantasauksen tarkoituksena on jakaa palvelunkäyttäjistä syntyvää liikennettä halutulla tavalla usealle palvelimelle. Kuormantasauksella pyritään optimoimaan resurssien käyttöä, minimoimaan vasteaikaa ja ehkäisemään palvelinresurssien ylikuormittumista. [1.]

Vikasietoisessa korkean saatavuuden ympäristössä yhden palvelimen tai palvelinklusterin vikaantuminen ei aiheuta katkoa palvelussa, koska kuormantasausjärjestelmän jatkuva monitorointi havaitsee vikaantuneet palvelimet ja uudelleenohjaa käyttäjät toimivalle resurssille. [2.]

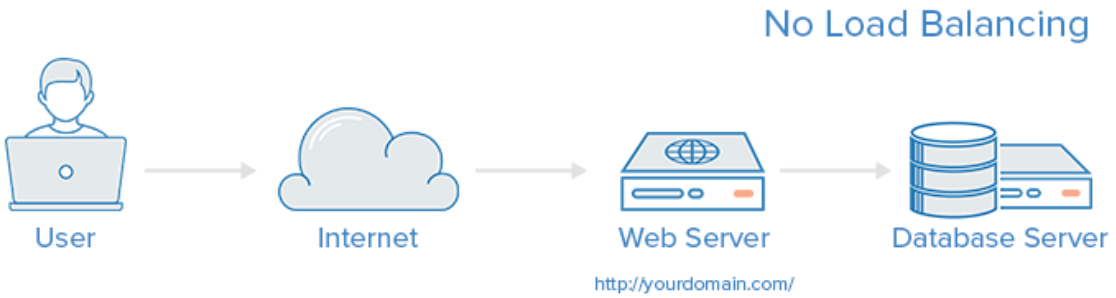
Kuormantasaaja parantaa myös palvelininfrastruktuurin skaalautuvuutta ja tieturvaa. Palvelun kysynnän kasvaessa voidaan lisätä serverikapasiteettia, jolloin kuormantasaaja aloittaa liikenteen välittämisen uusille palvelimille ilman katkoja. Vaihtoehtoisesti kysynnän vähentyessä palvelimia voidaan myös vähentää ilman käyttökatkoa. Nykyaikainen kuormantasausjärjestelmä sisältää myös monia muita ominaisuuksia kuin pelkän kuormanjaon. Järjestelmässä on esimerkiksi mahdollista ottaa käyttöön websovellus palomuuri, jolla voidaan turvata ympäristöä entistä paremmin jatkuvasti kasvavilta tietoturvauhilta. [3;4.]

Oikein toteutetulla palvelin- ja kuormantasausratkaisulla taataan palveluiden jatkuva saatavuus. Palveluiden jatkuvaa saatavuutta voidaan pitää yhtenä tärkeimpänä tekijänä yrityksille, jotka tarjoavat sovellusten käyttöä esimerkiksi pilvipalveluina asiakkaille.

2.1 Toiminta palvelinympäristössä

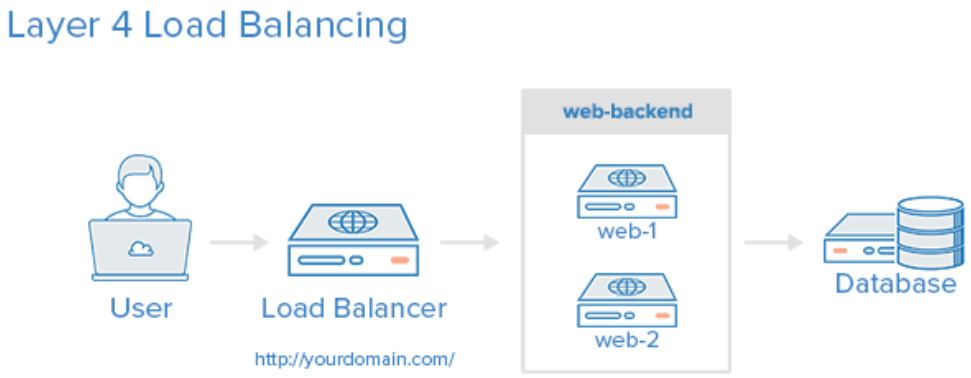
Yksinkertaisessa palvelinympäristössä, jossa ei ole kahdennettu palvelimia, estyy sovelluksen käyttö jo yhden laitteen vikaantuessa. Kuvassa 1 käyttäjä ottaa yhteyttä websovellukseen. Mikäli web-palvelin tai tietokantapalvelin vikaantuu, käyttäjällä ei ole enää yhteyttä käytettävään palveluun [5, s.195].

Toinen merkittävä ongelma yhden web-palvelimen ympäristössä on kuormituksen vaikutus palvelimen resursseihin. Yksittäinen palvelin pystyy hallitsemaan vain rajallisen määrän kuormitusta. Useiden käyttäjien yhdistäessä samanaikaisesti palveluun sen käyttö hidastuu tai estyy jopa kokonaan [2].



Kuva 1. Web-ympäristö ilman kuormantasausta [5].

Lisäämällä kuormantasausjärjestelmä ja kahdentamalla web-palvelimet kuvan 2 mukaisesti saavutetaan jo huomattava parannus palvelinympäristöön. Palvelun käyttäjistä syntyvää liikennettä voidaan tasapainottaa kahdelle web-palvelimille, jolloin palvelu saadaan toimimaan nopeammin ja ympäristön vikasietoisuus parantuu. Mikäli toinen web-palvelin vikaantuu, kuormantasaaja ohjaa kaiken liikenteen toiselle palvelimelle. [2.]



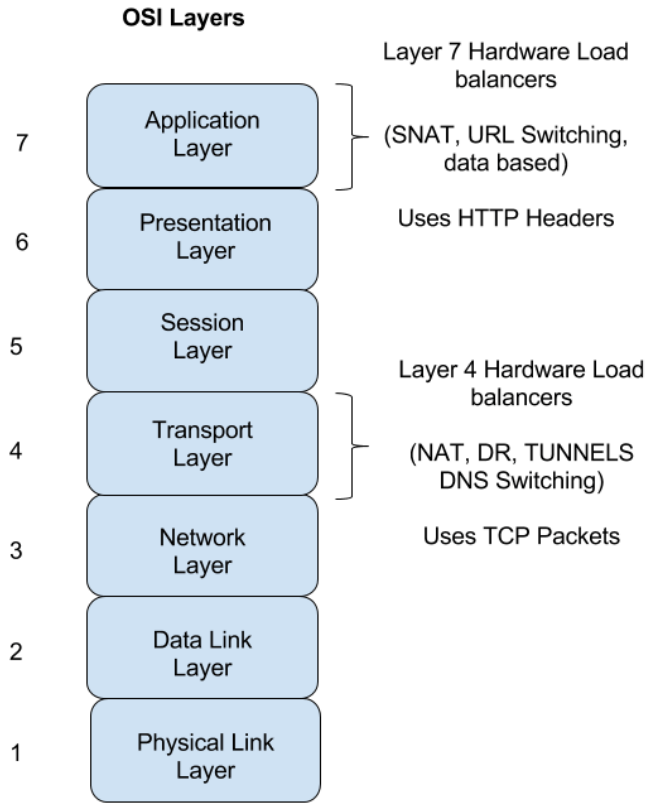
Kuva 2. Kuormantasaus, jossa web-palvelin on kahdennettu. [5.]

Pelkästään web-palvelimen kahdennus ei kuitenkaan tee palvelinympäristöstä vielä täysin vikasietoista. Toteutuksessa on useampi kahdentamaton laite, jonka myötä syntyy SPOF-pisteitä (Single point of failure), jolla tarkoitetaan yksittäistä komponenttia ympäristössä, jonka vikaantuminen aiheuttaa koko ympäristön toimimattomuuden. Mikäli tavoitellaan korkean saatavuuden ympäristöä SPOF-pisteet pitää tunnistaa ja pyrkiä poistamaan. Esitetyssä ympäristössä kuormantasaaja ja tietokantapalvelin tuli myös kahdentaa, jotta ympäristöstä saataisiin korkean saatavuuden ympäristö. [6, s. 195.]

Palvelinympäristöjen suunnittelussa tulee ottaa huomioon, kuinka kriittinen palvelu ja millaiset sopimukset palvelun saatavuudesta on sovittu. Jokaisen laitteen kahdentaminen tuo huomattavia lisäkustannuksia, joten kaikissa palveluissa se ei aina ole järkevää.

2.2 Kuormantasaus OSI-mallin kerroksissa

Kuormantasaus jaotellaan yleisesti kahteen kategoriaan: layer 4 ja layer 7 -kuormantasaukseen, jolla viitataan OSI-mallin kerroksiin. OSI-malli (Open Systems Interconnection Reference Model) on tietoliikenteenarkkitehtuurimalli, joka koostuu seitsemästä kerroksesta. Seitsenkerroksisen mallin ideana on määritellä yhtenäinen tapa toteuttaa tietoliikennejärjestelmiä. OSI-malli kehitettiin ISO:n (International Standardization Organisation) toimesta 1980-luvun alussa. OSI-arkkitehtuurimalli kerrokset on esitetty kuvassa 3. [7.]



Kuva 3. OSI-mallin kerrokset [8.]

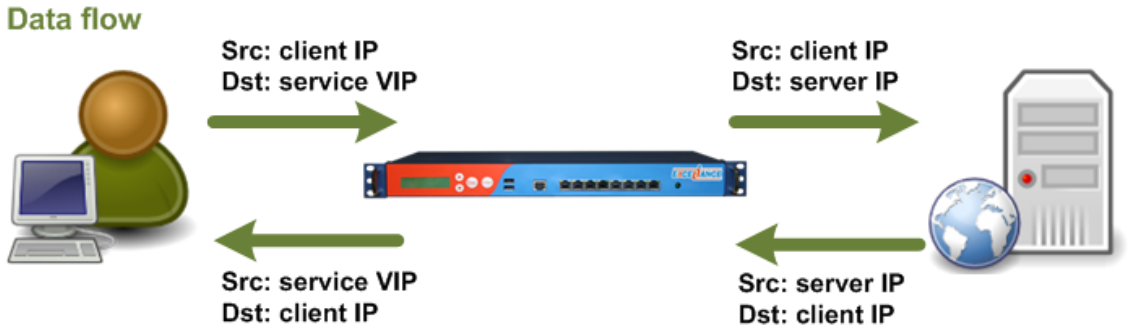
2.3 Layer 4 -kuormantasaus

Layer 4 kuljetuskerroksella kuormantasaaja reitittää liikenteen IP (Internet Protocol), TCP (Transmission Control Protocol) ja UDP (User Datagram Protocol) -protokollien perusteella. Layer 4 -kuormantasauksessa on mahdollista käyttää kolmea eri arkkitehtuuria. NAT mode (Network Address translation), DSR mode (Direct server return) ja IP tunnel mode [9].

2.3.1 NAT mode

Internetliikenteessä tulee kaikilla laiteilla olla uniikki IP-osoite. NAT eli osoitteenmuunnos on tekniikka, jonka tarkoitus on säästää IP-osoitteita sekä parantaa turvallisuutta. Osoitteenmuunnoksen tapoja on useita. Yleisin käyttötarkoitus osoitteenmuutokselle on mahdollistaa usean lähiverkon laitteen käyttää samaa julkista IP-osoitetta [10].

Käytettäessä NAT-toimintoa kuormantasaaja reitittää liikenteen käyttäjän ja palvelimen välillä ja suorittaa kohde IP-osoitteenmuutoksen. Kuvassa 4 asiakas ottaa yhteyttä palvelun julkiseen IP-osoitteeseen. Kuormantasausjärjestelmä tekee osoitteenmuutoksen ja ohjaa paketit palvelimen IP-osoitteeseen. Paluuliikenteessä tehdään myös osoitteenmuutos, jossa palvelimen lähdeosoite muunnetaan palvelun julkiosoitteeksi. [11.]

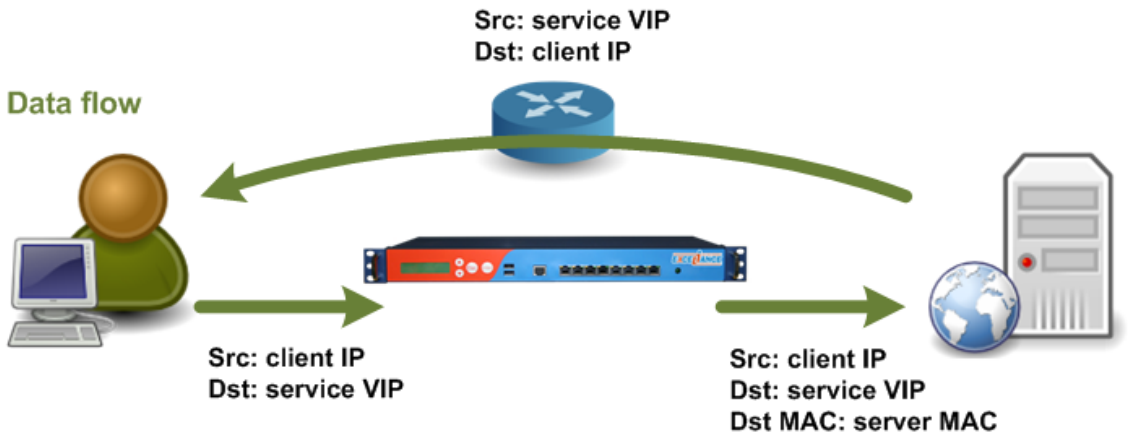


Kuva 4. Layer 4 Nat mode [9.]

Etuina NAT-toteutuksessa on nopea kuormanjako sekä helppo implementointi. Huonoja puolia toteutustavassa on tarve määritellä jokaisen palvelimen default gateway eli oletusyhdyskäytävä käyttämään kuormantasaajaa. Lisäksi kuormantasaajan tiedonsiirtokapasiteetti voi rajoittaa liikennettä, koska kaikki liikennöinti tapahtuu kuormantasaajan kautta. [11.]

2.3.2 DSR mode (Direct Server Return)

DSR-toteutuksessa kuormantasaaja reitittää liikenteen palvelimelle vaihtamalla ainoastaan pakettien kohde MAC-osoitteen. MAC-osoite on laitteen verkkokortin yksilöivä tunnistetieto [12]. DSR mahdollistaa paluuliikenteen suoran yhteyden palvelimen ja asiakkaan välille. Näin paluuliikenteen ei tarvitse kulkea kuormantasaajan läpi. Tällä tavoin ehkäistään ongelmaa, jossa kuormantasaaja tiedonsiirtokapasiteetti toimii pullonkaulana suurissa liikennemäärissä. DSR-toteutuksella saadaan parempi suorituskyky verrattuna NAT-toteutukseen. HTTP-liikenteessä DSR on keskimäärin kahdeksan kertaa nopeampi kuin NAT-toteutus. DSR:n soveltuu parhaiten sovelluksiin, joissa striimataan musiikki- tai videosisältöä. [13.]



Kuva 5. DSR-liikenne [14].

Haittapuolina DSR-toteutuksessa ei voida hyödyntää kuormantasausjärjestelmän kaikkia ominaisuuksia, kuten sovelluksen optimointitekniikoita ja tietoturvatoimintoja, koska paluuliikenne ei kulje kuormantasaajan kautta. Palvelimien tulee vastata oman IP-osoitteen lisäksi, myös määriteltyyn VIP-osoitteeseen (Virtual IP address), jota tarvitaan kuormantasaajalta tulevaan liikenteeseen. Palvelimen ei kuitenkaan tule vastata VIP-osoitteen ARP-kyselyihin (Address Resolution Protocol). [15.]

ARP on protokolla, jonka pääsääntöinen tehtävänä on muuntaa IP-osoitteet MAC-osoitteiksi. Laitteiden tulee selvittä muiden laitteiden fyysiset MAC-osoitteet, jotta liikennöinti onnistuu niiden välillä. ARP-kyselyissä laite lähettää kaikille verkossa oleville laitteille kyselyn ja tiedustelee IP-osoitteen omaava laitetta. Vastaanottajan tunnistaessaan IP-osoitteen se lähettää vastauksena MAC-osoitteen. Kyselyn lähettänyt laite tallentaa MAC-osoitteen ARP-välimuistiin, jolloin sen ei tarvitse tehdä kyselyä uudelleen liikennöidessä saman laitteen kanssa. [16.]

2.3.3 Tunnel mode

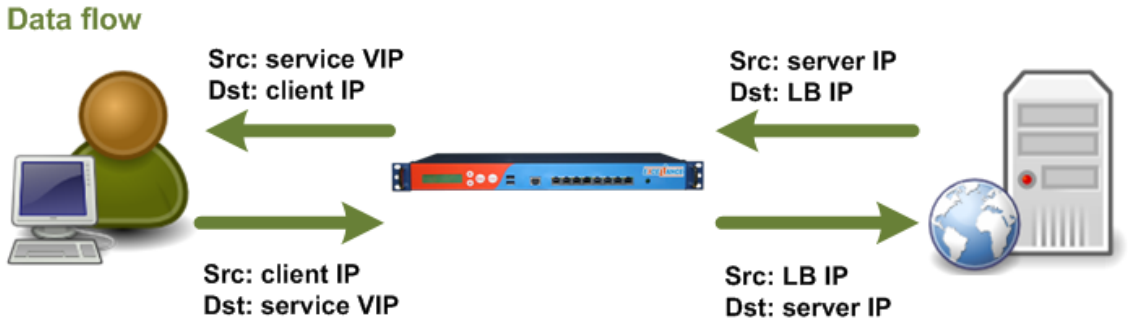
IP-tunnelointia mahdollistaa eri verkossa sijaitsevien palvelimien käytön kuormantasauksessa. Tunneloinnissa kuormantasaaja kapseloi datapaketit käyttäen tunnelointiprotokollaa. Tuetut tunnelointiprotokollat vaihtelevat järjestelmästä riippuen. Tunneloinnin avulla kuormantasaaja voi esimerkiksi käyttää toisen konesalin palvelimia. Liikennöinti kuormantasaajan ja palvelimen välillä tapahtuu vastaavasti kuten DSR-toteutuksessa. [17.]

2.4 Layer 7 -kuormantasaus

Sovelluskerroksessa kuormantasaaja pystyy tekemään kuormantasaus päätöksiä liikenteen sisällön perusteella. Liikennettä voidaan ohjata esimerkiksi HTTP-protokollan otsikkotietojen, lähde IP-osoitteen tai evästeiden avulla [18]. Layer 7 -kuormantasauksessa käytettävät arkkitehtuurit ovat Proxy mode (välityspalvelin) ja Transparent proxy mode (läpinäkyvä välityspalvelin) [9].

2.4.1 Proxy mode

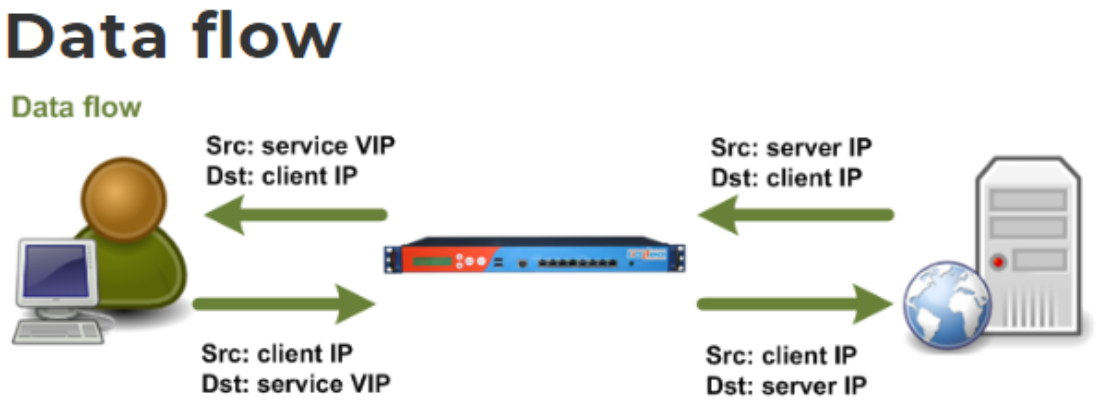
Välityspalvelin toteutuksessa kuormantasaaja ylläpitää kahta erillistä TCP-yhteyttä, asiakkaan ja kuormantasaajan välistä, sekä kuormantasaajan ja palvelimen välistä yhteyttä. Kuvassa 6 asiakas ottaa yhteyttä palvelun VIP-osoitteeseen. Kuormantasaaja terminoi tämän yhteyden ja muodostaa erillisen yhteyden palvelimelle. Kuormantasaaja muuntaa lähdeosoitteen, joten palvelin ei näe asiakkaan käyttämää IP-osoitetta. [19.]



Kuva 6. Proxy-liikenne [19.]

2.4.2 Transparent proxy mode

Läpinäkyvää välityspalvelintoteutusta käytettäessä kuormantasaaja ylläpitää vastaavasti kahta erillistä yhteyttä. Poikkeavuutena kuormantasaaja välittää asiakkaan lähde IP-osoitteen palvelimelle kuvan 20 mukaisesti. Paluuliikenne kulkee normaalisti kuormantasaajan kautta. Kuormantaaja tekee osoitteenmuutoksen, jossa palvelimen lähdeosoite vaihdetaan kuormantasaajan VIP-osoitteeksi. Läpinäkyvää välityspalvelin tilaa käytetään, mikäli halutaan välittää asiakkaan oikea IP-osoite palvelimelle. [20.]



Kuva 7. Transparent proxy -liikenne [20.]

2.5 Algoritmit

Kuormantasauksessa käytettävät algoritmit määritellevät, millä perusteella kuormantasaaja välittää yhteyspyynnöt eteenpäin. Eri kuormantasausalgoritmit käyttävät eri kriteerejä liikenteen välittämiseen. Osa kuormantasausalgoritmeista soveltuu paremmin tasaamaan kuormitusta esimerkiksi yksinkertaisilla web-sivustoilla. Jokin toinen algoritmi voi taas olla optimaalisempi vaikkapa erittäin korkeissa liikennemäärissä. Monimutkaisen web-applikaation liikenteen kuormantasauksessa, voisi dynaaminen algoritmi mahdollisesti tuottaa parhaimman tehokkuuden. Soveltuvimman algoritmin valitsemiseen vaikuttaa aina käyttötarkoitus. [21.]

Oikean algoritmin valitsemisessa tulee huomioida kuormantasauskohteen palvelu ja applikaatio, sekä käytössä oleva palvelin- ja verkkoinfrastruktuuri. Tarjolla olevat algoritmit saattava vaihdella eri laitevalmistajilla. Tässä luvussa käydään läpi yleisimmin käytettäviä algoritmeja.

2.5.1 Round Robin

Round Robin -algoritmi on melko yksinkertainen tapa jakaa kuormitusta. Algoritmissa käytetään palvelinlistausta, jota kierrätetään järjestyksessä. Uusi yhteys välitetään aina seuraavalle järjestyksessä olevalle palvelimelle, jolloin palvelinkohtaisten yhteyksien määrä jakautuu lopulta tasaisesti. Round Robin -metodia käytettäessä palvelimien tulisi olla yhtenäisiä resursseiltaan, jotta se olisi mahdollisimman toimiva. [22; 23.]

Round Robin -algoritmin ongelmana on vaihteleva yhteyksien kesto. Algoritmi ei huomioi ollenkaan palvelimen kuormitustasoa, kuormantasaajan välittäessä uusia yhteyksiä tasaisesti palvelimille, saattavat ne kuormittua epätasaisesti sessioiden pituusvaihtelun takia. [23.]

2.5.2 Weighted Round Robin

Mikäli palvelinympäristössä käytetään eri tehoisia palvelimia, voidaan niiden resursseja hyödyntää tehokkaammin käyttämällä painotusta. Painotetussa Round Robin -algoritmissa palvelimille määritellään staattinen painoarvo. Enemmän tehoa omaaviin palvelimiin voidaan jakaa enemmän liikennettä, jolloin näille määritellään korkeampi arvo. Vähemmän kuormitusta kestäville palvelimelle määritellään pienempi painoarvo. Liikennettä voidaan tämän jälkeen jakaa halutulla suhteella esim. korkealla arvolla omaaville välitetään kaksinkertainen määrä liikennettä suhteessa pienemmän painoarvon palvelimille. [22.]

2.5.3 Least connection

Least connection on dynaaminen kuormantasausalgoritmi, jossa kuormantasaaja monitoroi jatkuvasti palvelinten avoimien yhteyksien määrää. Least connection -algoritmissa uudet yhteydet välitetään palvelimelle, jolla on sillä hetkellä vähiten avoimia yhteyksiä. Algoritmi soveltuu parhaiten useimpien protokollien ja palvelinten käyttötarkoituksiin. Algoritmilla saavutetaan tasainen yhteysmäärä palvelinten kesken. Paras toimivuus saavutetaan palvelinympäristössä, jonka resurssit ovat kapasiteetiltaan yhtenäisiä. Least connection -metodissa on myös mahdollista käyttää painoarvoa, jolloin voidaan hyödyntää eri kapasiteetin omaavia palvelimia paremmin. [22; 23.]

2.5.4 Fastest connection

Fastest connection -algoritmissa kuormantasaaja vastaavasti monitoroi palvelimien vasteaikaa ja välittää uudet yhteydet palvelimelle, jonka vasteaika on nopein. On myös mahdollista käyttää Fastest- ja Least connection -algoritmien yhdistelmää Observed. Observed-algoritmissa monitoroidaan käyttäjämäärän ja vasteajan tasapainoa. Uudet yhteydet välitetään tasapainoltaan parhaimmalle palvelimelle. [23.]

2.5.5 Hash-algoritmit

Hash-algoritmeissa voidaan hyödyntää erilaisia tiivistearvoja (Hash value). Tiivistearvojen perusteella voidaan jakaa liikennettä tarkasti halutulle resurssille. Kuormantasaaja määrittää tiivistearvoja perustuen esimerkiksi IP-paketin otsikkotietoihin. Erilaisia Hash-arvoja voidaan määritellä lukuisia. [24.]

Esimerkiksi Source IP hash -algoritmissa liikennettä voidaan ohjata tietylle palvelimelle lähde-IP-osoitteen mukaan. Destination IP hash -algoritmi vastaavasti ohjaa liikennettä palvelimelle perustuen kohteen IP-osoitteeseen. URL hash -metodissa voidaan liikennettä tasapainottaa käytetyn verkko-osoitteen mukaisesti. Domain Hash -algoritmi jakaa kuormaa käytettävän verkkotunnuksen mukaisesti. [24.]

3 Citrix NetScaler ADC

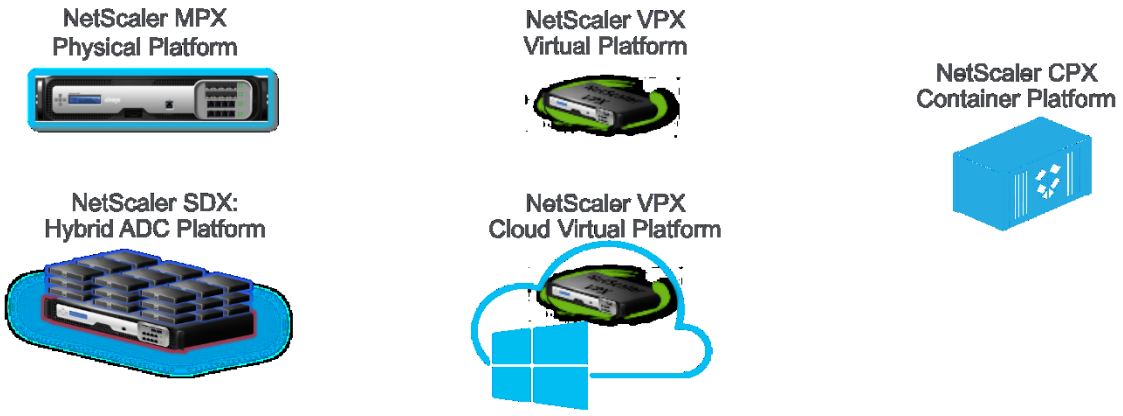
NetScaler Application Delivery Controller (ADC) -järjestelmällä ohjataan sovellusten toimitusta. Se on ominaisuuksiltaan monipuolinen Layer 4-7 -kuormantasausjärjestelmä, jolla voidaan parantaa palveluiden saavuutta ja palvelinympäristön kustannustehokkuutta skaalautuvuuden ansioista. Järjestelmällä voidaan optimoida, monitoroida ja parantaa sovellusten suorituskykyä sekä tietoturvaa. [25.]

NetScaler ADC -järjestelmällä voidaan myös kerätä ja analysoida käyttäjä- ja verkkoliikenneinformaatiota reaaliaikaisesti. Informaation avulla voidaan seurata esimerkiksi eri vasteaikoja ja tietoliikennekapasiteetin käyttöä. Tietoja hyödyntäen voidaan nopeuttaa vianselvitystä ja ennaltaehkäistä mahdollisia ongelmia. [25.]

ADC-järjestelmällä voidaan normaalin palvelinympäristön kuormantasauksen lisäksi uudelleen ohjata liikennettä kokonaan eri konesalissa toimivaan serverifarmiin. Global server load balancing -toiminnon avulla ADC-järjestelmä havaitsee käyttäjiä lähimpänä sijaitsevan konesalin ja ohjaa yhteydet tämän konesalin palvelimille. Toiminto mahdollistaa myös konesalien välisen vikasietoisuuden. Mikäli konesaliyhteydet katkeavat voidaan yhteydet reitittää toimivaan konesaliin. [25.]

Citrix NetScaler tarjoaa joustavan lisenssimallin, joka mahdollistaa kuormantasausjärjestelmän dynaamisen skaalautumisen muuttuneen infrastruktuurin mukaisesti. Tarjolla on kolme eri versioita: Standard Edition, Enterprise Edition ja Platinum Edition. Lisäksi saatavilla on useita eri laitemalleja, sekä virtualisointialustoja eri tarpeisiin. [26; 27.]

Fyysisiä laitealustoja ovat NetScaler MPX ja SDX. Virtualisoitavia ohjelmistoversiota ovat NetScaler VPX ja CPX [28]. Kuvassa 8 esitetty eri NetScaler-laitteistot ja niiden käyttötarkoitukset.

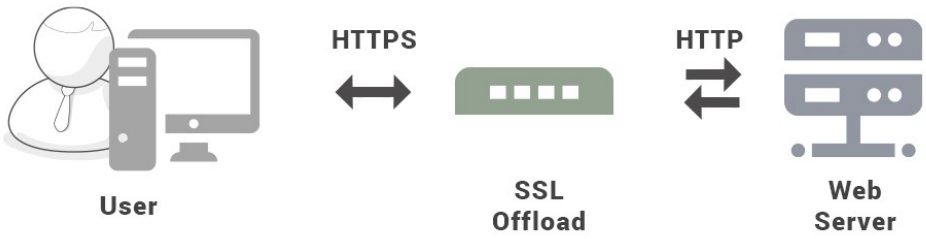


Kuva 8. NetScaler -laitteistot [28].

3.1 Fyysiset alustat NetScaler MPX & SDX

NetScaler MPX -laitteisto on suunnattu yrityksen tarpeeseen, jossa ei ole tarvetta usealla ADC-järjestelmälle. Fyysinen laitteisto on suorituskyvyltään tehokkain alustavaihtoehto. Järjestelmän Layer 7 -liikenteen läpäisy on 500 Mbps–200 Gbps laiteversiosta riippuen. [26.] Fyysisessä laitteistossa on lisäksi SSL (Secure Sockets Layer) kiihdytin, jonka tehtävänä on purkaa ja uudelleen salata HTTPS-liikennettä (SSL Offload) [29].

Kuvassa 8 esitetty SSL Offloadin toiminta. Käyttäjältä tuleva HTTPS-liikenne puretaan laitteistolla ja muunnetaan salaamattomaksi HTTP-liikenteeksi. Vastaavasti palvelimelta tuleva paluu HTTP-liikenne salataan. SSL-salauksen käyttöä esitetään tarkemmin luvussa 4.5.



Kuva 9. SSL Offload [29.]

NetScaler SDX -alusta on suunnattu isojen yritysten sekä palveluntarjoajien tarpeisiin. Yhdellä fyysisellä alustalla on mahdollista ajaa jopa 115 virtualisoitua NetScaler VPX-instanssia. Jokaista instanssia voidaan hallinnoida erikseen, ja ne ovat eriytetty toisistaan. Instansseille voidaan allokoida haluttu määrä laitteiston resursseista. [30.]

3.2 Virtualisoidut alustat VPX & CPX

NetScaler VPX:llä voidaan implementoida virtuaalisesti WMware ESXi, HyperV, KVM ja XenServer hypervisor -alustoille. Hypervisor-alustalla tarkoitetaan ohjelmistoa tai laitteistoa, joka suorittaa virtuaalikoneita. VPX on myös mahdollista implementoida Amazon Web Service-, Microsoft Azure- ja IBM Cloud -pilvipalveluihin.

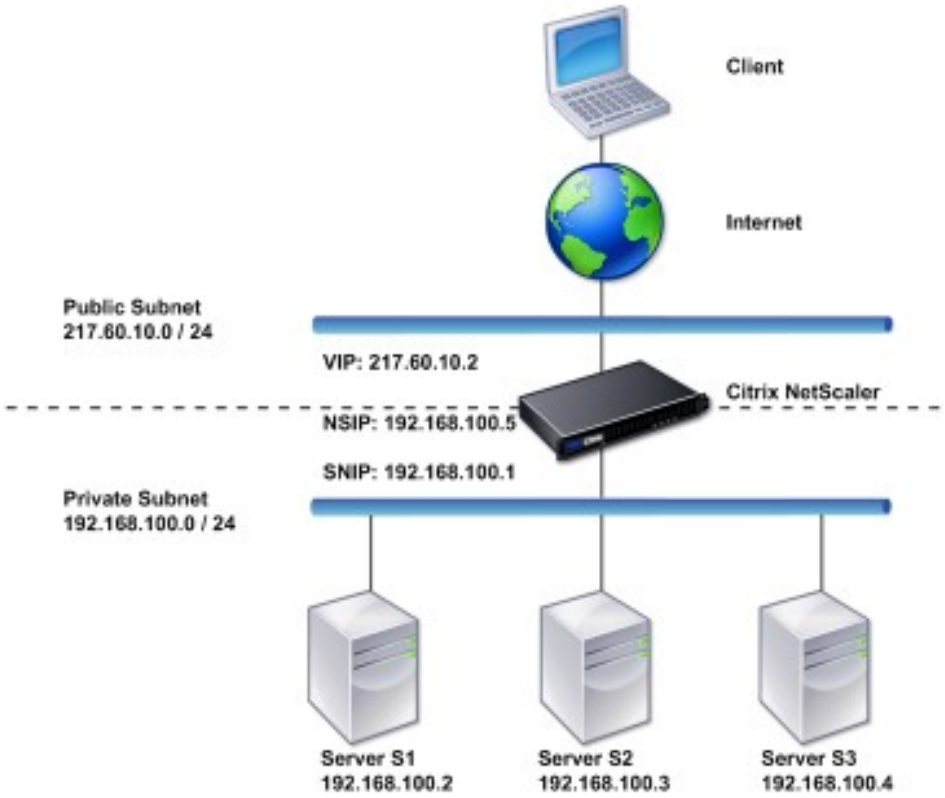
VPX:llä saadaan kustannustehokkaasti kuormantasausjärjestelmä, joka on helppo ja nopea ottaa käyttöön. VPX:stä on saatavilla useita eri malleja eri tarpeisiin. Järjestelmä soveltuu hyvin myös pienen yritysverkon tarpeisiin, jossa liikennemäärät ovat vähäisempiä. Layer 7 -liikenteen läpäisy on mallista riippuen 10 Mbps – 100 Gbps väliltä. [31.]

VPX:n toimintaan vaikuttaa ympäristössä käytettävä hypervisor-alusta. Normaaliin virtuaalikoneympäristöön implementoitu VPX pystyy hallitsemaan vain rajoitetun määrän SSL-liikennettä. Tähän vaikuttaa virtuaalikoneiden prosessorit, joita ei ole suunniteltu suorittamaan SSL offload -toimintoa kovinkaan hyvin verrattuna SDX-alustassa käytettävään SSL-kiihdyttimeen. [32, s.24.]

CPX on uusin Citrix NetScaler -tuote. Se on kevyt ADC-versio, joka tuo layer 4-7 ADC -palvelut Docker-sovelluskontteihin. Se on suunniteltu liitettäväksi varhaisessa vaiheessa sovelluskehitysprosessia ja täten helpottamaan sovelluksen tuotantoon vientiä. CPX tukee SSL offload -ominaisuutta sekä muita L4-L7-ominaisuuksia, kuten sisällönsuodatusta. CPX tukee useita suosittuja sovelluskonttien hallintasovelluksia, kuten Kubernetes, Mesosphere DC/OS ja Apache Mesos Marathon. [33.]

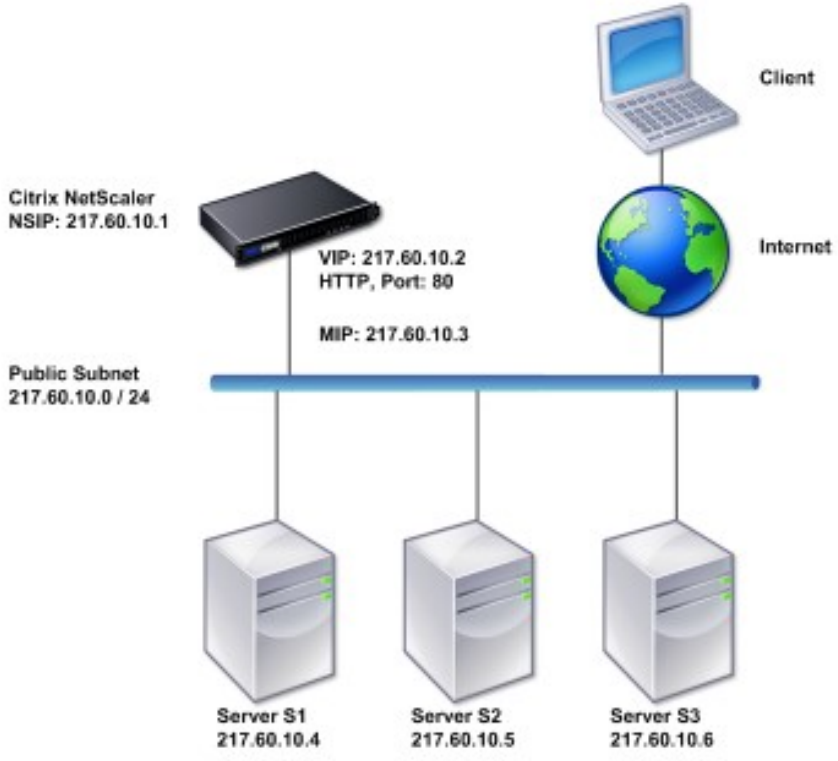
3.3 NetScaler-verkkotopologia

Kuormantasausjärjestelmä useimmiten sijoitetaan asiakkaiden ja servereiden väliin, jolloin kyseessä on two-arm-topologia. Kuvassa 9 esitetty two-arm-topologia on yleisin konfigurointi tapa kuormantasausjärjestelmälle. Tätä konfigurointia käytetään, kun asiakkaat ja serverit sijaitsevat eri aliverkoissa. NetScaler-laitteen verkkoliittimistä toinen on kytketty serveriverkkoon ja toinen asiakasverkkoon. Useimmissa skenaariossa asiakkaat tulevat internetistä ja serverit sijaitsevat sisäverkossa. [33.]



Kuva 10. Two-arm-topologia

Vaihtoehtoisesti voidaan käyttää one-arm-topologiaa, jolloin laite ei ole sijoitettu asiakkaiden ja palvelimen väliin. NetScaler käyttää vain yhtä verkkoliitäntää, joka on samassa aliverkossa servereiden kanssa kuvan 11 mukaisesti. Tämä on yksinkertaisin tapa sijoittaa laite. [33.]



Kuva 11. One-arm-topologia

4 NetScaler VPX:n ominaisuudet ja implementointi

Citrix tarjoaa verkkosivuillaan 90 päivän NetScaler VPX platinum -kokeilulisenssiä. Lisenssin saaminen edellyttää rekisteröitymistä sivustolle. Virtualisoinnissa käytetään Microsoft Hyper-V -ohjelmistoa. Hyper-V on virtualisointialusta, jolla voidaan luoda virtuaalikoneita. Hyper-V valikoitui virtualisointialustaksi, koska se on helposti saatavilla Windows-käyttöjärjestelmässä.

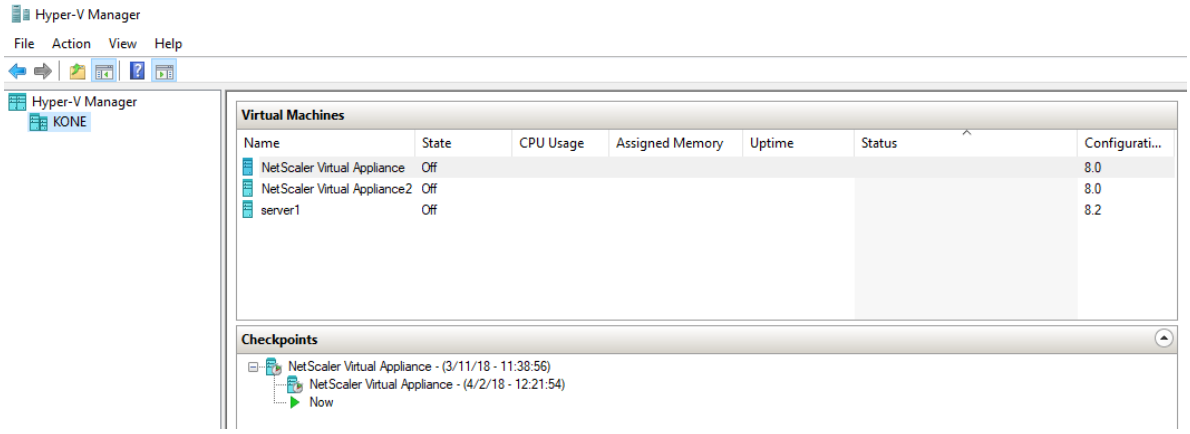
Hyper-V on ollut saatavilla Windows Server -versiosta 2008 alkaen sekä Windows 8.0, 8.1 ja 10 Pro- ja Enterprise-versiossa [34]. Se on helppo ottaa käyttöön käyttöjärjestelmän asetuksista eikä vaadi erillistä lisenssiä. Citrix-verkkosivustolta saa ladattua valmiin NetScaler VPX -virtuaalikonepaketin, joka liitetään Hyper-V Managerin avulla virtuaalikoneeksi.

Microsoft-palvelimen laitteistovaatimuksena on 1,4 GHz 64-bit prosessori, 3 GB:n keskusmuisti ja 32 GB kiintolevy. Yksittäisen NetScaler-virtuaalikoneen laitteistovaatimukset ovat 2 GB keskusmuisti, kaksi virtuaaliprosessoria ja 20 GB kiintolevy. [35.]

Konfiguroinnissa käytetään Hyper-V:n konsoliyhteyttä ja alkumääritysten jälkeen joko web-käyttöliittymää tai järjestelmän komentokehotetta. Yhteys komentoriviin saadaan käyttämällä ilmaista Putty-sovellusta. Putty-sovelluksella voidaan muodostaa SSH-yhteys järjestelmään. SSH on tietoliikenne protokolla, jolla voidaan muodostaa salattu etäyhteys [36].

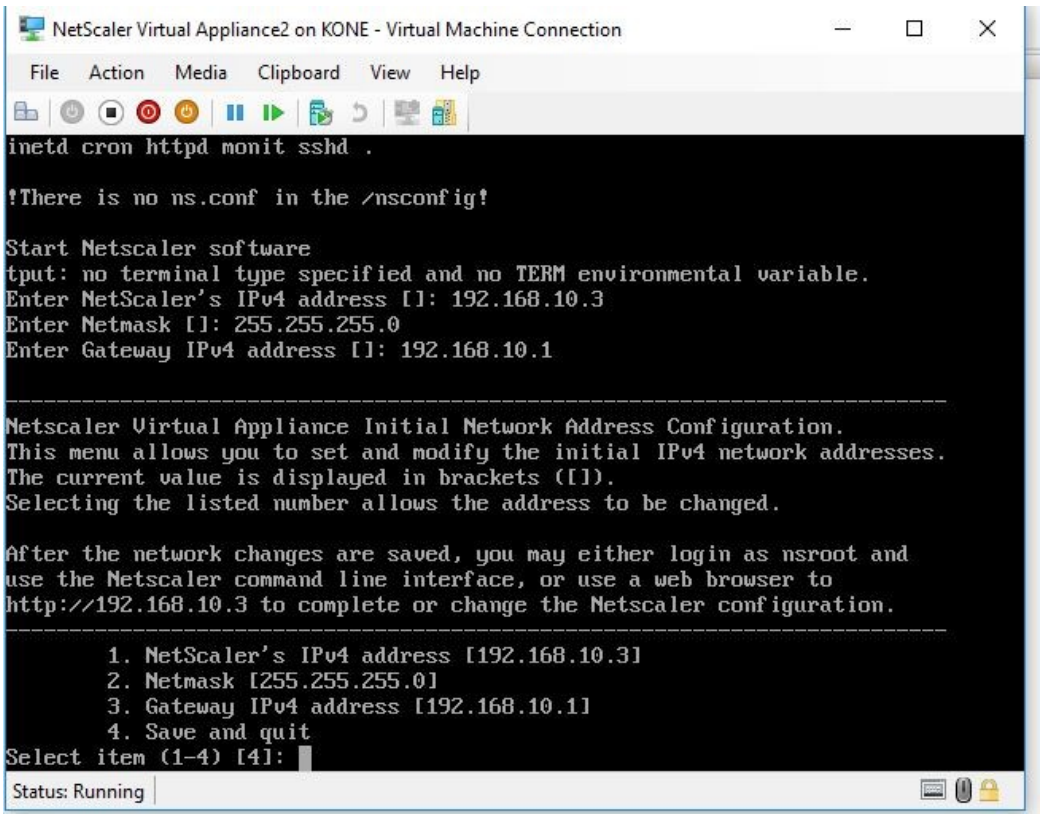
4.1 Virtualisointi ja alkuasetukset

NetScaler-virtuaalikone liitetään Hyper-V Manageriin import virtual machine -toiminolla. Virtuaalikoneen asetuksista määritellään käytettävä verkkokortti, jonka jälkeen voidaan suorittaa virtuaalikoneen käynnistys. Hyper-V Manageriin liitetty NetScaler-virtuaalikone näkyy kuvassa 12.



Kuva 12. Hyper-V-virtuaalikoneet

Ensimmäisellä käynnistyskerralla laitteistolle määritellään perusasetukset konsoliyhteyden kautta. Järjestelmälle määritellään hallintaosoite NetScaler IP Address (NSIP), aliverkon peite ja oletusyhdyskäytävä. Kuvassa 13 näkyy konsoliyhteys virtuaalikoneeseen sekä tehdyt alkuasetukset.

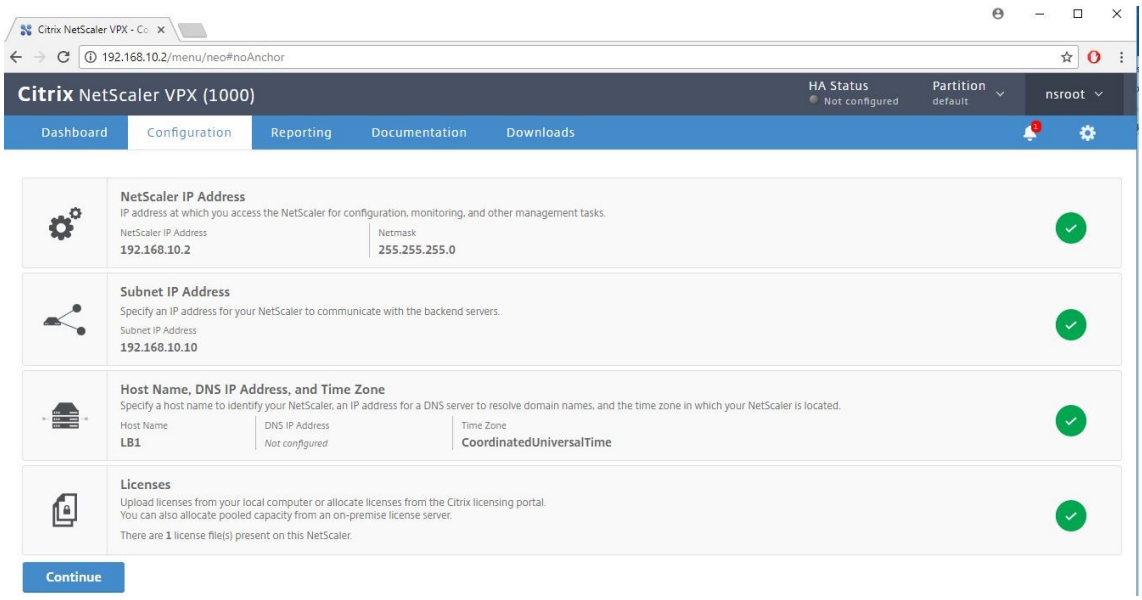


Kuva 13. Alkuasetukset

Alkuasetuksien konfiguroinnin jälkeen järjestelmä käynnistettään uudelleen, jonka jälkeen NSIP-osoitteella voidaan ottaa etäyhteys konsoliin tai vaihtoehtoisesti käyttää Web-käyttöliittymää selaimella.

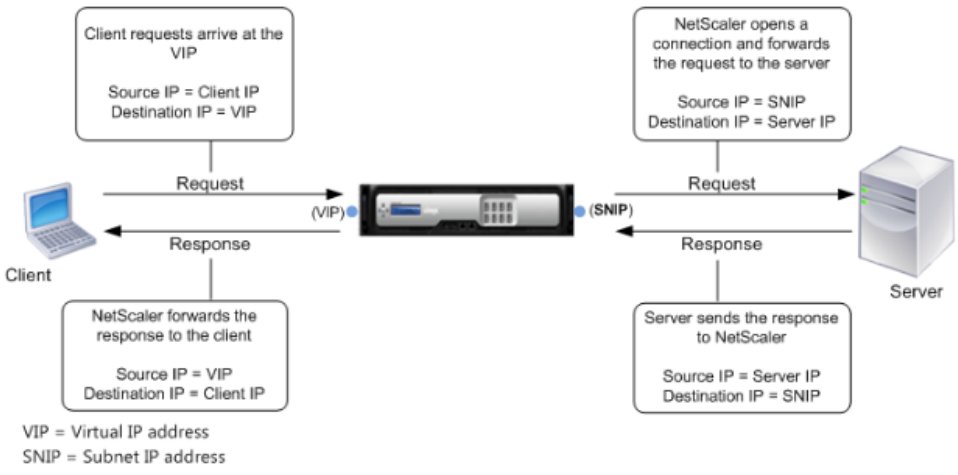
Ensimmäisellä kirjautumiskerralla web-käyttöliittymään avautuu kuvan 14 konfigurointiavustin. Järjestelmälle tulee määritellä nimi, aikavyöhyke ja DNS-palvelin (Domain Name System) eli nimipalvelujärjestelmä. DNS-palvelimien tehtävä on muuntaa verkkotunnukset IP-osoitteeksi. [37.]

Citrix VPX -lisenssi aktivoidaan kirjautumalla rekisteröidyllä tunnuksella Citrix-sivustolle. Sivustolla lisenssinavain sidotaan laitteen MAC-osoitteelle, jonka jälkeen voidaan ladata lisenssitiedosto. Laitteen aktivointi vaatii vielä lisenssitiedoston liittämisen. Onnistuneen aktivoinnin jälkeen lisenssin sallimat ominaisuudet voidaan ottaa käyttöön järjestelmässä.



Kuva 14. NetScaler Configuration wizard

Järjestelmälle tulee lisäksi konfiguroida aliverkon IP-osoite (Subnet IP Address, SNIP). SNIP-osoitetta NetScaler käyttää lähdeosoitteena välittäessä asiakasyhteyksiä palvelimille sekä palvelimien monitorointi yhteyksissä kuvan 15 mukaisesti. Riippuen ympäristön verkkotopologiasta SNIP-osoitteita voidaan tarvittaessa konfiguroida useampi.



Kuva 15. NetScaler VIP- ja SNIP-osoitteen toiminta.

NetScaleriin kytkimen kautta tulevaa palvelinliikennettä varten tulee määritellä SNIP-osoite, joka kuuluu samaan aliverkkoon palvelinten kanssa. Jokaista eri aliverkkoa varten määritellään vastaava SNIP-osoite NetScaleriin. Reitittimen kautta liikennöidessä tulee NetScaler SNIP -osoite kuulua samaan aliverkkoon, joka reitittimellä on määritelty kytkentään.

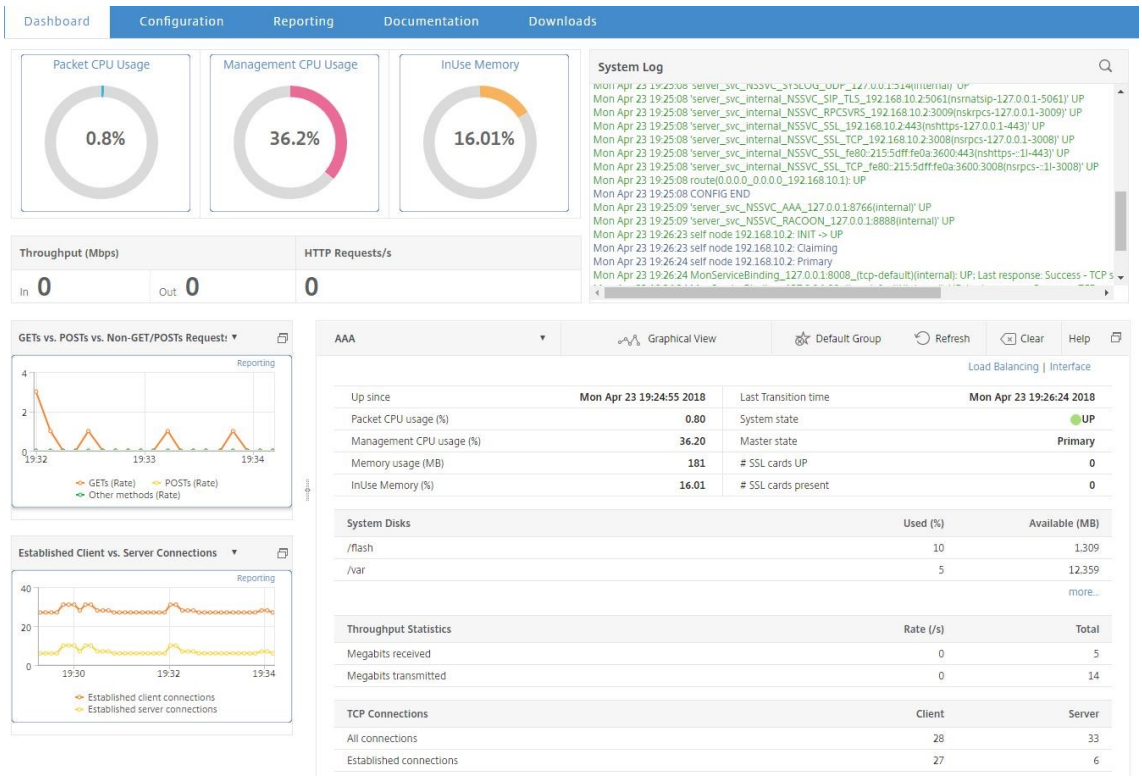
4.2 Web-käyttöliittymä

Web-käyttöliittymä koostuu Dashboard-, Configuration-, Reporting-, Documentation- ja Downloads-näkymistä. Dashboard-näkymässä ilmenee yleiskatsaus laitteen tilasta ja toiminnoista. Dashboardin näkymää voidaan kustomoida monipuolisesti halutulla tavalla. Perusasetuksilla oleva Dashboard näkyy kuvassa 16.

Configuration-näkymässä tehdään kaikki laitteen määritykset. Vaihtoehtoinen tapa tehdä konfiguroinnit komentokehotteen kautta.

Reporting välilehdeltä saadaan tehtyä raportteja käyttäen joko laitteen valmiita raportteja tai vaihtoehtoisesti niitä voi myös luoda itse. Raportteja voidaan myös ladata laitteelta ulos CSV-tiedostomuodossa. CSV tulee sanoista comma-separated values. CSV on tekstitiedosto, johon voidaan tallentaa taulukkomuotoista tietoa [38].

Documentation-valikosta löytyy linkit Citrixin sivuille, josta aukeaa käytössä olevan version dokumentointi. Downloads-valikosta löytyy erilaisia monitorointiin liittyviä tiedostoja. Esimerkiksi sieltä voidaan ladata valmiita SCOM-hallinnointipaketteja. SCOM eli System Center Operation Manager on Microsoftin tuote, jolla voidaan monitoroida palvelimia ja muita verkkoon kytkettyjä laitteita.



Kuva 16. Web-käyttöliittymän dashboard

4.3 Web-palvelimen kuormantasaus

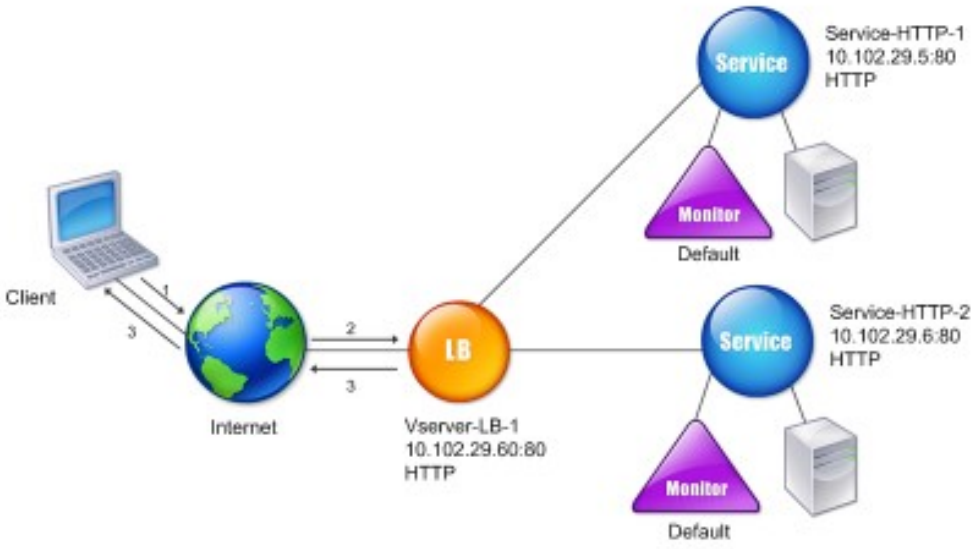
Konfigurointi aloitetaan aktivoimalla NetScalerin kuormantasausominaisuus. Tämän jälkeen lisätään palvelin ja sen IP-osoite serverilistalle. Palvelimen lisäyksen jälkeen määritellään service eli palvelu. Palvelut edustavat palvelimella toimivia sovelluksia. Jokaiselle sovelluspalvelimelle tulee luoda vähintään yksi palvelu, joka sidotaan luontivaiheessa palvelimeen. Palvelulle tulee määritellä nimi, protokolla ja portti. Määriteltyjen tietojen avulla liikennettä reititetään palvelimelle. [39; 40.]

Web-palvelimen palvelulle konfiguroidaan HTTP-protokolla ja portti 80. Servicen asetuksissa voidaan myös määritellä lukuisia eri ominaisuuksia. Halutessa voidaan esimerkiksi rajoittaa palveluun kohdistuvien yhteyksien määrää ja kaistanleveyttä.

Load Balancing Virtual Server (LB vserver) -asetukset tulee vielä määritellä, jotta kuormanjako ja yhteydet voidaan välittää palvelimille. LB vserverit edustavat palvelinfarmin

ryhmiä. LB vserver -asetuksissa määritellään VIP-osoite ja kuormantasauksessa käytettävä algoritmi. Oletuksena NetScaler käyttää least connection -metodia. Palvelu liitetään LB vserver -ryhmään, jolloin kaikki on linkitetty toisiinsa. [40.]

Kuvassa 17 asiakkaan ottaessa yhteyttä web-sovellukseen se lähettää pyynnön LB vserverille määriteltyyn VIP-osoitteeseen, joka on palvelun julkinen IP-osoite. NetScaler lähettää yhteyspyynnön palvelimelle käytettävän kuormantasausalgoritmin mukaisesti. Kuormantasaajan monitorointi valvoo jatkuvasti palvelun tilaa, mikäli palvelu on alhaalla sille ei välitetä yhteyksiä. [40.]



Kuva 17. NetScaler -kuormantasauksen toiminta [40].

Konfiguroinnit tehtiin muuten oletusasetuksilla kuvan 18 komentojen mukaisesti käyttäen komentokehotetta.

Kuva 18. komentokehote ja kuormantasauskonfiguroinnit

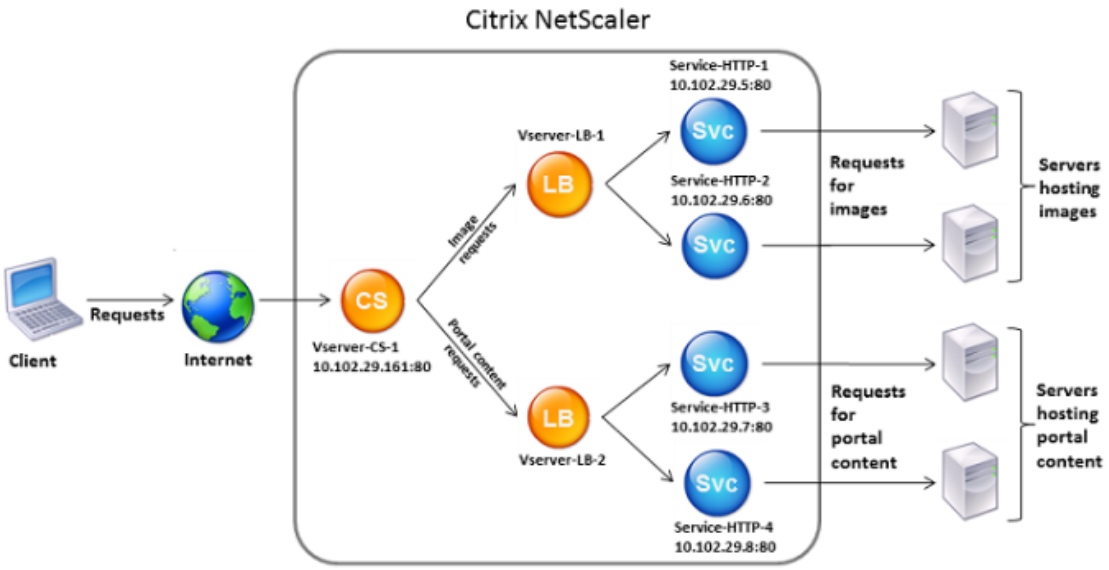
4.4 Content switching

Content switching eli sisällönsuodatus mahdollistaa eri sisällön näyttämisen eri käyttäjille. Sisällönsuodatuksen avulla voidaan esittää haluttua sisältöä esimerkiksi perustuen käyttäjän maantieteelliseen sijaintiin tai vaikkapa laitteeseen, jolla käyttäjä yhdistää palveluun. Netscalerin sisällönsuodatusta voidaan käyttää HTTP-, HTTPS-, TCP- ja UDP-yhteyksissä. HTTPS-yhteyksiä varten tulee SSL Offload -ominaisuuden olla käytössä. [41.]

Konfigurointi aloitetaan ottamalla käyttöön sisällönsuodatusominaisuus. Ensin luodaan content switching virtual server (CS vserver), johon liitetään content switching policy. Sääntöjen avulla määritellään kriteerit erityyppisille pyynnöille ja halutut toimenpiteet. Sääntöjä voi myös olla useita ja niitä voidaan priorisoida halutulla tavalla.

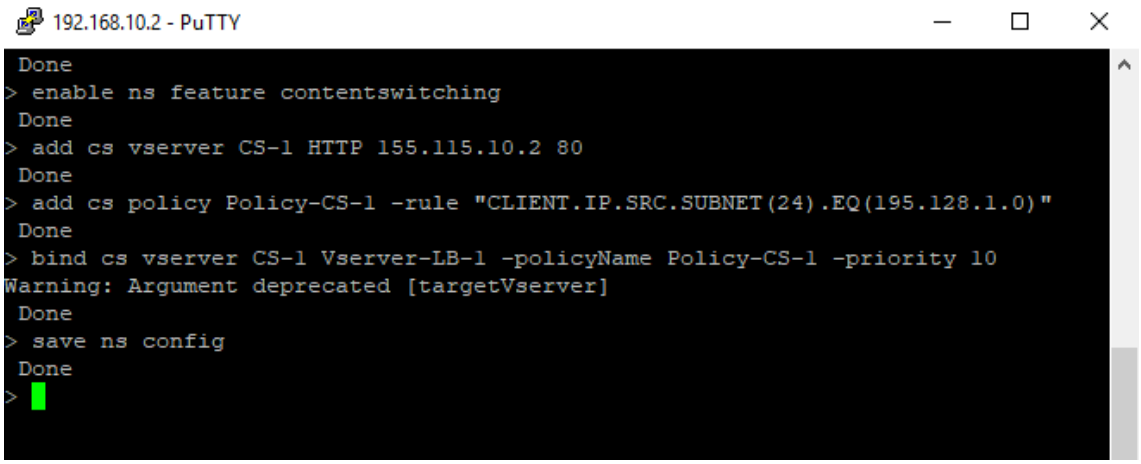
Content Switching -virtuaaliserveri uudelleenohjaa kaikki pyynnöt LB vserverille, joten jokaiselle eri sisältöversiolle tulee olla tehtynä oma LB vserver. Kuvassa 19 havainnol-

listettu Content Switching -ominaisuuden toimintaa. LB virtual server tulee aina määritellä, vaikka palvelimia olisi vain yksi kappale eikä näin ollen kuormantasausta tehdä. [41.]



Kuva 19. Netscaler Content Switching -arkkitehtuuri [41.]

Omassa konfiguroinnissa tein säännön, jossa on määritelty lähde IP-osoitteen rajaus. Liikenteen tullessa määritellystä osoitteesta yhteys välitetään kohde LB virtuaaliserverille. Konfiguroinnit tehtiin kuvan 20 komennoilla.



Kuva 20. Content switching -konfigurointi komentokehotteessa

4.5 SSL-salauksen käyttö

Salatun yhteyden käyttäminen vaatii NetScaler SSL offload -ominaisuuden käyttöönoton. Ominaisuus vaati toimiakseen SSL-sertifikaatin ja avainparin. SSL-varmenteita myöntävät luotetut sertifiointiauktoriteetit (CA, Certificate authority).

NetScaler SSL -asetuksista voidaan luoda CSR (Certificate Signing Request), joka lähetetään validoitavaksi varmenteen myöntäjälle. NetScalerissa on myös mahdollista käyttää järjestelmän omia sertifikaatteja testaustarkoitukseen. Oikeassa tuotantoympäristössä tulisi kuitenkin käyttää vain luotettujen sertifiointiauktoriteettien myöntämiä varmenteita. [42.]

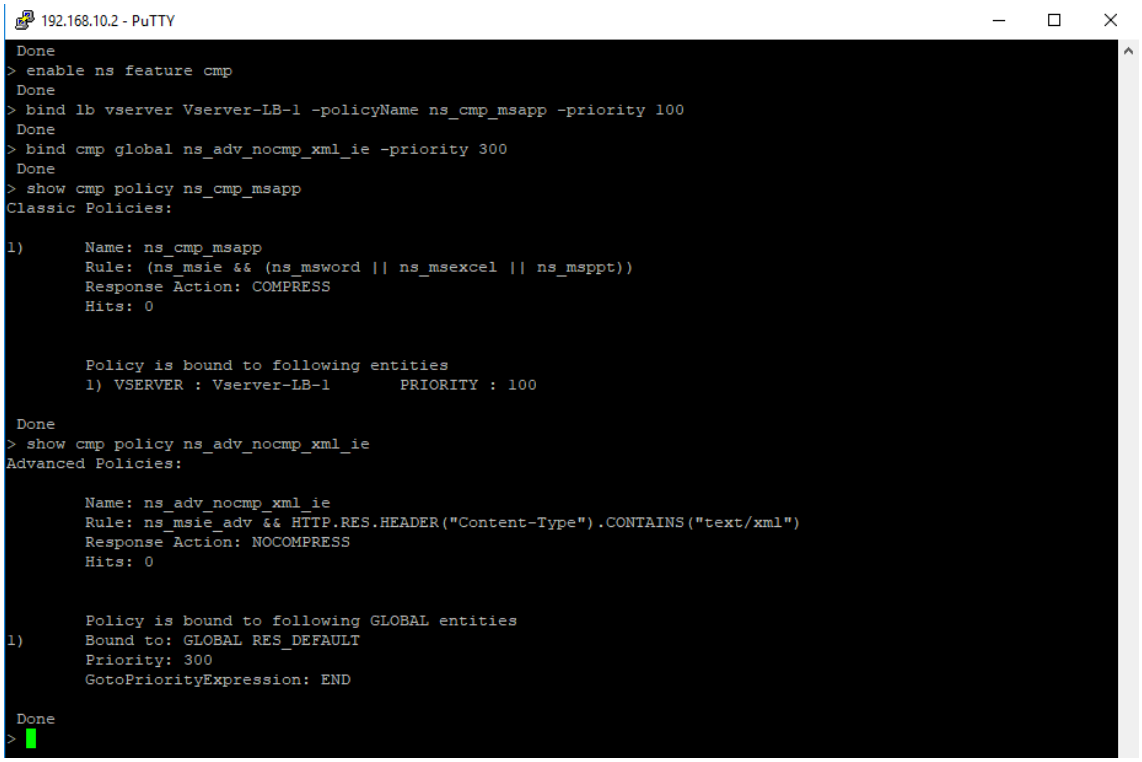
NetScalerilla voi luoda Root-CA-, Intermediate- ja End-user-sertifikaatteja. Ennen sertifikaatin luontia tulee luoda privaattiavain, joka normaalisti lähetettäisiin CSR-hakemuksessa varmenteen myöntäjälle. Tässä tapauksessa käytettiin NetScalerin CA-työkalua sertifikaatin luomiseen ja allekirjoittamiseen. SSL-asetuksista löytyvä työkalun avulla loin ja asensin End-user-sertifikaatin. [43.]

4.6 HTTP-kompressointi

NetScaler-laitteistolla voidaan optimoida ja vähentää liikennemäärää asiakkaan ja sovelluksen välillä. Optimointi ominaisuudet on mahdollista ottaa käyttöön Enterprise- ja Platinum-lisensseillä. Yksi tapa optimoida liikennettä on ottaa käyttöön HTTP-kompressointi. HTTP-kompressoinnissa käytetään GZIP- tai DEFLATE-kompressointialgoritmia, jonka avulla voidaan pakata asiakkaalle lähtevää tai tulevaa dataa. Asiakkaan selaimen tulee kuitenkin tukea kompressointia. [32, s.103-104.]

HTTP-kompressointi otetaan käyttöön NetScalerin ominaisuuksista. Kompressointi voidaan ottaa käyttöön globaalina sääntönä, jolloin se vaikuttaa kaikkeen NetScalerin kautta kulkevaan liikenteeseen tai vaihtoehtoisesti se voidaan sitoa sovelluskohtaisesti. Kompressointi toimii sääntöpohjaisesti, joten sille määritellään sääntö ja toimenpide. Toimenpiteeseen määritellään, mitä tehdään, mikäli sääntö toteutuu. NetScalerissa on valmiina useita sisäänrakennettuja sääntöjä, joita voidaan hyödyntää. [32, s.104-106.]

Kuvan 21 näkyvässä konfiguraatiossa käytin valmista kompressointisääntöä, joka kompressoi Microsoft Word-, Excel- ja Powerpoint-sovelluksien luomat tiedostot. Säännön liitin aiemmin luomaani LB-vserveriin. Lisäksi tein globaalin säännön, jossa XML- ja tekstitiedostot jätetään kompressoimatta, jos pyynnöt lähetään Microsoft Internet Explorer selaimesta.



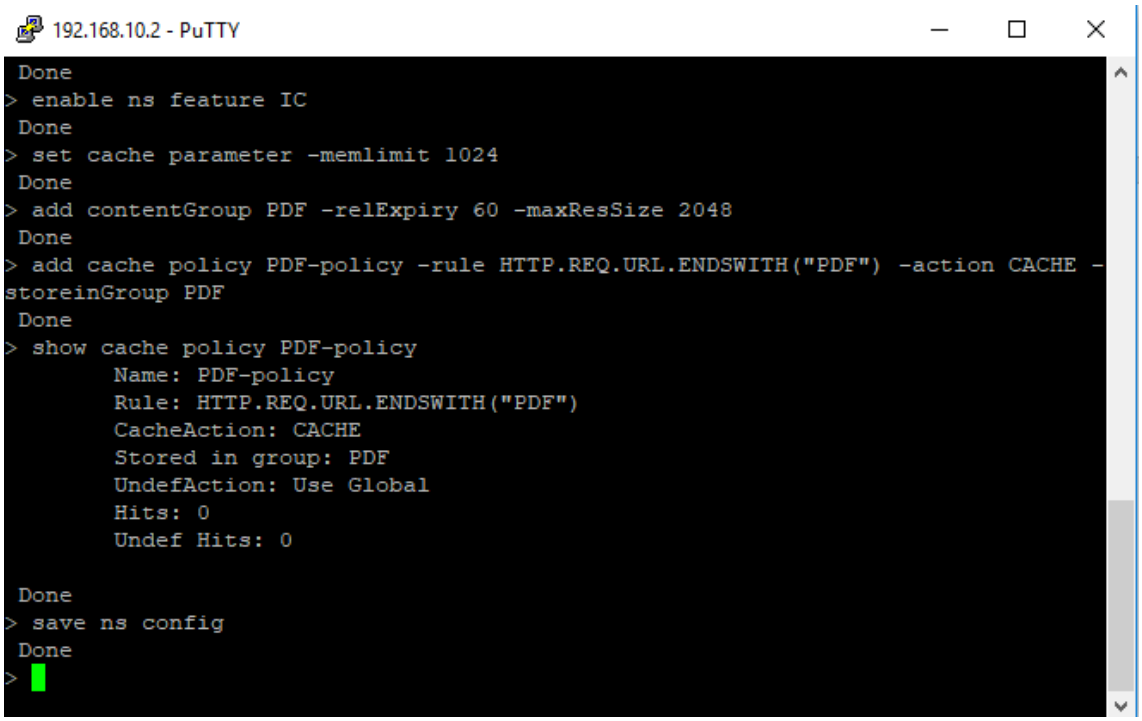
Kuva 21. Kompressointi konfiguroinnit

4.7 Integrated Caching

Integrated Caching -toiminnon avulla NetScaler voi tallentaa web-palvelimen sisältöä omaan välimuistiin ja välittää sitä suoraan käyttäjille, jolloin palvelun käyttö nopeutuu. Toiminnaltaan välimuistin käyttö toimii samalla tavalla kuin verkkoselaimissa. Välimuistio ominaisuutta voidaan käyttää vain HTTP-protokollassa. NetScalerin välimuistiin voidaan tallentaa staattista ja dynaamista dataa. Staattisella datalla tarkoitetaan esimerkiksi yksinkertaisia verkkosivuja ja kuvatiedostoja. Dynaamisella datalla tarkoitetaan muuttuvaa dataa. [32, s.112.]

Toiminnon käyttöönotossa tulee ensin määritellä käytettävän välimuistin määrä, joka voi olla maksimissaan puolet NetScalerille allokoidusta muistin määrästä. Välimuistin määrän asettamisen jälkeen järjestelmä pitää uudelleenkäynnistää. Integrated Caching toimii myös sääntöpohjaisesti. Lisäksi sille määritellään content group, jonne välimuistiobjektit säilötään. [32, s.112-113.]

Kuvan 22 konfiguroinneissa loin uuden sisältöryhmän, jonka tiedostoille määriteltiin 60 sekunnin säilömisaika sekä suurin sallittu välimuistiin tallennettava tiedostokoko 2048 MB. Luotu sääntö sallii NetScalerin tallentaa välimuistiin kaikki objektit, joiden verkko-osoitteen lopussa on pdf.



Kuva 22. Integrated Cachingin konfiguraatiot

NetScalerin optimointi ominaisuuden käyttävät järjestelmän prosessoria ja muistia, joten käyttöönotossa tulee huomioida järjestelmälle allokoitujen resurssien riittävyys.

4.8 High Availability

High Availability (HA) tarkoitetaan korkeaa saatavuutta. Ympäristön korkean saatavuuden takaamiseksi tulisi myös huomioida NetScaler-laitteiston kahdennuksesta. Kahdentamaton kuormantasaaja muodostaa kriittisen SPOF-pisteen ympäristössä, mikäli NetScaler vikaantuisi, aiheuttaisi se käyttökatkon koko palveluun.

NetScaler-järjestelmällä HA voidaan toteuttaa usealla eri tavalla. Yleisin tapa on lisätä toinen NetScaler-laitteisto samaan aliverkkoon ja muodostaa laitepari käyttämällä aktiivi/passiivi-tilaa. Aktiivi/passiivi-tilassa laiteparista toinen määritellään ensisijaiseksi, jolloin se toimii aktiivisena ja hoitaa kaiken liikenteen. Toissijainen passiivitilassa oleva laitteisto toimii varalla ja monitoroi aktiivilaitteen tilaa. Mikäli aktiivilaite lakkaa vastaamasta, toissijainen laite alkaa vastaanottamaan liikennettä. HA-laitepari on mahdollista ottaa käyttöön vain, jos laitteet ovat samaa mallia. [32, s.119.]

Ennen HA-määritystä tulee molempien järjestelmien olla lisäksi samassa ohjelmistoversiossa ja asetusten tulee olla yhtenäiset pois lukien NSIP-osoite sekä laitenimi. HA-konfigurointi olisi vaatinut toisen NetScaler-lisenssin, joten HA-konfigurointia en pystynyt toteuttamaan. [43.]

HA-asetuksista lisätään toinen laite NSIP-osoitteen avulla. Asetuksissa voidaan lisäksi määritellä monitorointiviestien lähetysväli sekä aikamääre, minkä puitteissa passiivilaite aktivoituu, mikäli se ei saa vastausta viesteihin. Oletusasetuksilla viestin lähetysväli on 200 ms ja yliheitto tapahtuu, mikäli laite ei saa vastauksia kolmeen sekuntiin. HA-toteutuksessa järjestelmien konfiguraatiot synkronoidaan aktiivilaitteelta passiivilaitteelle. Asetukset synkronoituvat, mikäli passiivilaite käynnistyy uudelleen tai aktiivilaitteesta tulee passiivilaite vikatilanteessa. [45.]

Konfigurointi muutoksia voidaan ainoastaan tehdä aktiivilaitteelta. Syötetty komento välitetään ensin passiivilaitteelle, jossa se suoritetaan. Jos komennon suorittaminen ei onnistu toissijaisella laitteella, ensisijainen laite suorittaa komennon ja tekee virhemerkinnän lokiin. [32, s.121; 46.]

5 Yhteenveto

Insinöörityössä tutkittiin yleisesti palvelin infrastruktuurin kuormantasauksen toimintaa ja käyttötarkoitusta. Kuormantasauksessa käytettävien tekniikoiden ja algoritmien ymmärtäminen auttaa valitsemaan soveltuvimmat vaihtoehdot ylläpidettäville palvelimille ja niiden sovelluksille. Lisäksi työssä implementoitiin Citrix NetScaler VPX -kuormantasausjärjestelmä virtualisoidusti käyttäen Microsoft Hyper-V -virtualisointialustaa.

Aiempi käyttökokemukseni NetScaler-järjestelmästä rajoittui lähinnä olemassa olevien konfigurointien muutoksiin, joten järjestelmän asennus ja ominaisuuksien käyttöönotto alusta alkaen auttoi ymmärtämään entistä paremmin järjestelmän toimintaa. Konfiguroinnissa pyrittiin käyttämään mahdollisimman paljon komentokehotetta, jotta käyttöjärjestelmässä käytettävät komennot tulisivat myös tutuksi. Komentojen käyttö oli alussa haastavaa, koska aiempi kokemus rajoittui pelkkään web-käyttöliittymän käyttöön.

NetScaler VPX -kuormantasausjärjestelmässä otettiin käyttöön useita eri ominaisuuksia, joten tavoitteena ollut perustason kuormantasaus sekä syvällisempi tutustuminen laitteen kuormantasaustoimintoihin täyttyi. Käytössä ollut NetScaler Platinum -lisenssi avaa laitteen kaikki ADC-toiminnot. Käyttöönotossa huomattiin monia muita lisäominaisuuksia, jonka avulla järjestelmällä voidaan suorittaa paljon muutakin kuin pelkkää kuormanjakoa.

Jatkuvasti kehittyvät web-sovellukset ja muuttuvat verkkoinfrastruktuurit vaativat myös kehittyviä tietoturvaratkaisuja. NetScaler ADC -järjestelmässä voitaisiin ottaa käyttöön verkkosovelluspalomuuri, jolla voidaan torjua nykyaikaisia sovelluksiin kohdistuvia hyökkäyksiä entistä tehokkaammin. Järjestelmän tietoturvaominaisuuksista on myös mahdollista ottaa käyttöön Webroot-palvelun IP-osoitetietokanta, jossa ylläpidetään aktiivisesti IP-osoitteiden mainetta. Automaattisesti päivittyvästä tietokannasta saatavat huono maineiset IP-osoitteet voidaan estää laitteella ja näin parantaa ympäristön tietoturvaa entisestään.

Lähteet

1 Agarwal, Aakash. 2015. Load Balancing Overview & NLBs (Network Load Balancer). Verkkoaineisto. <https://www.slideshare.net/AakashAgarwal15/ace-comcore-47431191> Luettu 27.10.2017.

2 Anderson, Melissa. 2017. What is Load Balancing. Verkkoaineisto <https://www.digitalocean.com/community/tutorials/what-is-load-balancing>. Luettu 27.10.2017.

3 What is load balancing. Verkkoaineisto. Citrix Systems, Inc. <https://www.citrix.com/glossary/load-balancing.html>. Luettu 24.4.2018.

4 NetScaler AppFirewall and Web Security Service. Verkkoaineisto. Citrix Systems, Inc. Verkkoaineisto. <https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/citrix-netscaler-application-firewall-datasheet.pdf>. Luettu 24.4.2018.

5 Anicas Mitchell. 2014. An Introduction to HAProxy and Load Balancing Concepts. Verkkoaineisto <https://www.digitalocean.com/community/tutorials/an-introduction-to-haproxy-and-load-balancing-concepts>. Luettu 27.10.2017.

6 Membrey, Peter; Plugge, Eelco & Hows, David. 2012. Practical Load Balancing. E-kirja. Apress.

7 Colliander Andreas. 1999. ISOn OSI-mallin rakenne ja käyttö. Verkkoaineisto. <http://www.tml.tkk.fi/Studies/Tik-110.300/1999/Essays/essee_OSI.html>. Luettu 4.11.2017.

8 Issac Luke P. 2016. How does Software and Hardware Load Balancer Work. Verkkoaineisto. <https://www.thegeekstuff.com/2016/01/load-balancer-intro/>. Luettu 5.11.2017.

9 Load balancing FAQ. Verkkoaineisto. HAProxy Technologies, LLC. <https://www.haproxy.com/blog/loadbalancing-faq/>. Luettu 5.11.2017.

10 Osoitteenmuunnos. Verkkoaineisto. Wikipedia. < https://fi.wikipedia.org/wiki/Osoitteenmuunnos>. Luettu 29.4.2018.

11 Assman, Baptiste. 2011. Layer 4 load balancing NAT mode. Verkkoaineisto. <https://www.haproxy.com/blog/layer-4-load-balancing-nat-mode/>. Luettu 5.11.2017.

12 MAC address, Verkkoaineisto. Wikipedia. < https://en.wikipedia.org/wiki/MAC_address>. Luettu 1.5.2018.

13 Turnbull, Malcolm. 2015. What are the best load balancing methods and algorithms. Verkkoaineisto <http://www.loadbalancer.org/blog/load-balancing-methods/>. Luettu 16.12.2017.

14 Assmann, Baptiste. 2011. Layer 4 load balancing Direct Server Return mode. Verkkoaineisto. <https://www.haproxy.com/blog/layer-4-load-balancing-direct-server-return-mode/>. Luettu 16.12.2017.

15 MacVittie, Lori. 2008.The Disadvantages of DSR. Verkkoaineisto. < https://devcentral.f5.com/articles/the-disadvantages-of-dsr-direct-server-return/>. Luettu 9.5.2017.

16 TCP/IP -protokollat. 2001. Verkkoaineisto <http://mrin.mbnet.fi/paatto-tyo/arp.html>. Luettu 1.5.2018.

17 Assmann, Baptiste. 2011. Layer 4 load balancing tunnel mode. Verkkoaineisto. <https://www.haproxy.com/blog/layer-4-load-balancing-tunnel-mode/>. Luettu 7.5.2018.

18 Phatak, Prashant. 2011. Layer 7 Load Balancers. Verkkoaineisto. <http://opensourceforu.com/2011/04/layer-7-load-balancers/>. Luettu 13.1.2018.

19 Assman, Baptiste. 2011. Layer 7 load-balancing proxy mode. Verkkoaineisto. <https://www.haproxy.com/blog/layer-7-load-balancing-proxy-mode/>. Luettu 5.11.2017.

20 Assman, Baptiste. 2011. layer 7 load-balancing transparent proxy mode. Verkkoaineisto. < https://www.haproxy.com/blog/layer-7-load-balancing-transparent-proxy-mode/>. Luettu 5.11.2017.

21 Load Balancing Algorithms. Citrix Systems, Inc. Verkkoaineisto. <https://docs.citrix.com/zh-cn/netscaler/11/traffic-management/load-balancing/load-balancing-customizing-algorithms.html>. Luettu 4.1.2018.

22 Load Balancing Algorithms. Avi networks. Verkkoaineisto. <https://avinetworks.com/docs/17.2/load-balancing-algorithms/>. Luettu 4.1.2018.

23 Mac, Vittie. 2010. Intro to Load Balancing for Developers – The Algorithms. Verkkoaineisto. <https://devcentral.f5.com/articles/intro-to-load-balancing-for-developers-ndash-the-algorithms>. Luettu 4.1.2018.

24 About Hashing Methods. Citrix Systems, Inc. Verkkoaineisto. <https://docs.citrix.com/ja-jp/netscaler/11/traffic-management/load-balancing/load-balancing-customizing-algorithms/hashing-methods.html>. Luettu 1.5.2018.

25 What is an Application Delivery Controller? Verkkoaineisto. Citrix Systems, Inc. < https://www.citrix.com/products/netscaler-adc/resources/what-is-an-adc.html/>. Luettu 15.3.2018.

26 Why NetScaler. Verkkoaineisto. Citrix Systems, Inc. <https://www.citrix.com/products/netscaler-adc/why-netscaler.html>. Luettu 15.3.2018.

27 Platforms. Verkkoaineisto. Citrix Systems, Inc <https://www.citrix.com/products/netscaler-adc/platforms.html>. Luettu 15.3.2018.

28 Maslo, Alexander. 2016. Why is NetScaler the Best ADC for Microsoft Customers. Verkkoaineisto. <https://www.citrix.com/blogs/2016/03/21/why-is-netscaler-the-best-adc-for-microsoft-customers/>. Luettu 15.3.2018.

29 Verkkoaineisto. Comodo Group, Inc. <https://securebox.comodo.com/ssl-sniffing/ssl-offloading/>. Luettu 24.4.2018.

30 Maximizing Multi-tenancy with. Citrix NetScaler. Verkkoaineisto. Citrix Systems, Inc. <https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/maximizing-multi-tenacy-with-citrix-netscaler-en.pdf>. Luettu 25.4.2018.

31 Verkkoaineisto. Citrix Systems, Inc. < https://www.citrix.com/products/netscaler-adc/resources/netscaler-cpx-data-sheet.html>. Luettu 25.4.2018.

32 Sanbu, Marius. 2014. Implementing NetScaler VPX. Birmingham. Packt Publishing.

33 Understanding Common Network Topologies. Verkkoaineisto Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/getting-started-with-netscaler/network-topologies.html>. Luettu 30.4.2018.

34 Hyper-V. Verkkoaineisto. Wikipedia <https://en.wikipedia.org/wiki/Hyper-V>. Luettu 30.4.2018

35 Installing Citrix NetScaler Virtual Appliances on Microsoft Hyper-V Servers. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/getting-started-with-vpx/install-vpx-on-hyper-v.html?_ga=2.250016262.1014746668.1522657280-1251695107.1506444723>. Luettu 29.4.2017.

36 SSH. Verkkoaineisto. SSH Communications Security Inc. <https://www.ssh.com/ssh/>. Luettu 27.4.2018.

37 Configuring Subnet IP Addresses (SNIPs). Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/12/networking/ip-addressing/configuring-netscaler-owned-ip-addresses/configuring-subnet-ip-addresses-snips.html>. Luettu 27.4.

38 CSV. Verkkoaineisto. Wikipedia. <https://fi.wikipedia.org/wiki/CSV>. Luettu 27.4.2018.

39 Setting Up Basic Load Balancing. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/traffic-management/load-balancing/load-balancing-setup.html>. Luettu 27.4.2018.

40 How a NetScaler Communicates with Clients and Servers. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/getting-started-with-netscaler/communicate-with-clients-servers.html>. Luettu 10.5.2018.

41 Content Switching. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/traffic-management/content-switching.html>. Luettu 29.4.2018.

42 Obtaining a Certificate from a Certificate Authority. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/traffic-management/ssl/manage-certs/obtain-cert-frm-cert-auth.html/>. Luettu 10.5.2018.

43 Generating a Test Certificate. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/traffic-management/ssl/manage-certs/generate-self-signed-cert.html>. Luettu 12.5.2018.

44 Considerations for a High Availability Setup. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/system/high-availability-introduction/high-availability-consideration-points.html>. Luettu 13.5.2018.

45 Configuring Synchronization. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/system/high-availability-introduction/configuring-synchronization-high-availability.html>. Luettu 13.5.2018.

46 Configuring Command Propagation. Verkkoaineisto. Citrix Systems, Inc. <https://docs.citrix.com/en-us/netscaler/11/system/high-availability-introduction/configuring-command-propagation-high-availability.html>. Luettu 13.5.2018.