

Kristopher Vuorela

# VIRTUALISOINTIALUSTOJEN SUORITUSKYKYVERTAILU

Opinnäytetyö  
Tietotekniikka

2018



**Kaakkois-Suomen  
ammattikorkeakoulu**

<b>Tekijä/Tekijät</b>	<b>Tutkinto</b>	<b>Aika</b>
Kristopher Vuorela	Tietotekniikka	Kesäkuu 2018
<b>Opinnäytetyön nimi</b>		55 sivua
Virtualisointialustojen suorituskykyvertailu		
<b>Toimeksiantaja</b>		
Kaakkois-Suomen ammattikorkeakoulu, ICTLAB		
<b>Ohjaaja</b>		
Lehtori Tomi Pahula		
<b>Tiivistelmä</b>		
<p>Opinnäytetyön tavoitteena oli suunnitella sekä toteuttaa virtualisointialustojen suorituskykyvertailu. Tämä toteutettiin mahdollisimman tasavertaisesti suorittamalla täysin samat testit samoilla yksityiskohdilla ja samalla palvelimella. Lopuksi tulokset dokumentoitiin ja vertailtiin keskenään. Opinnäytetyössä käytettiin kvalitatiivisia eli laadullisia tutkimusmenetelmiä.</p> <p>Työssä käytettiin kolmea eri virtualisointialustaa: Proxmox, ESXi, sekä Hyper-V. Proxmox on Proxmox Server Solutions GmbH:n kehittämä Linux-pohjainen avoimen lähdekoodin alusta. ESXi on VMwaren kehittämä virtualisointialusta, joka kuuluu vSphere-tuoteryhmään. Hyper-V on Microsoftin kehittämä alusta, joka pyörii uusimmissa Windows-tuotteissa. Kullekin virtualisointialustalle asennettiin ensin viisi Windows 10- sekä viisi Fedora Linux -käyttöjärjestelmää. Tämän jälkeen suoritettiin ensimmäiset suorituskykytestit, jonka jälkeen virtuaalilaitteiden määrä nostettiin yhteensä kahteenkymmeneen. Samat suorituskykytestit suoritettiin, kun virtuaalilaitteita oli kaksikymmentä. Työt suoritettiin kaksi kertaa, jotta voitiin nähdä, aiheuttiko virtuaalilaitteiden määrä suuria eroja tuloksissa. Kun testit olivat valmiit, tulokset syötettiin Excel-ohjelmaan helposti luettavaan muotoon.</p> <p>Työ tehtiin onnistuneesti, sillä tulokset saatiin onnistuneesti talteen kaikilta virtualisointialustoilta. Työn aikana ilmeni joitain ongelmia tulosten kanssa, mutta ongelmia tuottaneiden tulosten testit suoritettiin uudestaan tarvitsemalla virtualisointialustalla. Tämän opinnäytetyön testejä voidaan jatkossa soveltaa esimerkiksi tehokkaammalla palvelimella, josta voi saada lisää mielenkiintoista tilastotietoa tutkittavaksi.</p>		
<b>Asiasanat</b>		
virtualisointi, hyper-v, kvm, esxi, suorituskykytesti, suorituskyky		

<b>Author (authors)</b>	<b>Degree</b>	<b>Time</b>
Kristopher Vuorela	Bachelor of Engineering	June 2018
<b>Thesis title</b>		55 pages
Comparison of virtualization performances		
<b>Commissioned by</b>		
South-Eastern Finland University of Applied Sciences, ICTLAB		
<b>Supervisor</b>		
Senior Lecturer Tomi Pahula		
<b>Abstract</b>		
<p>The goal of this thesis was to design and implement a performance comparison of certain virtualization platforms. This was carried out as equally as possible by executing the exact same benchmarks and tests with the same details. All hypervisors were operated within the same server. After the benchmarks and tests, the results were documented and compared with each other. Qualitative research methods were used in this thesis.</p> <p>There were three hypervisors that were used in this thesis: Proxmox, ESXi and Hyper-V. Proxmox is a Linux-based open-source hypervisor, ESXi is a vSphere product developed by VMware, and Hyper-V is developed by Microsoft which runs on newest Microsoft Windows products. Ten virtual machines were first installed on each hypervisor: five Windows 10 and five Linux virtual machines. After this, the first set of benchmarks and tests were executed. The same benchmarks and tests were also executed with twenty virtual machines. The reason to operate the same tests twice with a different number of virtual machines was to see if the amount of virtual machines would affect the results and by how much. When the tests were completed, the results were entered into an easily readable form inside an Excel file.</p> <p>This goal was reached as planned, as the results were successfully recovered from all virtualization platforms. During the work, there were some problems with the results, but all the tests that had these problems were re-run on the needed hypervisor. This bachelor's thesis can be continued with, for example, a more efficient server to get more interesting statistics.</p>		
<b>Keywords</b>		
virtualization, hyper-v, proxmox, esxi, benchmark, performance		

# SISÄLLYS

LYHENTEET.....	6
1 JOHDANTO.....	7
2 VIRTUALISOINTI.....	8
2.1 Historia .....	8
2.2 Palvelimen virtualisointi .....	9
2.3 Pöytäkoneen virtualisointi .....	9
2.4 Sovellusten virtualisointi .....	10
2.5 Hyödyt .....	10
2.6 Virtualisointialusta.....	12
2.6.1 Tyypin 2 virtualisointialusta .....	13
2.6.2 Tyypin 1 virtualisointialusta .....	13
3 WINDOWS SERVER 2016 JA HYPER-V.....	14
3.1 Laitevaatimukset.....	15
3.2 Hyper-V .....	15
3.3 LISENSOINTI .....	16
4 VSPHERE JA ESXI .....	18
4.1 ESXi.....	18
4.2 LISENSOINTI .....	19
4.3 Laitevaatimukset.....	20
5 PROXMOX .....	20
5.1 KVM.....	21
5.2 Laitevaatimukset.....	21
6 LAITTEISTO .....	21
7 VIRTUALISOINTIALUSTOJEN ASENNUKSET JA VALMISTELU.....	22
7.1 Proxmox-asennus ja -valmistelu .....	23
7.2 Windows Server 2016 sekä Hyper-V-asennus ja -valmistelu .....	26
7.3 ESXi-asennus ja -valmistelu .....	29

8	SUORITUSKYKYTESTIT JA TULOSTEN VERTAILU.....	32
8.1	IOPS-mittaukset.....	33
8.1.1	Ensimmäinen IOPS-testi.....	35
8.1.2	Toinen IOPS-testi .....	35
8.1.3	Kolmas IOPS-testi .....	36
8.2	Keskusmuistin nopeuden mittaukset .....	37
8.2.1	Ensimmäinen lukutesti.....	38
8.2.2	Toinen lukutesti.....	38
8.2.3	Ensimmäinen kirjoitustesti .....	39
8.2.4	Toinen kirjoitustesti .....	40
8.3	Verkon mittaukset.....	41
8.3.1	Ensimmäinen verkonmittaus.....	41
8.3.2	Toinen verkonmittaus.....	42
8.4	Prossessorin suorituskykytestit.....	44
8.4.1	Prossessorin suorituskykytestit ensimmäisellä ohjelmalla .....	45
8.4.2	Prossessorin suorituskykytestit toisella ohjelmalla .....	47
8.4.3	Prossessorin suorituskykytestit kolmannella ohjelmalla .....	48
9	YHTEENVETO .....	50
	LÄHTEET.....	52

## LYHENTEET

BIOS	Basic Input/Output System
CAL	Client Access License
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
IOPS	Input Output Per Second
IP	Internet Protocol
IT	Information Technology
KVM	Kernel-based Virtual Machine
LXC	Linux Containers
SFTP	SSH File Transfer Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VE	Virtual Environment
VM	Virtual Machine

## 1 JOHDANTO

Mielenkiintoni heräsi virtualisointia kohtaan, koska siitä on kehkeytynyt eräänlainen muoti-ilmiö tietotekniikassa. Valitsin kyseisen aiheen myös mielenkiinnostani statistiikkoja kohtaan. Tässä opinnäytetyössä pääsen analysoimaan sekä vertailemaan eri virtuaalialustojen suorituskykyä toisiinsa.

Tämän opinnäytetyön tavoitteena oli tutkia kolmea eri virtualisointialustaa ja verrata näiden suorituskykyä. Teoriaosuudessa kerrotaan virtualisoinnista ja kolmen eri virtualisointialustan ominaisuuksista sekä lisensoinnista.

Käytännön vaiheessa fyysiselle palvelimelle asennettiin kolme eri virtualisointialustaa vuoroittain: avoimen lähdekoodin Linux-pohjainen virtualisointialusta nimeltä Proxmox, vSphere-tuoteryhmään kuuluva VMwaren kehittämä ESXi, sekä Microsoftin kehittämä Hyper-V-alusta. Kullekin virtualisointialustalle asennettiin tietty määrä virtuaalilaitteita, jonka jälkeen suoritettiin erilaisia suorituskykytestejä. Tuloksia otettiin ylös prosessorin käytöstä, muistinnopeudesta, kiintolevyn kirjoitus- ja lukunopeuksista, sekä verkon viiveistä ja luotettavuudesta. Tulosten keräyksen jälkeen virtualisointialustoja verrattiin toisiinsa.

Aiheessa keskitytään eniten alustojen suorituskykyjen tasapuoliseen vertailuun. Kaikki virtualisointialustat asennettiin samalle palvelimelle, ja virtuaalikooneita tehtiin tietty määrä samoilla tehoilla. Täten vertailusta saatiin mahdollisimman tarkka ja yksipuolinen.

Opinnäytetyössä käytettiin kvalitatiivista eli laadullista tutkimusmenetelmää. Laadullisen tutkimusmenetelmän tavoitteena on luoda syvälinen ymmärrys jostakin suppeasta aiheesta, tässä tapauksessa virtualisointialustoista. Kvalitatiivinen tutkimusmenetelmä soveltuu hyvin vaihtoehtojen etsimiseen. (Heikkilä 2014.)

## 2 VIRTUALISOINTI

Virtualisointi on prosessi, jolla tietokonejärjestelmän virtuaalinen esiintymä suoritetaan itse fyysisestä laitteistosta otetusta kerroksesta. Tätä kerrosta kutsutaan ohjelmistokerrokseksi. Ohjelmistokerroksessa virtuaalisia laitteita voidaan tarpeen mukaan luoda, poistaa, siirtää ja optimoida. Yleisimmin virtualisoinnilla viitataan tietokoneen-, palvelimen-, tallennustilan-, ohjelmiston- tai tietoverkon virtualisointiin. (Mäntylä 2018.)

### 2.1 Historia

Vaikka virtualisointi on erittäin suosittua nykypäivänä, ei siinä kuitenkaan ole kyse uudesta tekniikasta. Virtualisoinnin alkujuuret yltävät jopa 1960-luvulle asti IBM:n yhteensopivuusongelmien ansiosta. Tuohon aikaan tietokoneet olivat jopa huoneen kokoisia, ja todella kalliita ylläpitää. Yksi laite pystyi suorittamaan vain yhtä ohjelmaa kerralla, jonka takia yhtiö kehitti time-sharing-ominaisuus. Tämä ominaisuus salli useamman ohjelman suorittamisen samanaikaisesti. Tästä keinosta syntyi kuitenkin ongelmia, sillä yhden ohjelman kaatuminen vaikutti myös muihin ohjelmiin. Kyseisen ongelman ehkäisemiseksi alettiin kehittämään virtualisointia, jotta ohjelmille saatiin eristetyt alueet jossa niitä voitiin suorittaa. (Conroy 2018.)

Vuonna 1987 Insignia Solutions esitteli ohjelmiston nimeltä SoftPC, joka salli käyttäjien suorittaa DOS-pohjaisia ohjelmia Unix-pohjaisilla työalustoillaan. Tuona aikana MS DOS -käyttöjärjestelmän suorittamiseen kykenevät tietokoneet maksoivat noin 1 500 dollaria, mutta SoftPC salli Unix-käyttäjien ajaa DOS-pohjaisia ohjelmia vain noin 500 dollarin hintaisilla Unix-pohjaisilla työasemilla. (Hirt 2010, 1; Conroy 2018.)

SoftPC:n suosion johdosta myös muut yritykset hyökkäsivät virtualisointitekniikan pariin. Vuonna 1997 Apple loi VirtualPC-ohjelman, ja se salli käyttäjien suorittaa Windows järjestelmiä Mac-tietokoneilla. Vuonna 1998 luotiin yhtiö nimeltä VMware, ja vuotta myöhemmin ne alkoivat myymään VirtualPC:n tapaista tuotetta nimeltä VMware Workstation. Ensimmäiset versiot toimivat ainoastaan Windows-käyttöjärjestelmissä, mutta myöhemmin yhtiö lisäsi tuen myös muita käyttöjärjestelmiä varten. (Conroy 2018.)



Vuonna 2001 VMWare julkaisi kaksi uutta tuotetta, ESX- ja GSX-serverin. GSX server on hypervisor eli virtualisointialusta, joka sallii käyttäjien käyttää virtuaalisia laitteita olemassa olevan käyttöjärjestelmän kuten Microsoft Windowsin päällä. Kyseistä tekniikkaa kutsutaan tyyppin 2 virtualisointialustaksi. ESX server on tyyppin 1 virtualisointialusta, ja se ei vaadi isäntäkäyttöjärjestelmää suorittaakseen virtuaalilaitteita. (Conroy 2018.)

VMwaren jälkeen useat muutkin yhtiöt hyppäsivät virtualisoinnin maailmaan, jonka seurauksena on tullut paljon muitakin virtualisointialustoja. Näihin kuuluu esimerkiksi tässäkin opinnäytetyössä käytetyt Microsoftin Hyper-V sekä Proxmox.

## **2.2 Palvelimen virtualisointi**

Palvelimen virtualisointi koostuu fyysisestä laitteistosta, joka suorittaa virtualisointialustaa. Virtualisointialusta erottaa fyysisen kerroksen virtuaalilaitteista ja myös esittää fyysisen laitteen tiedot niille. Virtualisointialusta asennetaan suoraan palvelimelle ilman mitään käyttöjärjestelmää alustan ja fyysisten laitteiden välillä. Täten virtualisointialustasta tulee käyttöliittymä fyysisen palvelimen ja virtuaalisten laitteiden välille, ja virtualisointialusta esittää vain osan fyysisen laitteiston resursseista yhdelle virtuaalilaitteelle. Virtualisointialustat eivät toimi ainoastaan virtuaalilaitteiden luomisen alustana, vaan tarjoavat paranneltuja saatavuustoimintoja ja hallintatyökaluja. Virtuaalilaitteita voi kloonata, päivittää, tai jopa siirtää toiselle palvelimelle ilman käyttäjän häirintää. (Portnoy 2012, 15.)

## **2.3 Pöytäkoneen virtualisointi**

Yhdestä tietokoneesta on vaikea saada kaikkia tehoja hyödyllisesti käyttöön ilman virtualisointia. Erilliset tietokoneet vaativat myös henkilökuntaa huolehtimaan päivityksistä sekä tietokoneiden huolloista. Virtuaalityökoneet sijaitsevat datakeskuksien palvelimissa, jotka ovat paljon luotettavampia kuin normaalit tietokoneet. Käyttäjien suorittamat ohjelmistot ovat myös datakeskuksen palvelimissa, joka vähentää verkkoruuhtaa suuresti. Virtuaalikoneita käytetään kevyillä asiakaspäätteillä tai muilla laitteilla, joista suurin osa on halvempia

kuin normaalit tietokoneet. Kevyillä asiakaspäätteillä on pitkä elinikä, ja ne käyttävät vain jopa 10 prosenttia normaalin tietokoneen sähkönkulutuksesta. Jos kevyt asiakaspääte hajoaa, on se helppo vaihtaa omin käsin huoltomiehen sijaan, ja kaikki tiedot ovat tallella datakeskuksessa uutta käyttöä varten. (Portnoy 2012, 15-16.)

## **2.4 Sovellusten virtualisointi**

Myös tietokoneohjelmia tai sovelluksia voidaan virtualisoida. Kuten palvelimen ja työpöydän virtualisoinnissa, tähän on useita erilaisia ratkaisuja. Sovelluksia virtualisoidaan periaatteessa kahdesta syystä, ja ensimmäinen on helppokäyttöisyys. Joidenkin yritysten on hallittava satoja tai jopa tuhansia eri sovelluksia laitteillaan. Aina kun kunkin uuden sovelluksen uusi versio on saatavana, yrityksen täytyy työntää kopio kaikkiin tietokoneisiin, jos he aikovat päivittää sovelluksen. Yhdelle tietokoneelle tämä voi olla pieni vaiva, mutta vaikka tuhannelle tietokoneelle työ on aikaa vaativaa ja toistuvaa. Tätä varten yritysten IT-henkilökunnalla on työkaluja, jotka auttavat hallitsemaan ja automatisoimaan päivityksen, jotta tehtävä tapahtuu luotettavasti ja vaivatta. (Portnoy 2012, 17.)

Toinen syy liittyy siihen, miten erilaiset sovellukset toimivat yhdessä muiden kanssa. Yhden sovelluksen lataaminen tai päivittäminen voi rikkoa toisen sovelluksen, joka on toiminut aiemmin normaalisti. Jotkin sovellusten virtualisointityypit voivat lieventää tai ehkäistä kyseistä ongelmaa eristämällä koko ohjelman sekä prosessin. Tämä on nopeasti kehittyvä tekniikka, jolle esiintyy säännöllisesti uusia käyttötapauksia etenkin mobiililaitteiden kanssa. (Portnoy 2012, 17.)

## **2.5 Hyödyt**

Virtualisointi parantaa fyysisten laitteiden käyttöastetta ja nopeuttaa niiden käyttöönottoja. Palvelinhuoneissa säästyy paljon ylimääräistä tilaa virtualisoinnin vuoksi, ja se laskee myös virrankulutusta sekä nopeuttaa ongelmatilanteista toipumista esimerkiksi kloonauksen tai tilannekaappausten avulla. Virtualisointi voi myös tuoda joustavuutta tietohallintoon ja auttaa sitä vastaamaan liiketoiminnan vaatimuksiin. (Mäntylä 2018.)

Ennen virtualisoinnin yleistymistä tietokonelaitteistojen resursseja käytettiin keskimäärin vain muutaman kymmenen prosentin käyttöasteella, jonka seurauksena suurin osa tietokoneen tehoista jää hyödyntämättä. Tässä virtualisointi tulee hyödyksi, sillä vähäisestikin hyödynnetty tietokone voi viedä tilaa ja sähköä melkein saman verran kuin täysin hyödynnetty laitteisto. (Golden 2007, 4.)

IT-organisaatioiden on käsiteltävä ongelmia päivittäin hienostuneiden infrastruktuurien takia. Tietokoneiden komponentit hajoavat, käyttöjärjestelmät ylikuormittuvat, ja ohjelmat kaatuvat pakottaen uudelleenkäynnistykseen. Tämän takia ongelmien ylläpito voi olla itseään toistavaa työtä. Jos jokin tärkeä palvelin tai tallennustila hajoaa, korjaaminen voi viedä aikaa tunneista jopa päiviin. Virtualisointi voi auttaa laitteisto-ongelmien, ohjelmistojen tai käyttöjärjestelmien huoltamisessa. Virtualisointiohjelmistot voidaan konfiguroida tarkkailemaan virtuaalilaitteita esimerkiksi niin, että jos yksi virtuaalilaitte kaatuu, voi ohjelmisto käynnistää korvaavan virtuaalilaitteen samalla tai jopa eri laitteistolla. (Golden 2007, 6.)

Ennen virtualisointia palvelimien, tallennustilojen, ja tietokoneiden ylläpito sekä päivittäminen oli haastavaa ja aikaa vaativaa. Koska tuotantolaitteita ei voida ottaa pois käytöstä, huolto- ja päivitystoimenpiteet tehtiin yleensä työntuntien ulkopuolella. Joskus tärkeät huoltotoimenpiteet jopa ohitettiin työn vaikean aikataulutuksen takia, jotka voivat johtaa laitteisto- tai ohjelmistovikoihin. Virtualisointi voi tehdä huoltotoimenpiteistä paljon helpompaa ja edullisempaa, sillä virtuaalilaitteen voi siirtää palvelimelta toiselle, jotta huoltotoimenpiteitä vaativa palvelin vapautuu huoltoa varten. Virtuaalilaitteiden päivitykset ja korjaukset voidaan myös keskittää, sillä laitteiden ohjelmistot voidaan säilyttää esimerkiksi datakeskuksen tallennustilassa itse virtuaalilaitteiden sijaan. (Golden 2007, 7.)

Virtualisointi voi nostaa tietotekniikkaorganisaatioiden tarjoamaa palvelun laatua, jos sitä hallinnoidaan tehokkaasti. Aikataulutettuja varmuuskopioita toteuttamalla tietotekniikkaorganisaatiot voivat välttää häiriötekijöitä tai koordinoimattomia toimintaa. Lisäämällä palvelun laatua, tietotekniikkaorganisaatiot voivat myös varmistaa, että asiakkaille toimitetut yrityspalvelut suoritetaan tehokkaasti ja johdonmukaisesti. (Golden 2007, 7.)

Virtualisointi voi myös parantaa palvelujen saatavuutta yritystoiminnassa. Kun vastaan tulee katkoksia esimerkiksi laitteiston tai ohjelmiston takia, virtualisointi auttaa ehkäisemään suunniteltuja sekä suunnittelemattomia katkoksia, sillä käynnissä olevia virtuaalipalvelimia voidaan siirtää ongelmalaitteistosta toiseen. (Golden 2007, 8.)

Ilmaston lämpeneminen on yksi nykypäivän suuria haasteita, ja sen ehkäisemisen tietoisuutta yritetään jakaa ympäri planeettaa. Energian käytön vähentäminen voi vähentää myös kuluja ja se on tärkeää nykypäivän yrityksille. Tässä virtualisointi tulee hyödyksi, sillä se on yksi tekniikoista, joka voi vähentää suuresti energian käyttöä. Tuhansien vähäisen käytön alaisena olevien tietokoneiden ja kiintolevyjen sijaan voidaan virtualisoinnin avulla vähentää käytettyjen tietokoneiden ja kiintolevyjen määrää moninkertaisesti. Jopa paremman hyödyn saa, jos käytetään palvelimia, jotka ovat suunniteltuja palvelemaan virtualisointialustoina. Virtualisointi vähentää myös tarvetta valmistaa valtavia määriä uusia komponentteja. (Golden 2007, 10.)

## **2.6 Virtualisointialusta**

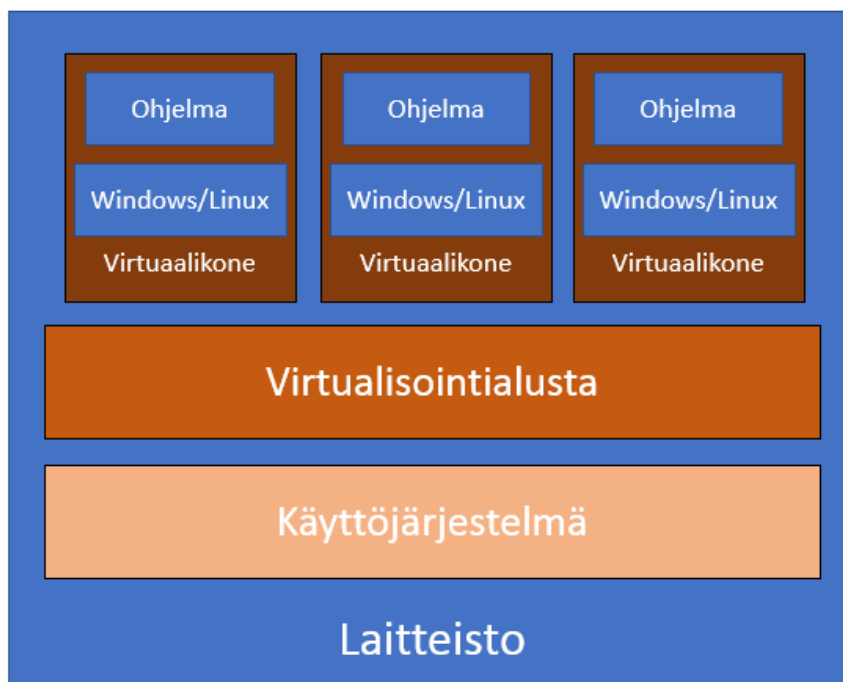
Virtualisointialusta mahdollistaa useamman käyttöjärjestelmän suorittamiseen yksittäisellä fyysisellä laitteella. Virtualisointialustan päätehtävä on tuottaa eristetyn suoritusympäristön jokaiselle virtuaalilaitteelle, sekä hallita kommunikointia virtuaalilaitteiden ja fyysisen tietokoneen välillä. (Tulloch 2010, 23.)

Ensimmäinen virtualisointialusta näki päivänvalon vuonna 1972, kun IBM päävitti System/370-sarjan suurtietokoneet tukemaan virtualisointia. Virtualisointialustan luominen oli tietojenkäsittelyn merkkipaalu, sillä se antoi ratkaisun suurtietokoneiden arkkitehtuurisille rajoitteille sekä suurelle käyttö hinnalle. Virtualisointialustoja on monentyyppisiä ja niitä voi kategorisoida esimerkiksi tyyppin mukaan siten, että pyörivätkö ne suoraan fyysisellä laitteella vai pyörivätkö ne käyttöjärjestelmän isännöimänä. (Tulloch 2010, 23.)

### 2.6.1 Tyypin 2 virtualisointialusta

Kun virtualisointialustaa suoritetaan isäntäkäyttöjärjestelmän alaisuudessa, tätä kutsutaan tyypin 2 virtualisointialustaksi. Tietokoneeseen asennetaan ensin käyttöjärjestelmä, ja tämän päälle luodaan virtualisointialusta virtuaalisia laitteita varten. (Portnoy 2012, 23 - 24; Tulloch 2010, 24.)

Koska tietokoneen tehoja kuluu myös isäntäkäyttöjärjestelmän suoritukseen sekä isäntäkäyttöjärjestelmän ja virtuaalilaitteen välillä suorittamiin muunnoksiin, tämä tapa ei ole yhtä tehokas kuin tyypin 1 virtualisointialusta. Peruskäyttäjälle tämä on kuitenkin helpompi tapa luoda virtuaalilaitteita, ja tätä keinoa hyödynnetään enemmän yksityiseen käyttöön. Esimerkiksi VMware workstation on tyypin 2 virtualisointialustoja. Diagrammi virtualisointialustan toiminnasta on nähtävissä kuvassa 1. (Portnoy 2012, 23 - 24; Tulloch 2010, 24.)



Kuva 1. Tyypin 2 virtualisointialusta (Portnoy 2012, 23.)

### 2.6.2 Tyypin 1 virtualisointialusta

Tässä menetelmässä laitteiston ja virtualisointialustan välissä ei ole isäntäkäyttöjärjestelmää, vaan virtualisointialustan omaava käyttöjärjestelmä asennetaan suoraan fyysiselle alustalle. Virtualisointialusta toimii siis suoraan laitteiston päällä. Tässä menetelmässä virtuaalisoinnissa tarvittavat pyynnöt eivät

kulje isäntäkäyttöjärjestelmän kautta, vaan virtualisointialusta kommunikoi suoraan laitteistolle. (Hämäläinen 2007; Portnoy 2012, 21 - 22; Tulloch 2010, 23.)

Myös bare-metal-virtualisoinniksi kutsutun virtualisointialustan edut ovat suorituskyvyn, turvallisuuden ja käyttövarmuuden lisäksi myös yhteensopivuusongelmien vähäisyys, sillä virtuaalilaitteilla on laitteiston välissä ainoastaan virtualisointialusta. Tämä tekniikka kiinnostaa erityisesti raskaan sarjan palvelinohjelmistojen valmistajia, koska kirjoittamalla sovelluksensa suoraan virtualisointialustalle ne välttyvät palvelinkäyttöjärjestelmän lisensoinnilta. Kyseistä menetelmää käytetään enemmän esimerkiksi datakeskuksissa. Esimerkiksi Microsoft Hyper-V ja VMware ESXi ovat tyyppin 1 virtualisointialustoja. Diagrammi virtualisointialustan toiminnasta on nähtävissä kuvassa 2. (Hämäläinen 2007; Portnoy 2012, 21 - 22; Tulloch 2010, 23.)



Kuva 2. Tyyppin 1 virtualisointialusta (Portnoy 2012, 22.)

### 3 WINDOWS SERVER 2016 JA HYPER-V

Windows Server 2016 on Microsoftin valmistama palvelinkäyttöjärjestelmä, jota kehitettiin samaan aikaan Windows 10 -käyttöjärjestelmän kanssa. Käyttöjärjestelmä julkaistiin 12. lokakuuta vuonna 2016, ja päivityksiä on tullut tasaiseen tahtiin siitä asti. Windows Server käyttää Hyper-V-virtualisointialustaa,

jolla voi luoda ja ylläpitää erillisiä virtuaalilaitteita. Windows Server 2016:een on tullut paljon uusia ja päivitettyjä ominaisuuksia etenkin tietoverkon, tallennustilan ja Hyper-V:n osa-alueille. Windows Server 2016:n käyttöliittymä muistuttaa hyvin paljon Windows 10:ä, jonka vuoksi se on selkeä ja helppo käyttää. (Serre 2016.)

### 3.1 Laitevaatimukset

Windows Server 2016:n laitevaatimukset ovat samat Windows Server 2012 kanssa, vaikka uusia ominaisuuksia onkin paljon:

- 1.4GHz 64-bittinen prosessori
- 512 MB keskusmuistia (2GB Desktop Experiencen kanssa)
- Vähintään 32 GB vapaata levytilaa

(Microsoft 2017.)

### 3.2 Hyper-V

Microsoft on ottanut johtoaseman virtualisointiteknologian kehityksessä Hyper-V:n ansiosta. Hyper-V esiteltiin ensimmäistä kertaa osana Windows Server 2008:a, jonka jälkeen laajennetuissa ja parannelluissa Windows Server 2008 R2:ta sekä Windows Server 2012:ta. Hyper-V tarjoaa organisaatioille työkalun optimoida palvelinlaitteistojen sijoitusomaisuuksia yhdistämällä useita palvelinrooleja erillisinä virtuaalilaitteina yhdelle fyysiselle palvelimelle. Organisaatiot voivat myös käyttää Hyper-V:tä suorittamaan useita käyttöjärjestelmiä yhdellä palvelimella. (Microsoft 2013.)

Windows Server 2016 toi Hyper-V:lle uusia ominaisuuksia, joita ei aikaisemmissa versioissa ole. Yksi tärkeistä ominaisuuksista on tuotannon tarkastuspisteet, jotka sallivat tehdä tarkastuspisteitä jokaiselle työkuormalle. Toinen hyödyllinen ominaisuus on VHDS. Tämä ominaisuus on ainoastaan Windows Server 2016:n tukema, ja se mahdollistaa virtuaalisen kiintolevyn jakamisen usealle virtuaalilaitteelle. Tämä on uusi tapa implementoida vierasklusteri jae-  
tuilla tallennustiloilla Windows Server 2016:ssa. Hyper-V voi nyt myös antaa

virtuaalilaitteelle täyden pääsyn PCI-E-väylässä olevalle laitteelle, kuten näyttöohjaimelle. Tämä ominaisuus parantaa suorituskykyä ohittamalla virtualisointipinon. (Serre 2016.)

### 3.3 LISENSOINTI

Windows Server 2016 versiosta on kolme eri lisensointimallia: Datacenter-lisenssi raskaasti virtualisoidulle datakeskusympäristöille, Standard-lisenssi vähän virtualisoidulle tai virtualisoimattomalle ympäristölle, ja Essentials-lisenssi pienille yrityksille joilla on alle 25 käyttäjää ja 50 laitetta. Essentials versiolla ei ole virtualisointioikeuksia ja se on rajoitettu kahdelle suorittimelle. Essentials ei tarvitse erillisiä CAL-lisenssejä. (Moonsoft s.a.)

Datacenter- ja Standard-versiot lisensoidaan ydinten pohjalta, kun taas Essentials-versio lisensoidaan prosessorin pohjalta. Tämä perustuu siihen, että nykypäivänä voidaan ostaa Intelin Xeon-prosessori, jossa on 24 ydintä. Jos Microsoft myisi lisenssejä prosessoreille ydinten sijaan, menettäisi Microsoft suuren osan voitoista isojen yhtiöiden takia. Ennen yhtiöiden olisi tarvinnut maksaa kahdesta 24-ytimisestä prosessorista vain yhden lisenssin. Datacenter- ja Standard -versiot ovat myös CAL-lisenssoituja. Kuvassa 3 on lueteltuna Windows Server 2016:n lisenssien hintaerot. Datacenter- ja Standard-lisenssit ovat 16 ytimelle. (Finn 2016; Pricing and licensing for Windows Server 2016 s.a.)

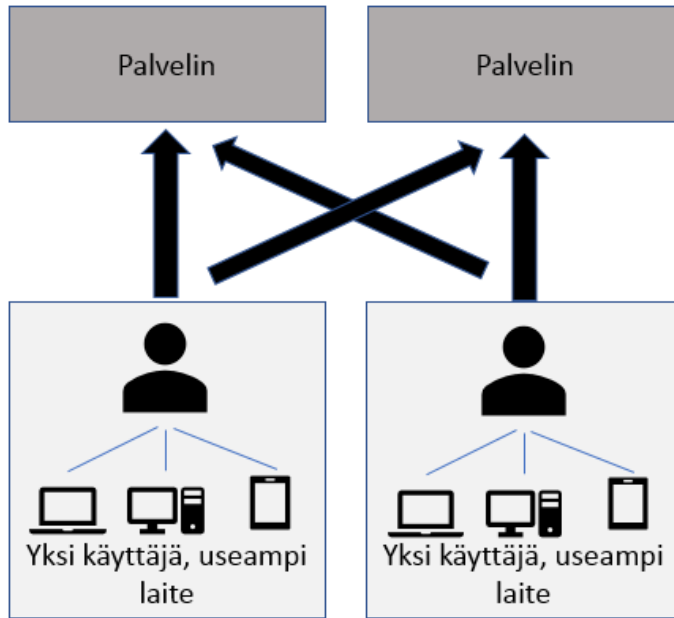
<b>Windows Server 2016 Datacenter</b>	\$ 6 155,00
<b>Windows Server 2016 Standard</b>	\$ 882,00
<b>Windows Server 2016 Essentials</b>	\$ 501,00

Kuva 3. Lisenssien hintaerot (Pricing and licensing for Windows Server 2016 s.a.)

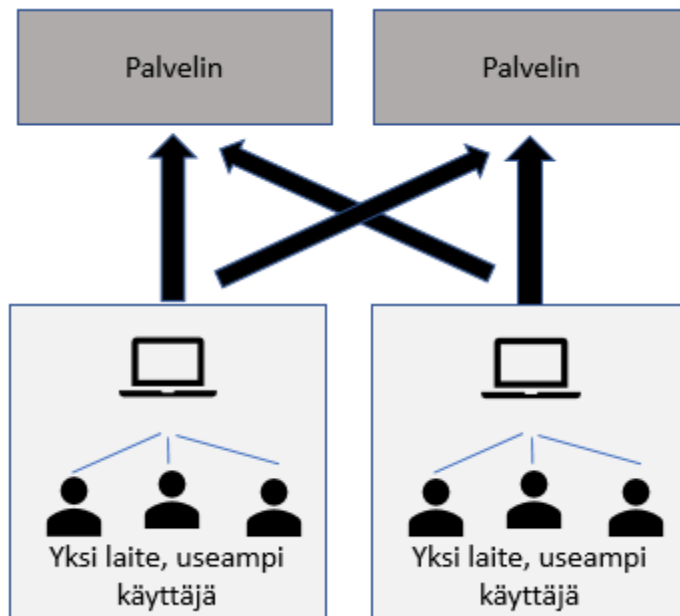
CAL eli Client Access License on lisenssi, joka antaa käyttäjälle tai laitteelle pääsyn palvelimen ominaisuuksiin. Yhtiö voi tarpeen mukaan hankkia Client



Access Licensen joko käyttäjälle tai laitteelle. Käyttäjälisenssillä voi yksi käyttäjä hankkia pääsyn palvelimelle usealla laitteella, ja laitelisenssillä voi useampi käyttäjä hallita palvelinta yhdeltä tietyltä laitteelta. Kuvissa 4 ja 5 havainnollistetaan näiden kahden lisenssin eroavaisuus. (Client access licenses and management licenses s.a.)



Kuva 4. Käyttäjän Client Access License (Client access licenses and management licenses s.a.)



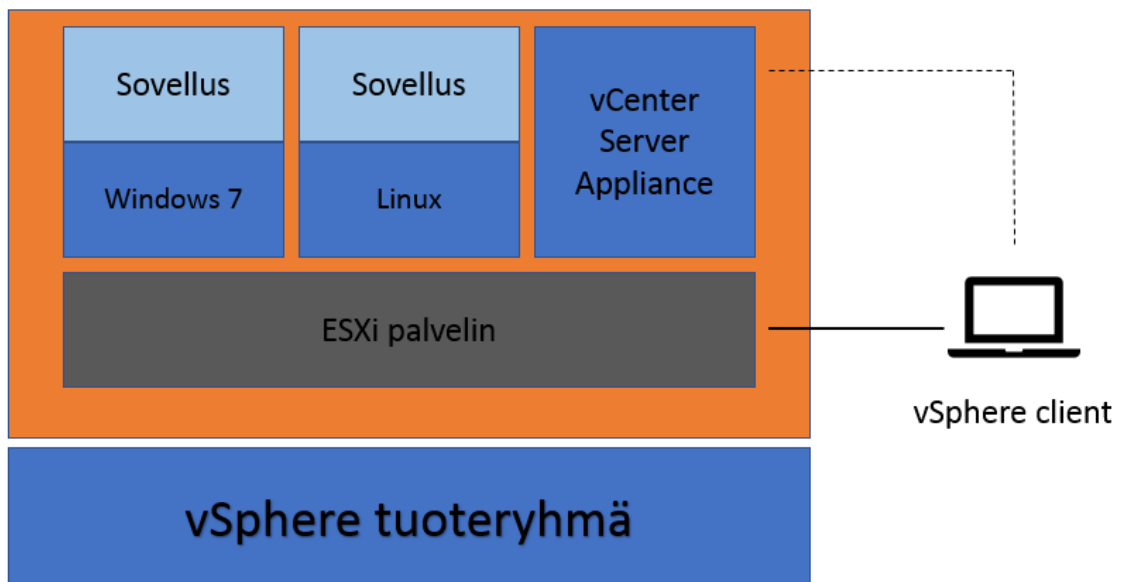
Kuva 5. Laitteen Client Access (Client access licenses and management licenses s.a.)

## 4 VSPHERE JA ESXI

Vsphere on VMwaren valmistama tuoteryhmä, johon sisältyy esimerkiksi vSphere client, vCenter ja ESXi. Aiemmin VMware Infrastructurena kutsuttu paketti julkaistiin ensimmäistä kertaa vSphere-nimellä 21. toukokuuta 2009, jonka jälkeen tuoteryhmälle on tullut päivityksiä tasaiseen tahtiin. Yksi merkittävimmistä päivityksistä oli tuen lisääminen Windows 7 ja Windows Server 2008 R2:ta varten. (vSphere-land 2009; VMware 2009.)

### 4.1 ESXi

ESXi on VMwaren valmistama virtualisointialusta, ja se asennetaan suoraan fyysiselle alustalle, joka tekee siitä tyypin 1 virtualisointialustan. Kaikki virtuaalilaitteet asennetaan ESXi-palvelimelle, ja niiden asentamiseen ja ylläpitämiseen tarvitaan vSphere clientin tai vCenterin. Kyseinen vSphere client asennetaan esimerkiksi kannettavalle tietokoneelle, ja sillä muodostetaan yhteys ESXi-palvelimeen, jotta käyttäjä voi hallita palvelinta. Kuva 6 visualisoi koko vSphere-tuoteryhmän. (Giri 2012.)



Kuva 6. vSphere tuoteryhmä. (Giri 2012.)

Käytettävyydeltään vCenter on lähes samanlainen kuin vSphere client, mutta se asennetaan palvelimelle, ja siinä on enemmän tehoa. Fyysisen palvelimen sijaan vCenter voidaan asentaa myös virtuaalisesti ESXi:n päälle. VMware

ESXi 5.5:n jälkeen vCenteriä ei ole tarvinnut enää asentaa fyysiselle tai virtuaaliselle palvelimelle vCenter Server Appliancen ansiosta. vCSA on valmiiksi konfiguroitu Linux-virtuaalilaite, joka on optimoitu VMware vCenter Serverin sekä siihen liittyvien palvelujen suorittamiseen. (Giri 2012; VMware 2018.)

## 4.2 LISENSOINTI

VMware tarjoaa tällä hetkellä suuremmille yhtiöille kolmea eri vSphere-lisensivaihtoehtoa: vSphere Standard, vSphere Enterprise Plus ja vSphere with Operations Management Enterprise Plus. vSphere Standard on aloitustason lisenssi, joka on suunniteltu tarjoamaan palvelimen vahvistamista sekä liiketoiminnan jatkuvuutta. vSphere Enterprise Plus on vankempi kuin vSphere Standard, ja se tarjoaa resurssienhallinnan sekä parannetun sovellusten saatavuuden ja suorituskyvyn. vSphere with Operations Management Enterprise Plus tarjoaa täyden valikoiman vSphere-ominaisuuksia, kuten datakeskuksen optimoinnin älykkäillä toiminnoilla sekä johdonmukaisen hallinnan ja automaation ennakoivan analyysin avulla. (Rouse 2017.)

VMware tarjosi aiemmin myös vSphere Enterprise -version, vSphere Standard with Operations Management -version, sekä vSphere Enterprise with Operations Management -version, mutta näiden versioiden elämänsykli päättyi kesäkuussa 2017. Näille versioille on kuitenkin luvattu aktiivinen tuki vuoteen 2021 asti. (Rouse 2017.)

VMware tarjoaa kaksi vSphere-lisenssiä pienemmille ympäristöille, vSphere Essentials Kitin ja vSphere Essentials Plus Kitin. vSphere Essentials Kit sisältää vSphere-virtualisointialustan ja sallii järjestelmänvalvojan hallita sekä vahvistaa ohjelmia vähentääkseen laitteisto- ja suoritusmaksuja. vSphere Essentials Plus Kit tarjoaa laajemman valikoiman ominaisuuksia. (Rouse 2017.)

Haara- sekä etäkonttoreja varten on myös kaksi vSphere-lisenssiä: vSphere Remote Office Branch Office (ROBO) Standard sekä vSphere ROBO Advanced. vCenter Serverille on myös kaksi eri lisenssiä: vCenter Server Foundation pienemmille ympäristöille sekä vCenter Server Standard laajemmille ympäristöille. Kuvassa 7 vertaamme eri lisenssihintoja. (Rouse 2017.)

<b>Vmware vSphere versiot</b>	Vmware vSphere Standard	\$ 995,00
	Vmware vSphere Enterprise Plus	\$ 3 495,00
	Vmware vSphere with Operations Management Enterprise Plus	\$ 4 395,00
<b>Vmware vSphere ROBO versiot</b>	Vmware vSphere ROBO Standard	\$ 3 000,00
	Vmware vSphere ROBO Advanced	\$ 4 495,00
<b>Vmware vSphere Essentials paketit</b>	Vmware vSphere Essentials Kit	\$ 495,00
	Vmware vSphere Essentials Plus Kit	\$ 4 495,00
<b>vCenter Server</b>	Vmware vCenter Server Foundation	\$ 1 495,00
	Vmware vCenter Server Standard	\$ 5 995,00

Kuva 7. VMwaren eri lisenssien hintaluokkia. (Rouse 2017.)

### 4.3 Laitevaatimukset

- vähintään 2 ytiminen 64-bit x86 prosessori
- vähintään 4GB keskusmuistia
- 8 GB kiintolevytilaa

Yllä olevassa luetelmassa on listattuna VMware ESXi:n laitevaatimukset. Virtuaalikoneiden suorittamiseen normaalissa ympäristössä suositellaan kuitenkin vähintään 8 GB keskusmuistia.

## 5 PROXMOX

Proxmox VE on avoimen lähdekoodin palvelinvirtualisoinnin hallintaohjelmisto, joka on rakennettu KVM:n, QEMU:n ja LXC:n ympärille. Proxmox VE:llä voi hallita virtuaalilaitteita, klustereita, tallennustilaa ja tietoverkkoja helppokäyttöi-

sellä integroidulla web-käyttöliittymällä tai komentorivillä. Proxmox VE on lisensoitu GNU AGPL versio 3:n mukaan, ja projektia kehittää ja ylläpitää Proxmox Server Solutions GmbH. (Proxmox 2018.)

## 5.1 KVM

KVM (Kernel-based Virtual Machine) on täydellisen virtualisoinnin tuki Linux-käyttöjärjestelmille. Se on alun perin Qumranetin kehittämä, mutta Redhat hankki Qumranetin omistukseensa lokakuussa 2008. Qumranet julkaisi KVM:n koodin avoimen lähdekoodin yhteisölle. Versiosta 2.6.20 lähtien KVM on ollut osana tavallista Linux-ydintä. (Hirt 2010.)

## 5.2 Laitevaatimukset

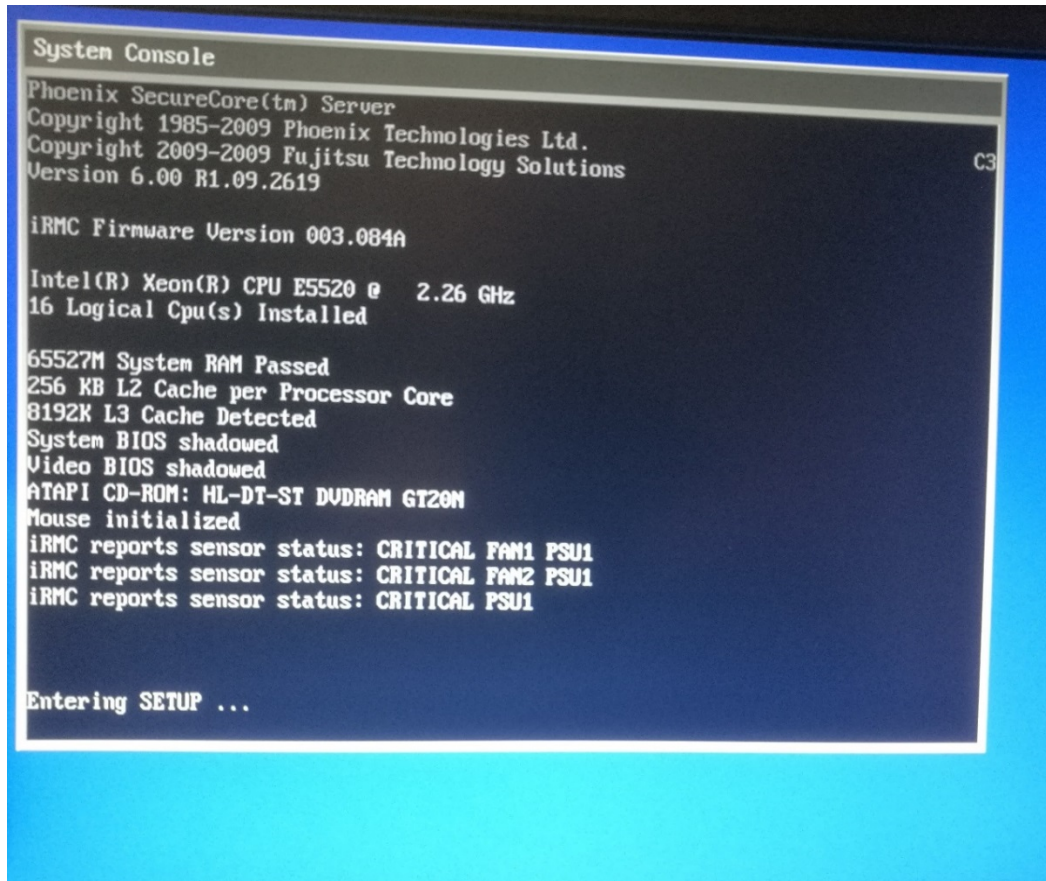
- 64-bit (Intel EMT64 tai AMD64) moniytiminen prosessori
- 8 GB keskusmuistia
- 16GB kiintolevytilaa

Yllä olevassa luetelmassa on listattuna Proxmoxin laitevaatimukset. Proxmox toimii hyvin vähäisellä keskusmuistimäärällä, mutta suositeltu määrä on 8GB keskusmuistia.

## 6 LAITTEISTO

Tässä opinnäytetyössä käytettiin kaikissa asennuksissa Fujitsu Technology Solutionsin Primergy RS300 S5 -palvelinta. Saman palvelimen käyttö samoilla teknisillä tiedoilla on todella tärkeä osa suorituskykyvertailua. Kyseisen palvelimen tekniset tiedot olivat:

- kaksi kappaletta Intel Xeon E5520 2.26GHz prosessoria
- 64 GB 800 MHz DDR3 keskusmuistia
- 1672 GB tallennustilaa



Kuva 8. Palvelimen POST-screen

Kuvassa 8 on nähtävissä palvelimen POST-diagnoosi eli power-on self-test -diagnoosityökalu, joka käy läpi tietyt komponentit ja ilmoittaa virheistä, jos niitä on. Tässä tapauksessa todettiin, että diagnoosi huomasi jotain ongelmia muutamien tuulettimen sensorin ja itse virtalähteen kanssa. Varoituksesta huolimatta palvelin vaikutti toimivan normaalisti ja opinnäytetyötä jatkettiin normaalisti. Kytkimenä käytettiin Cisco Systemsin Catalyst 2960 24-porttista kytkintä.

## 7 VIRTUALISOINTIALUSTOJEN ASENNUKSET JA VALMISTELU

Virtualisointialustat asennettiin samalle palvelimelle, joten alustoja ei voinut pitää käynnissä samanaikaisesti. Kaikkien alustojen asennus oli helppoa ja yksinkertaista. Windows Hyper-V:n asennuksessa kuitenkin vaadittiin hieman enemmän asennusvaiheita itse virtualisointialustan toimimiseen, sillä se asennettiin erikseen Server Manager -ohjelmasta. Erikseen asennettavia paravirtualisointityökaluja ei käytetty tässä opinnäytetyössä.

Kaikkien virtualisointialustojen virtuaalilaitteissa käytettiin samoja asetuksia. Windows 10 -käyttöjärjestelmässä käytettiin seuraavia asetuksia:

- prosessori: 4 ydintä (2 suoritinkantaa, 2 ydintä)
- keskusmuisti: 8 GB
- kiintolevytila: 32 GB

Fedora Linuxissa käytettiin vähemmän laitteistoresursseja:

- prosessori: 4 ydintä (2 suoritinkantaa, 2 ydintä)
- keskusmuisti: 2 GB
- kiintolevytila: 16 GB

## 7.1 Proxmox-asennus ja -valmistelu

Proxmoxin asennus oli helppo ja nopea suorittaa, ja asennuksessa oli vähän vaiheita. Asennuksen aloittaessa voitiin valita joko tavallinen asennus tai vika-sietotilassa suoritettava asennus. Tämä asennusmuoto avasi shell-konsolin useissa asennusvaiheissa, jotta asennuksen ohella voitiin korjata virheitä niitä ilmetessä. Tätä asennusmuotoa kuitenkin suositeltiin vain kehittäjille, ja sitä ei ole tarkoitettu yleiseen käyttöön.

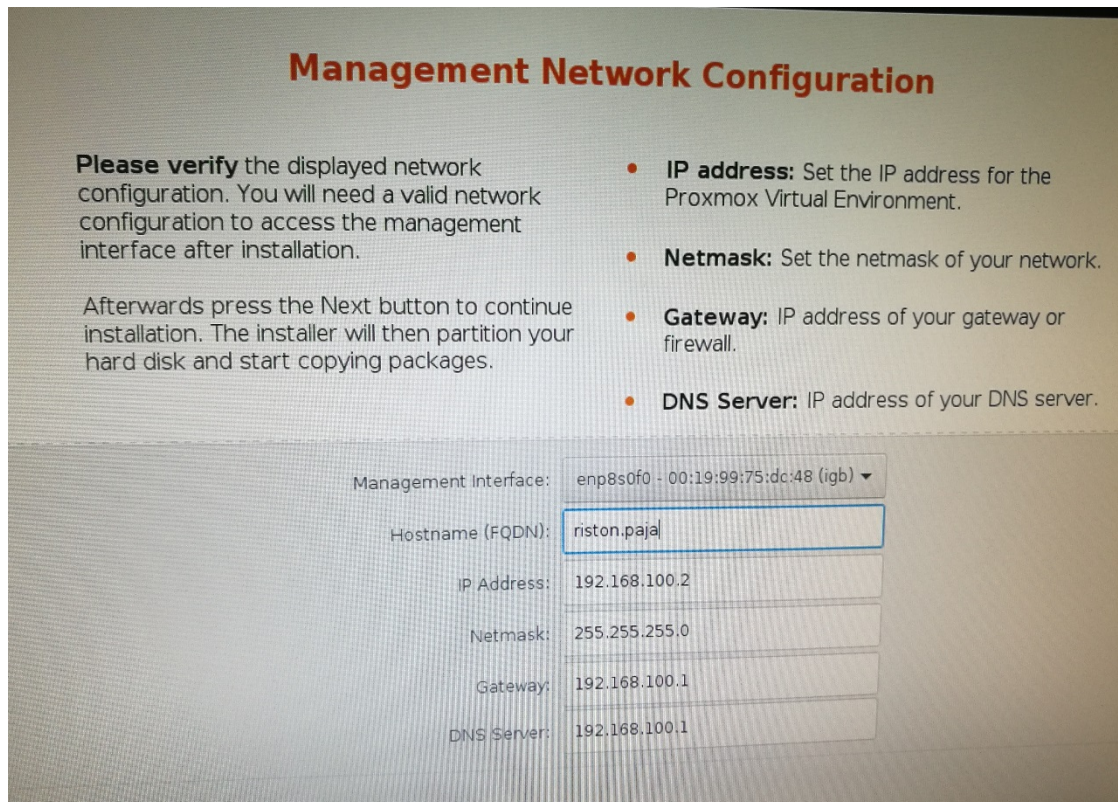
Näiden lisäksi valikosta voitiin valita myös Rescue Boot -vaihtoehto, joka mahdollisti olemassa olevan asennuksen käynnistämisen. Tämä vaihtoehto etsi kaikki kiintolevyt ja jos se löysi olemassa olevan asennuksen, voitiin se käynnistää suoraan olemassa olevan Linux-ytimen avulla. Tämä keino voi olla hyödyllinen jos käynnistyslohkossa on ongelmia, tai BIOS ei pysty lukemaan käynnistyslohkoa levytä. Lisäksi asentaja pystyi valitsemaan Test Memory -vaihtoehdon, joka ajaa Memtest86+-ohjelman. Tämä ohjelma on kätevä, jos asentaja haluaa tarkistaa onko palvelimen keskusmuisti virheettömässä kunnossa.

Seuraavaksi asennuksessa valittiin kiintolevy, jolle Proxmox haluttiin asentaa, asennuskieli ja aikavyöhyke, sekä salasana ja sähköpostiosoite. Maan valintaa käytettiin valitsemaan läheisiä peilipalvelimia, joka nopeutti latauksia sekä teki päivityksistä luotettavampia.



Asennuksen seuraavassa vaiheessa alustalle syötettiin hallintaverkon asetukset. Asetukset ovat nähtävissä kuvassa 9:

- Isäntänimi: riston.paja
- IP-osoite: 192.168.100.2
- Aliverkon peite: 255.255.255.0
- Yhdyskäytävä: 192.168.100.1
- DNS-palvelin: 192.168.100.1



Kuva 9. Hallintaverkon asetukset

Tämän jälkeen Proxmox oli asennettu ja sen käyttö voitiin aloittaa. Proxmoxin komentorivi ehdotti käyttäjää yhdistämään palvelimeen verkkoselaimella. Samassa verkossa olevan tietokoneen verkkoselaimeen syötettiin palvelimen IP-osoite käyttäen porttia 8006. Kuvassa 10 on nähtävissä Proxmox virtualisointialustan komentorivi.



```

-----
Welcome to the Proxmox Virtual Environment. Please use your web browser to
configure this server - connect to:

https://192.168.100.2:8006/

-----

riston login: root
Password:
Linux riston 4.13.13-2-pve #1 SMP PVE 4.13.13-32 (Thu, 21 Dec 2017 09:02:14 +0100)

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

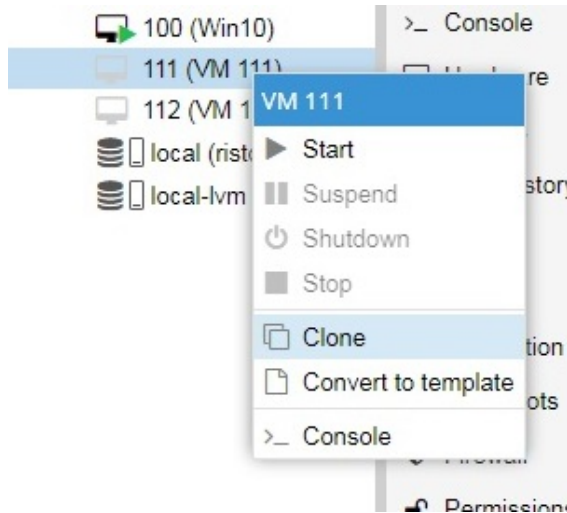
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@riston:~#
root@riston:~#
root@riston:~#
root@riston:~#
root@riston:~#

```

Kuva 10. Proxmoxin komentorivi

Kun samassa verkossa olevaan tietokoneeseen oli syötetty palvelimen osoite, voitiin aloittaa virtuaalilaitteiden asennus. Proxmoxilla virtuaalilaitteiden asennus oli hyvin yksinkertaista nopean ja helpon valikon takia. Ennen virtuaalilaitteiden asentamista käyttöjärjestelmien levykuvat siirrettiin palvelimelle SFTP-protokollaa käyttäen. Tämän jälkeen käyttöliittymän oikeasta yläkulmasta painettiin Create VM -painiketta, jonka jälkeen alustalle syötettiin virtuaalilaitteen tiedot. Koska Proxmoxissa on virtuaalilaitteen kloonaustyökalu, alustalle asennettiin vain yhdet kappaleet Windows 10- sekä Linux-käyttöjärjestelmiä edellä mainituilla laitteistoresursseilla.

Proxmoxilla virtuaalilaitteiden kloonaus on nopeaa ja yksinkertaista. Vasemalla olevasta listasta klikattiin haluttua virtuaalilaitetta hiiren oikealla, jonka jälkeen pudotusvalikosta valittiin Clone-työkalu. Kuvassa 11 on nähtävissä virtuaalilaitteen pudotusvalikko, josta Clone-työkalu valittiin. Valittuaan työkalun käyttäjän piti vain valita kohde, jonka alle virtuaalilaitte asennetaan, virtuaalilaitteen ID, virtuaalilaitteen nimi, sekä osio, jolle virtuaalilaitte haluttiin asentaa. Tässä opinnäytetyössä käytettiin vain yhtä osiota sekä kohdetta, joten näitä ei tarvinnut valita. Virtuaalilaitteet oli mahdollista kloonata käynnissä olevasta virtuaalilaitteesta, mutta näin prosessi oli hitaampi.



Kuva 11. Virtuaalilaitteen pudotusvalikko, josta virtuaalilaitteet kloonataan

Virtuaalilaitteita kloonattiin ensin niin, että Windows- sekä Linux-laitteita oli viisi kappaletta kumpaakin, eli kymmenen laitetta yhteensä. Suorituskykytestien jälkeen virtuaalikoneiden määrää nostettiin kloonamalla yhteensä kahteenkymmeneen.

## 7.2 Windows Server 2016 sekä Hyper-V-asennus ja -valmistelu

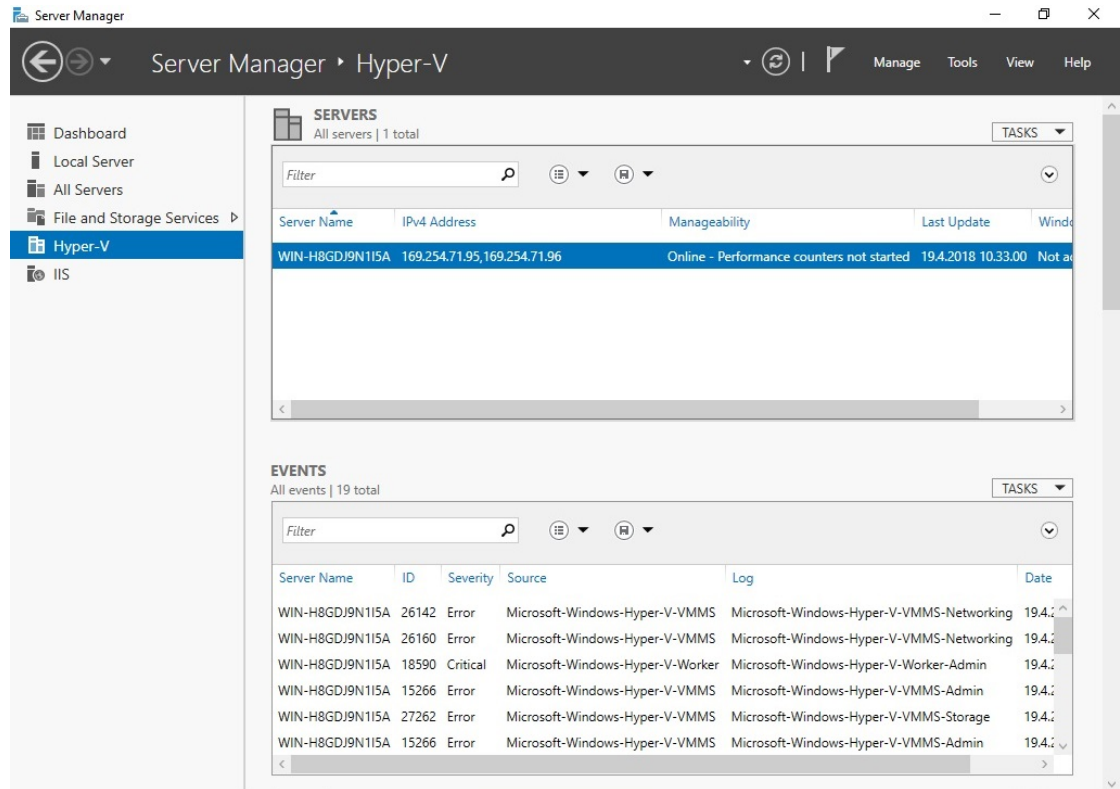
Hyper-V:n käyttöönotto aloitettiin asentamalla Windows Server 2016. Windows Server 2016 on yhtä yksinkertainen asentaa kuin muutkin nykypäivän Windows-tuotteet. Asennus kysyi vain tuotekoodin, jonka jälkeen asennus alkoi. Palvelimen muut asetukset suoritetaan itse käyttöjärjestelmän sisällä.

Käyttöjärjestelmälle syötettiin ensin IP-osoite:

- IP-Osoite: 169.254.71.96
- Aliverkon peite: 255.255.255.0
- Yhdyskäytävä: 169.254.71.95

Tämän jälkeen palvelimelle asennettiin Hyper-V virtualisointialusta. Server managerista valittiin Add roles and features -ominaisuus, joka sisälsi useita eri palvelinrooleja, kuten DNS- ja DHCP-palvelimen. Valikosta valittiin Hyper-V, jonka jälkeen asennus kysyy verkkoadapteria, jota virtuaalinen kytkin käyttää.

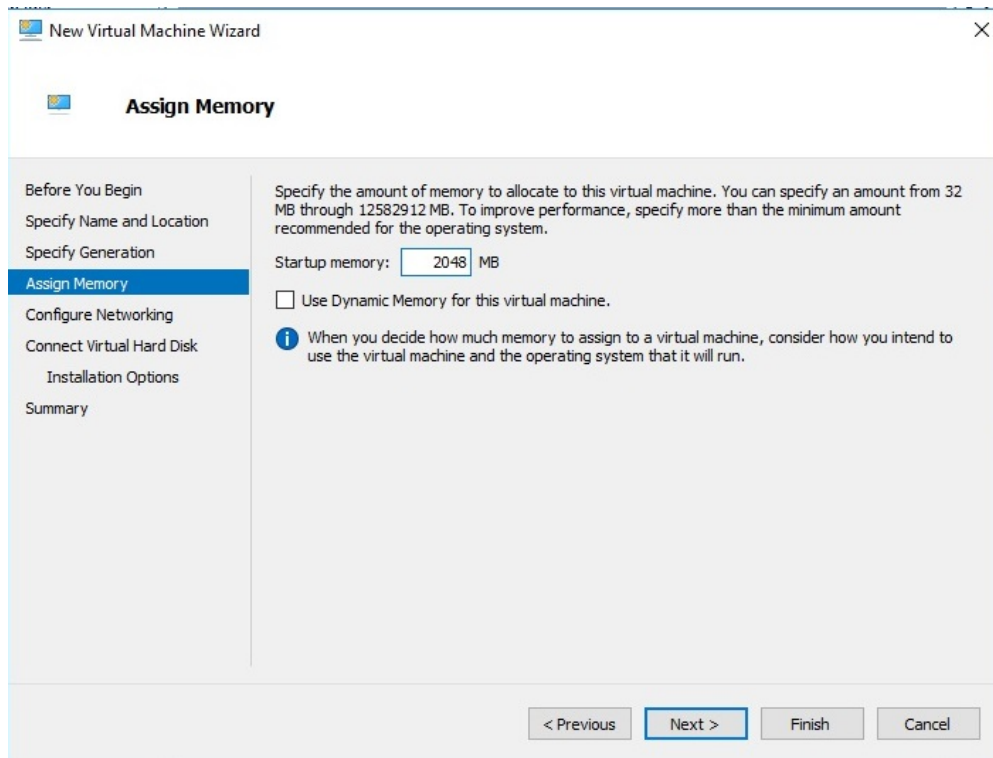
Tämä kohta ohitettiin ja asetettiin myöhemmin Hyper-V managerin kautta. Kuvassa 12 on nähtävissä Hyper-V-ikkuna Server Managerissa, josta pääsee Hyper-V Manageriin.



Kuva 12. Hyper-V Server Managerissa

Hyper-V Managerissa voidaan aloittaa itse virtuaalilaitteiden luominen. Asennus aloitettiin valitsemalla oikeasta yläkulmasta New virtual machine, jonka jälkeen työkaluun syötettiin virtuaalilaitteen tiedot. Tässä vaiheessa ei vielä pystytty muokkaamaan prosessorin arvoja, vaan se muutettiin myöhemmin virtuaalilaitteen ominaisuuksista.

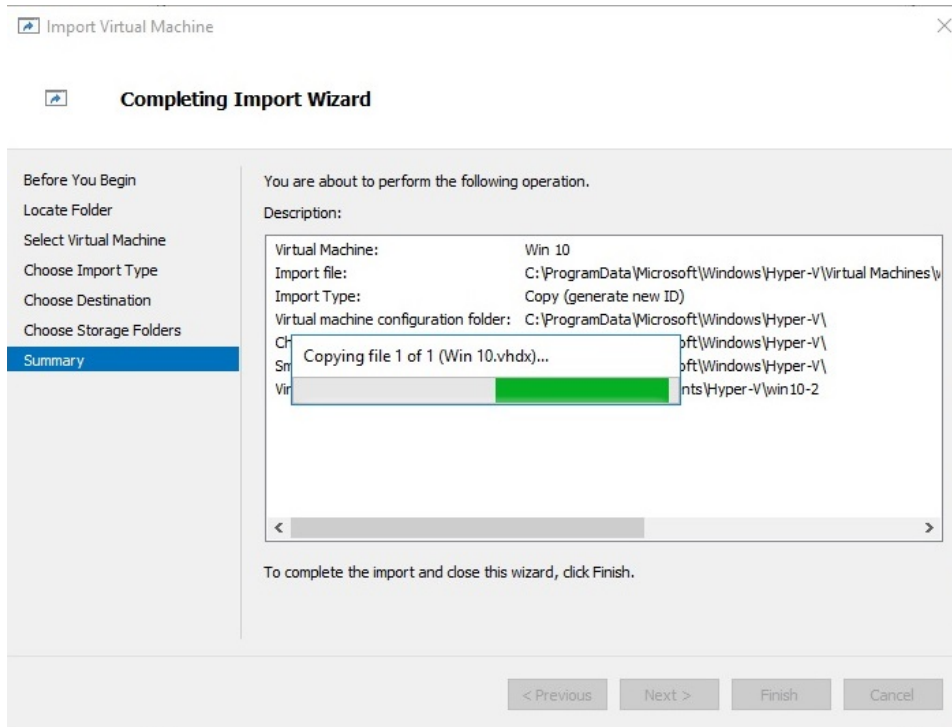
Virtuaalilaitteelle syötettiin nimi, keskusmuistin määrä, virtuaalikytkin, sekä virtuaalisen kiintolevyn koko. Arvoina käytettiin samoja lukuja kuin muilla virtualisointialustoilla. Tämän jälkeen siirryttiin virtuaalilaitteen asetuksiin, josta vaihdettiin prosessorin arvoja samoiksi kuin muilla alustoilla. Kyseisellä tyylillä asennettiin yhdet kappaleet Windows 10- ja Linux-käyttöjärjestelmiä. Kuvassa 13 on nähtävissä uuden virtuaalilaitteen asennustyökalu.



Kuva 13. Uuden virtuaalilaitteen asennustyökalu

Koska Hyper-V:ssä ei ole suoraan virtuaalilaitteen kloonamista, täytyi käyttäjän ensin kopioida virtuaalilaite kiintolevylle, josta voitiin kopioida toimiva virtuaalilaite käytettäväksi. Import eli vientityökalu kysyy ensin kohteen, jossa virtuaalilaite on, jonka jälkeen valitaan vientitapa. Kyseisellä työkalulla voitiin valita virtuaalilaitteen rekisteröinti, virtuaalilaitteen eheyty, sekä virtuaalilaitteen kopiointi. Tätä keinoa käyttäen virtuaalilaitteita kopioitiin ensin viisi kappaletta kumpaakin käyttöjärjestelmää, ja suorituskykytestien jälkeen virtuaalilaitteita kopioitiin toiset viisi lisää. Kuvassa 14 on nähtävissä työkalu, jolla virtuaalilaitteet kopioitiin kiintolevyltä.

Windows Server 2016 Hyper-V:n kanssa voi olla monimutkaisempi käyttää kuin esimerkiksi muut virtualisointialustat, joita tässä työssä käytetään. Palvelinroolien asennus ja käyttö voi olla ensin aikaa vaativaa, sekä Hyper-V:n käyttöliittymä on aika kankea verrattuna muihin. Myös kloonauksen puuttuminen monimutkaistaa kyseisen alustan käyttöä.

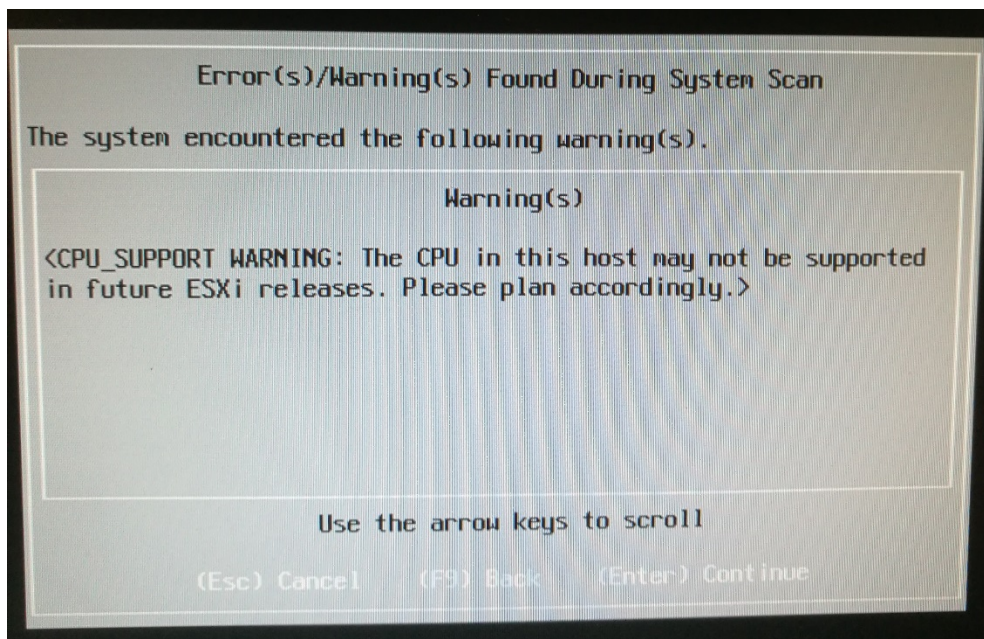


Kuva 14. Virtuaalilaitteen kopiointi kiintolevyltä

### 7.3 ESXi-asennus ja -valmistelu

Proxmoxin tapaan myös ESXi oli yksinkertainen asentaa. Myös ESXi:n konfigurointi on yksinkertaista ja vaatii hyvin vähän osaamista. Asennustyökalu kysyi vain loppukäyttäjän lisensointisopimuksen hyväksymistä, kiintolevyn jolle alusta asennettiin, näppäimistökielen sekä salasanan. Tässä tapauksessa asennus varoitti prosessorin tuesta, sillä myöhemmät julkaisut ESXi:stä eivät välttämättä tukisi kyseistä prosessoria. Koska alustaa ei tarvinnut päivittää opinnäytetyön aikana, työtä jatkettiin normaalisti ilman ongelmia. Kuvassa 15 on nähtävissä ESXi:n asennuksen yhteydessä tullut varoitus prosessorista.



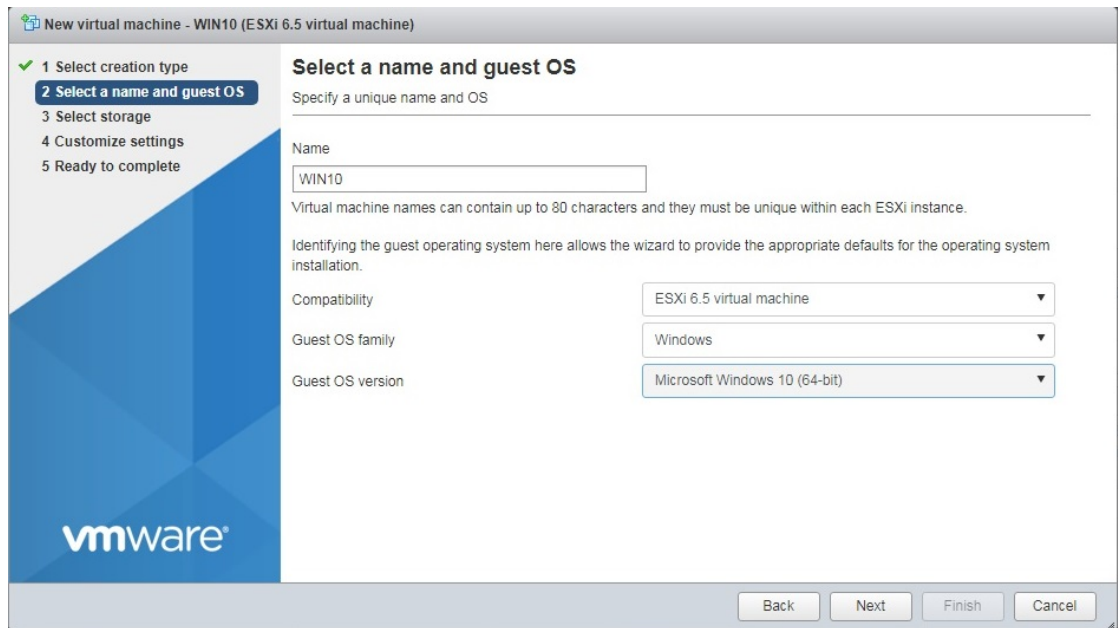


Kuva 15. ESXi:n asennuksen varoitus prosessorista

Asennuksen jälkeen alustalle asetettiin IP-osoite alustan verkkoasetuksista:

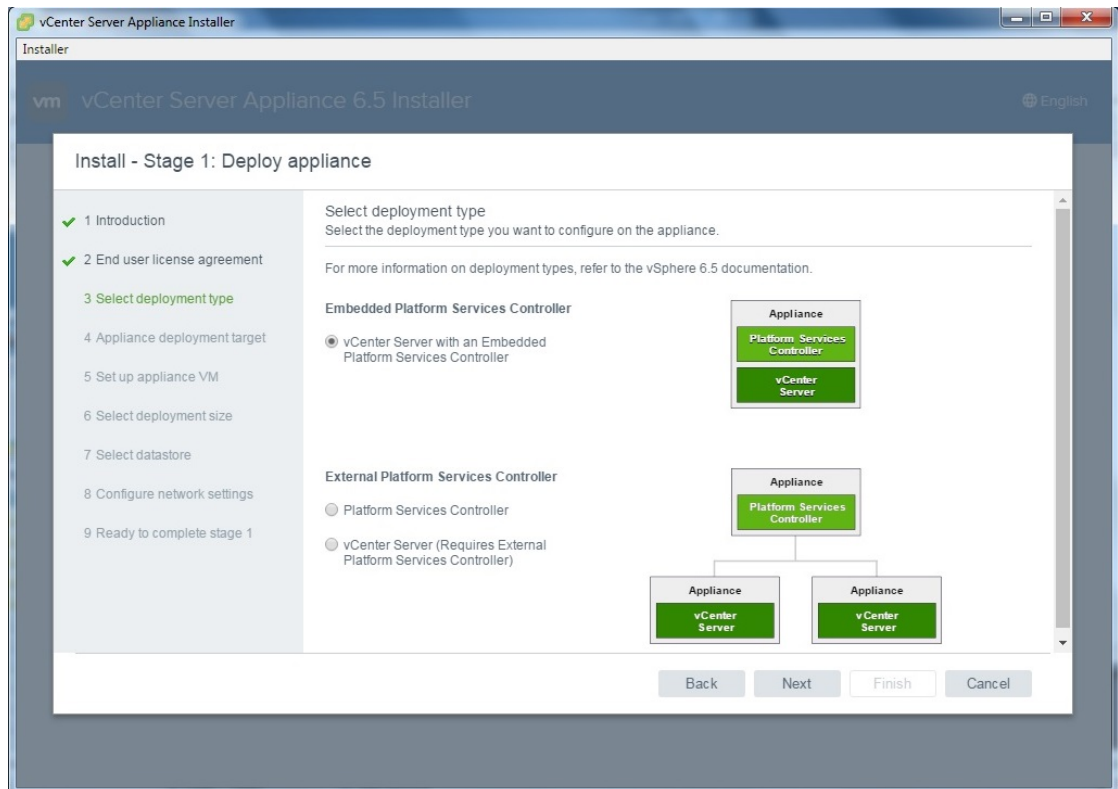
- IP-Osoite: 192.168.100.2
- Aliverkon peite: 255.255.255.0
- Yhdyskäytävä: 192.168.100.1

Verkkoasetusten asettamisen jälkeen voitiin palvelimelle ottaa etäyhteys samassa verkossa olevan tietokoneen kautta. Samassa verkossa olevan tietokoneen verkkoselaimeen kirjoitettiin palvelimen IP-osoite, jonka jälkeen alustalle kirjauduttiin sisään aikaisemmin asettamilla tunnuksilla. Tämän jälkeen Web Clientillä asennettiin Windows 10- sekä Linux-käyttöjärjestelmät. Virtuaalilaitteiden asetuksiksi asetettiin samat arvot kuin muissa virtuaalialustoissa. Kuvas-  
sassa 16 on nähtävissä ESXi Web Clientin virtuaalilaitteen asennustyökalu.



Kuva 16. ESXi Web Clientin virtuaalilaitteen asennustyökalu

Koska VMware ESXi:n Web Client ei tarjoa kaikkia samoja ominaisuuksia kuin vCenter, virtualisointialustan sisälle asennettiin vCenter Server Appliance -niminen virtuaalilaite. vCSA tarjoaa samat toiminnot kuin perinteinen vCenter Server, mutta se on pakattuna Windows-palvelimen sijasta Linux-pohjaiselle palvelimelle. VMware on valmistanut vCSA:ta varten myös käyttäjille yksinkertaisen graafisen asennusohjelman. Asennusohjelmaan syötettiin ESXi-palvelimen IP-osoite ja tunnukset, sekä vCSA:n haluamat verkkoasetukset sekä tunnukset. Kuvassa 17 on nähtävissä vCenter Server Applianceen graafinen asennustyökalu.



Kuva 17. vCenter Server Appliancen graafinen asennustyökalu

Asennuksen jälkeen vCSA on valmiiksi käynnissä, ja se on konfiguroitu asennusohjelmaan syötetyillä asetuksilla. Seuraavaksi virtuaalilaitteeseen otettiin yhteys verkkoselaimella. Jotta virtuaalilaitteiden lisäys ja kloonaminen voitiin aloittaa, vCSA:han lisättiin vSphere Clientillä datakeskus ja sille isäntä. Isännän IP-osoitteeksi syötettiin ESXi:n IP-osoite sekä tunnukset, jolla siihen kirjauduttiin.

Virtuaalilaitetta oikealla hiiren painikkeella klikkaamalla voitiin haluama virtuaalilaitte kloonata joko toiseksi virtuaalilaitteeksi tai valmiiksi malliksi tulevia virtuaalilaitteita varten. Virtuaalilaitteita kloonattiin yhteensä 10 kappaletta, viisi kumpaakin käyttöjärjestelmää. Testien jälkeen virtuaalilaitteiden määrä nostettiin kahteenkymmeneen kloonamalla.

## 8 SUORITUSKYKYTESTIT JA TULOSTEN VERTAILU

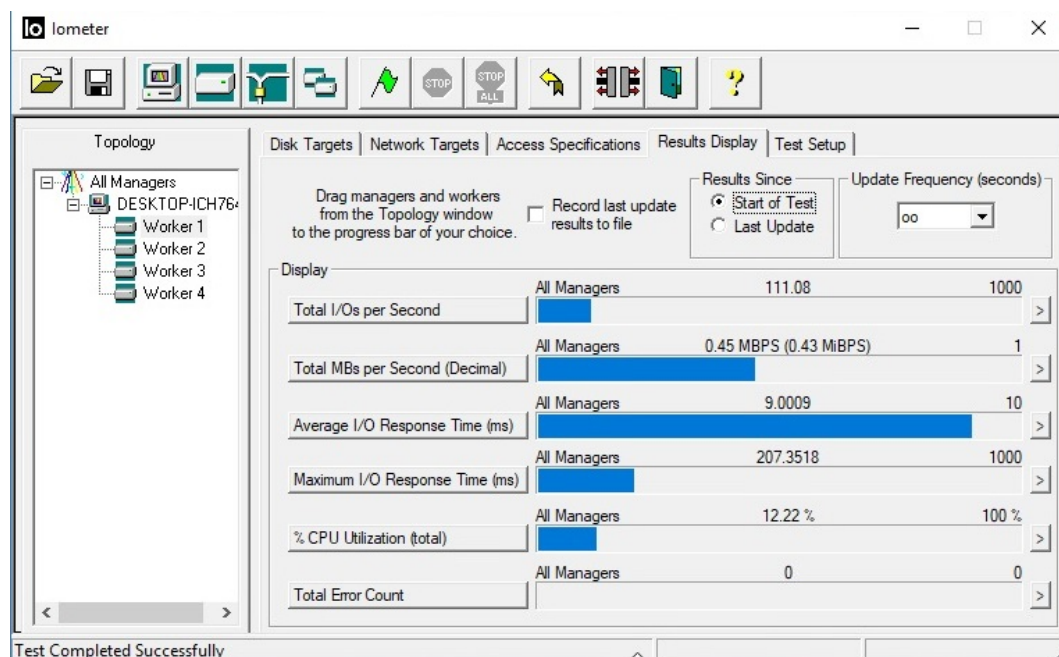
Suorituskykyvertailussa valittiin ohjelmia, jotka olivat ilmaisia tai joista oli saatavilla ilmainen kokeiluversio. Näin virtualisointialustojen vertailua voidaan jatkaa samoilla toimenpiteillä jatkossa ilmaiseksi, esimerkiksi tehokkaammalla tai vähemmän tehokkaalla palvelimella. Tässä suorituskykyvertailussa verrattiin



IOPS-arvoja, keskusmuistin luku- ja kirjoitusnopeutta, verkon viiveitä sekä prosessorin suoriutumista erilaisista suorituskykytesteistä. Prosessorin suorituskykytestit valittiin sillä perusteella, että jokainen testi keskittyisi prosessorin eri vahvuuksiin, kuten 3D-kuvan piirtämiseen tai yhden ytimen suorituskykyyn.

## 8.1 IOPS-mittaukset

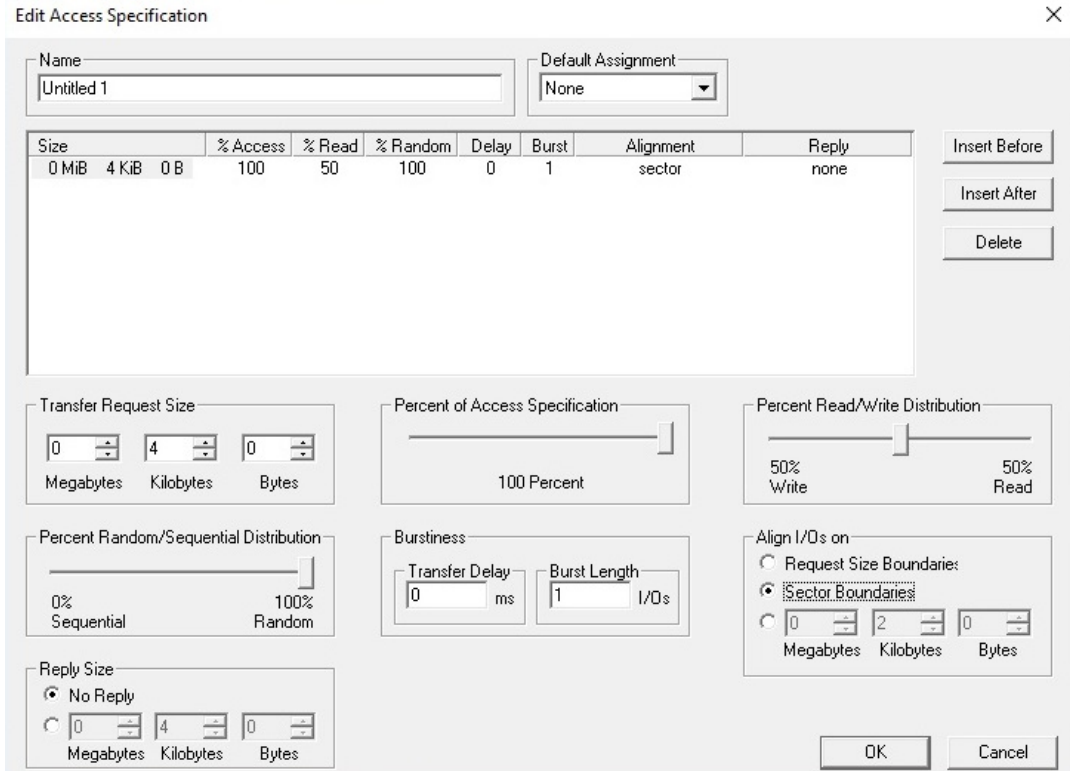
IOPS eli Input Output Per Second on datan luku- ja kirjoitusmenetelmä, jossa lasketaan kuinka monta syöte- ja tulostetehtävää kiintolevy suorittaa sekunnissa. Tässä opinnäytetyössä käytettiin IOmeter-ohjelmaa, joka mittaa IOPS-arvojen lisäksi esimerkiksi kirjoitus- ja lukunopeutta megatavua sekuntia kohti, keskimääräisen ja korkeimman input- ja output-vasteajan, sekä prosessorin käyttöprosentin. Tässä työssä tarkastellaan vain IOPS-tuloksia. Kuvassa 18 on nähtävissä suoritettu IOmeter-testi.



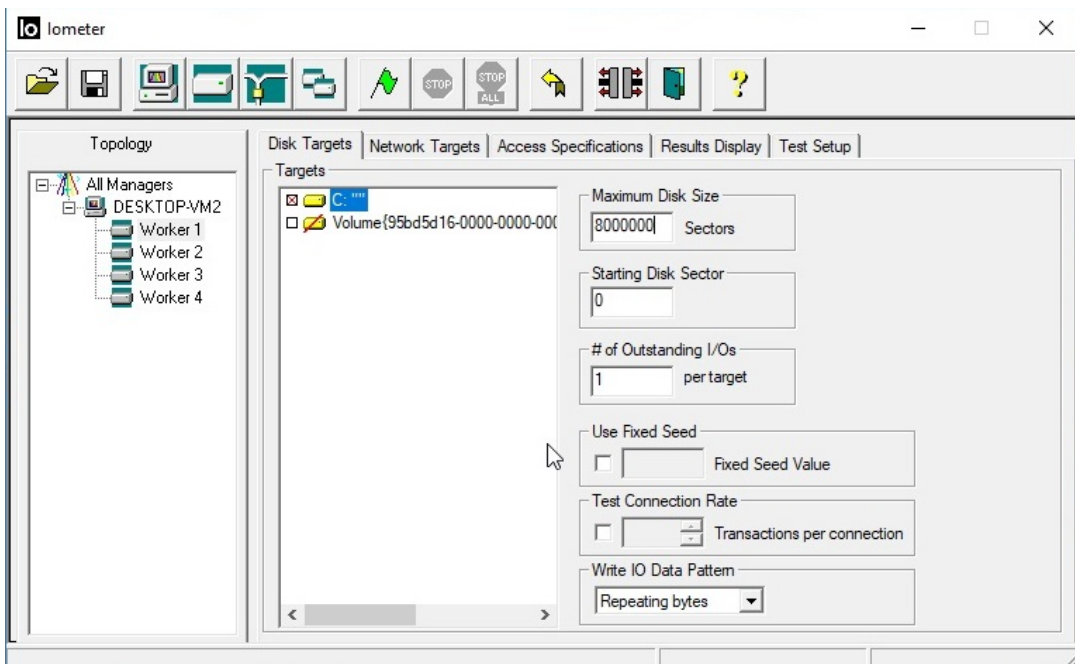
Kuva 18. Suoritettu IOmeter-testi

IOmeterillä luotiin virtuaalilaitteelle 4 gigatavua suuri väliaikainen tiedosto, jota ohjelma käyttää testin aikana. IOmeter pyytää tiedoston koon sektoreina, ja 4 gigatavua on karkeasti noin 8 miljoonaa sektoria. Ohjelma tarjosi valmiita profiileja joilla testi voitiin suorittaa, mutta tässä opinnäytetyössä luotiin oma profiili. Siirtopyynnön kooksi valittiin neljä kilotavua, luku- ja kirjoitussuhteeksi 50 %, sekä satunnaissuhteeksi 100 %. Kaikki testit asetettiin viiden minuutin pituisiksi, ja testejä suoritettiin kolme kappaletta kutakin, jotta saatiin tarkempi

tulos keskiarvon perusteella. Kuvissa 19 ja 20 esitetään testiprofiilin luonti sekä IOMeterin aloitusikkuna, johon tiedoston koko syötettiin. Aloitusikkunasta valittiin myös osio, jolle väliaikainen tiedosto tehtiin.



Kuva 19. Testiprofiilin luonti-ikkuna



Kuva 20. IOMeterin aloitusruutu

### 8.1.1 Ensimmäinen IOPS-testi

Ensimmäiset testit suoritettiin kaikkien virtuaalilaitteiden ollessa mahdollisimman vähäisen rasituksen alaisena. Kymmenen virtuaalilaitteen kanssa tulokset olivat seuraavat parhaimmasta huonoimpaan:

- **Proxmox:** 244 IOPS
- **Hyper-V:** 233 IOPS
- **ESXi:** 160 IOPS

Vaikka Proxmox ja Hyper-V suoriutuivat testistä lähes samoilla tuloksilla, ESXi kuitenkin jäi muille alustoille noin 35 prosentilla. Seuraavaksi suoritettiin sama testi niin, että virtualisointialustoilla oli käynnissä yhteensä 20 virtuaalilaitetta. Tuloksista huomattiin, että alustat käyttäytyivät samalla tavalla kuin kymmenellä virtuaalilaitteella, mutta tulokset olivat kaikilla hieman matalimmat suuremman kiintolevyjen käytön vuoksi:

- **Proxmox:** 214 IOPS
- **Hyper-V:** 200 IOPS
- **ESXi:** 135 IOPS

### 8.1.2 Toinen IOPS-testi

Seuraavaksi suoritettiin sama testi siten, että toisella virtuaalilaitteella on samaan aikaan kyseinen IOPS-testi päällä. Näin voitiin imitoida tasa-arvoisesti useamman virtuaalilaitteen kiintolevyn käyttöä, samalla keräten tulokset helposti ylös. Tulokset kymmenellä virtuaalilaitteella olivat seuraavat:

- **Hyper-V:** 205 IOPS
- **Proxmox:** 204 IOPS
- **ESXi:** 106 IOPS

Tässä testissä havaittiin yhtenäisyyksiä ensimmäiseen testiin. Hyper-V ja Proxmox suoriutuvat testistä yhtä hyvin, kun taas ESXi:llä testattu IOPS-arvo

jäi muista alustoista melkein puolella. Seuraavaksi tämä testi suoritettiin uudelleen, kun virtuaalilaitteita oli kaksikymmentä:

- **Hyper-V:** 198 IOPS
- **Proxmox:** 165 IOPS
- **ESXi:** 94 IOPS

Kahdellakymmenellä virtuaalilaitteella havaittiin, että ESXi:n sekä Hyper-V:n suorituskyky ei laskenut juuri ollenkaan, vaikka virtuaalilaitteita oli tuplasti enemmän. Hyper-V:n kohdalla todettiin myös, ettei tuloksiin juurikaan vaikuttanut testin samanaikainen suorittaminen kahdella virtuaalilaitteella.

### 8.1.3 Kolmas IOPS-testi

Kolmanneksi IOPS-testiksi valittiin sama viiden minuutin testi, mutta taustalla suoritettiin samalla virtuaalilaitteella Prime95-ohjelmaa. Prime95-ohjelmalla imitoitiin virtuaalilaitteen raskasta käyttöä. Tulokset olivat kymmenellä virtuaalilaitteella seuraavat:

- **Proxmox:** 176 IOPS
- **ESXi:** 163 IOPS
- **Hyper-V:** 109 IOPS

Verrattuna ensimmäiseen testiin, Hyper-V näytti kärsivän rasituksesta testin aikana eniten, kun taas ESXi selvisi testistä lähes samoilla tuloksilla ilman rasitusta. Kahdenkymmenen virtuaalilaitteen tuloksissa Proxmox suoriutui parhaiten:

- **Proxmox:** 230 IOPS
- **ESXi:** 134 IOPS
- **Hyper-V:** 111 IOPS

Vaikka Hyper-V:n tulos oli lähes sama kuin kymmenellä virtuaalilaitteella, vaikutti se kärsivän räsityksestä eniten tässäkin testissä. ESXi pysyi taas samoissa arvoissa kuin ilman virtualisointialustan räsitystä. Proxmox toisaalta suoriutui useamman virtuaalilaitteen testistä jopa paremmin kuin kymmenellä virtuaalilaitteella.

Vaikka tulosten erot olivat osittain hyvin pienet, Proxmox kuitenkin suoriutui testeistä parhaiten. ESXi:n arvojen erot olivat joissain testeissä jopa puolet parhaiten suoriutuvasta, joka on aika suuri ero kirjoitus- ja lukunopeuden kannalta.

## 8.2 Keskusmuistin nopeuden mittaukset

Keskusmuistin luku- ja kirjoitusnopeus mitattiin PassMark PerformanceTestin Advanced Memory -testillä. Testit suoritettiin Memory Speed Per Block Size -testillä, ja Access Data Type asetettiin yhden tavun kokoiseksi. Kaikki testit suoritettiin kolme kertaa, jotta tuloksista saatiin mahdollisimman tarkat keskiarvon perusteella. Kuvassa 21 on esitettyjä asetukset, jolla testit suoritettiin.

Memory Speed Graph

Test Type

Block Read/Write: Write

Memory Speed Per Access Step Size:

Memory Speed Per Block Size:

Access Data Type: Byte (8-bits)

NUMA Settings

Processor #: 0

NUMA Node: 0

NUMA Allocation Node: N/A (SMP)

Total Physical Memory: 4294430720 Bytes

Block Size: 2147483648 Bytes

Progress

Test Complete

Kuva 21. Advanced memory testiin syötetyt asetukset

### 8.2.1 Ensimmäinen lukutesti

Ensimmäinen lukutesti suoritettiin kaikkien virtuaalilaitteiden rasituksen ollessa mahdollisimman matalalla. Kymmenellä virtuaalilaitteella tulokset olivat:

- **Proxmox:** 1101 MB/s
- **ESXi:** 1047 MB/s
- **Hyper-V:** 807 MB/s

Proxmox ja ESXi suoriutui testistä lähes samoilla tuloksilla, kun taas Hyper-V:n tulokset olivat melkein 30 % huonommat kuin Proxmoxilla. Kun virtuaalilaitteita oli 20 kappaletta, tulokset olivat seuraavat:

- **ESXi:** 1128 MB/s
- **Proxmox:** 1082 MB/s
- **Hyper-V:** 785 MB/s

Proxmoxin ja Hyper-V:n tuloksiin virtuaalilaitteiden määrä vaikutti hyvin vähän. ESXi:n tulokset olivat kuitenkin jopa hieman paremmat kuin pienemmällä virtuaalilaitemäärällä.

### 8.2.2 Toinen lukutesti

Toinen lukutesti suoritettiin niin, että samalla virtuaalilaitteella suoritettiin taustalla Prime95-niminen ohjelma. Kymmenen virtuaalilaitteen tulokset olivat:

- **ESXi:** 1105 MB/s
- **Proxmox:** 1003 MB/s
- **Hyper-V:** 914 MB/s

Proxmox suoriutui rasituksen alaisena huonommin kuin ilman, Hyper-V ja ESXi sen sijaan suoriutuivat rasituksen alla jopa paremmin kuin ilman rasi-  
tusta. Kahdenkymmenen virtuaalilaitteen tulokset olivat seuraavat:

- **ESXi:** 1074 MB/s
- **Proxmox:** 965 MB/s
- **Hyper-V:** 945 MB/s

Tästä testistä Proxmox ja ESXi suoriutui huonommin rasituksen kanssa kuin ilman. Hyper-V kuitenkin suoriutui paremmin, kuin missään muussa testissä.

Havaittiin, että Hyper-V suoriutuu parhaiten suuremman käytön ja kovemman rasituksen alaisena, ainakin tämän opinnäytetyön käsittämässä skaalassa. Muuten alustat suoriutuivat joko miltei yhtä hyvin, tai hieman huonommin, mitä enemmän alustalla oli rasi-  
tusta.

### 8.2.3 Ensimmäinen kirjoitustesti

Kirjoitustestit ovat samankaltaisia kuin lukutestit, mutta testissä mitataan kirjoit-  
tusuopeutta. Ensimmäisessä kirjoitustestissä siis kirjoitustestien keskiarvo, en-  
sin kymmenellä virtuaalilaitteella:

- **Proxmox:** 1182 MB/s
- **ESXi:** 1135 MB/s
- **Hyper-V:** 854 MB/s

Virtualisointialustojen tulokset kirjoitustestissä muistuttivat paljon lukutestejä. Hyper-V suoriutui taas huonoiten, Proxmox ja ESXi pienellä eroavaisuudella. Seuraavaksi suoritettiin sama testi kahdellakymmenellä virtuaalilaitteella:

- **ESXi:** 1143 MB/s
- **Proxmox:** 1096 MB/s
- **Hyper-V:** 984 MB/s

Myös kirjoitustestissä ESXi suoriutui useammalla virtuaalilaitteella paremmin, tällä kertaa tosin hyvin pienellä määrällä. Hyper-V:n kirjoitusnopeus oli nopeampi useammalla virtuaalilaitteella, kun taas Proxmoxilla huonompi.

#### 8.2.4 Toinen kirjoitustesti

Kuten lukutestissä, tämäkin testi suoritettiin Prime95-ohjelman pyöriessä taustalla. Kymmenellä virtuaalilaitteella tulokset olivat:

- **Proxmox:** 1085 MB/s
- **ESXi:** 1082 MB/s
- **Hyper-V:** 958 MB/s

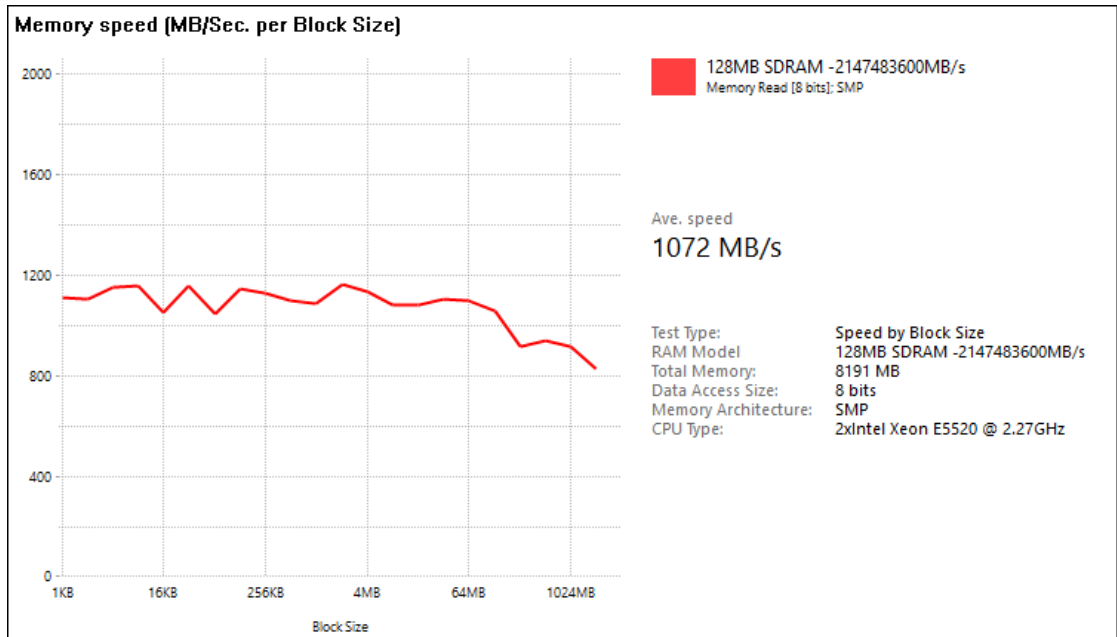
Ilman rasitusta Proxmox ja ESXi:n tulokset olivat paremmat, kun taas Hyper-V:n tilanne oli päinvastainen. Kahdellakymmenellä virtuaalilaitteella tulokset olivat seuraavanlaiset:

- **ESXi:** 1087 MB/s
- **Proxmox:** 993 MB/s
- **Hyper-V:** 922 MB/s

Sekä luku- että kirjoitustesteissä voitiin nähdä paljon samankaltaisuuksia. Vaikka Hyper-V suoriutui kaikista testeistä huonoiten, sen tulokset olivat paremmat rasituksen kanssa kuin ilman. Vaikka Hyper-V:n tulokset olivat muita virtualisointialustoja huonommat rasituksen kanssa, olivat ne suhteessa paremmat verraten testeihin ilman rasitusta.

ESXi vaikutti kärsivän virtuaalilaitteiden määrästä hyvin vähän, sillä tulokset olivat samaa luokkaa useammalla virtuaalilaitteella. Proxmoxilla taas tulokset huonivat keskimäärin saman verran joka testissä. Muistinnopeuden perusteella rasituksen kanssa ja ilman ESXi suoriutui testeistä parhaiten. Kuvassa 22 on esitetty esimerkkitulo lukutestistä.





Kuva 22. Muistin lukunopeustesti ESXi:n sisällä olevalla virtuaalilaitteella

### 8.3 Verkon mittaukset

Verkon toimivuutta ja luotettavuutta mitattiin virtuaalisen koneen ja fyysisen koneen välillä iPerf3-ohjelmalla. iPerf3 avattiin kahdella laitteella, tässä tapauksessa fyysisellä koneella sekä virtuaalilaitteella. Fyysisen koneen puolella iPerf3-ohjelmalle syötettiin komento `iperf3.exe -s 192.168.100.5`, joka kuuntelee TCP- ja UDP-yhteyksiä toimien tässä testissä palvelimena.

#### 8.3.1 Ensimmäinen verkonmittaus

Ensimmäisessä testissä iPerf3-ohjelmaan syötettiin käyttäjäpuolella:

```
iperf3.exe -c 192-168.100.5 -f m -t 70 -P 2
```

Kyseisissä komennoissa `-s` tarkoittaa palvelinta, ja `-c` käyttäjää. Käyttäjäpuolelle syötettiin myös `-f`, joka tarkoittaa formaattia. Tämän arvoksi syötettiin `m`, joka tulostaa testin tulokset megabitteinä sekunnissa. Aikaintervalliksi syötettiin `-t 70`, joka tarkoittaa 70 sekuntia, ja `-P` kuinka monella samanaikaisella yhteydellä testi suoritettiin.

Kun molemmat komennot käynnistettiin, saatiin mitattua verkon suoritusteho. Koska kyseisen testin pullonkaula oli kytkimessä, suoritettiin samanaikaisesti komentorivillä ping-komento, joka mittasi viivettä haluttuun kohteeseen:

```
ping 192.168.100.5 -l 50000 -n 60
```

Tämä ping-testi syötettiin virtuaalilaitteen komentoriville, jossa iPerf3-palvelin oli käynnissä. Komennossa *-l* tarkoittaa puskurin kokoa, joka on asetettu 50 kilotavuun. Arvo *-n* määrittää testin pituuden, joka on asetettu 60 sekunniksi.

Tulokset ovat keskiarvo 60 sekunnin testin tuloksista:

- **Hyper-V:** 37 ms
- **ESXi:** 39 ms
- **Proxmox:** 39 ms

Sama testi kahdellakymmenellä virtuaalilaitteella:

- **Hyper-V:** 38 ms
- **ESXi:** 38 ms
- **Proxmox:** 39 ms

Koska tulokset olivat kaikilla vain muutaman millisekunnin heittoa lukuun ottamatta samat, todettiin että tämän testin kannalta kaikki suoriutuivat yhtä hyvin. Myöskään virtuaalilaitteiden määrä ei vaikuttanut verkon toimintaan tällä skaalalla. Maksimiviiveetkin olivat samaa luokkaa kaikissa.

### 8.3.2 Toinen verkonmittaus

Toisessa testissä käytettiin TCP-protokollan sijasta UDP-protokollaa, ja testillä mitattiin virtualisointialustojen eri jitter-arvoja. Jitter tarkoittaa kahden järjestelmän välillä olevan pakettivirran latenssin vaihtelua. Tässäkin testissä käytettiin iPerf3-ohjelmaa, mutta eri komennolla:

```
iperf3.exe -c 192.168.100.5 -u -b 90m -t 60
```

Tässä komennossa *-u* merkitsee UDP-protokollan käyttöä, *-b* kaistanleveyttä joka asetettiin 90 megabittiin sekunnissa, ja aikaintervalliksi *-t* 60 sekuntia.

Kymmenellä virtuaalilaitteella tulokset olivat:

- Hyper-V: 0.41 ms
- ESXi: 0.78 ms
- Proxmox: 6.97 ms

Tässä testissä huomattiin jo pieniä eroja, sillä Proxmoxin tulos on yli 6 millisekuntia muista virtualisointialustoista. Kahdellakymmenellä virtuaalilaitteella tulokset olivat:

- Hyper-V: 0.49 ms
- ESXi: 5.20 ms
- Proxmox: 5.23 ms

Hyper-V:n ja Proxmoxin kohdalla eroja ei juurikaan havaittu, mutta ESXi:n tulos nousi melkein viidellä millisekunnilla. Tulokset olivat lopussa hyvin pieniä, joten tälläkään testillä ei saatu merkittäviä tuloksia opinnäytetyötä varten. Jos verkon skaala olisi suurempi, eroja voitaisiin huomata. Kuvassa 27 on nähtävillä suoritettu UDP-testi.

```

Komentokehote
[ 4] 38.00-39.00 sec 11.0 MBytes 92.0 Mbits/sec 1403
[ 4] 39.00-40.00 sec 10.6 MBytes 88.8 Mbits/sec 1356
[ 4] 40.00-41.00 sec 10.5 MBytes 88.0 Mbits/sec 1343
[ 4] 41.00-42.00 sec 11.1 MBytes 93.0 Mbits/sec 1419
[ 4] 42.00-43.00 sec 10.9 MBytes 91.3 Mbits/sec 1393
[ 4] 43.00-44.00 sec 10.6 MBytes 88.8 Mbits/sec 1355
[ 4] 44.00-45.00 sec 10.6 MBytes 89.0 Mbits/sec 1358
[ 4] 45.00-46.00 sec 10.6 MBytes 88.8 Mbits/sec 1355
[ 4] 46.00-47.00 sec 10.9 MBytes 91.7 Mbits/sec 1400
[ 4] 47.00-48.00 sec 10.6 MBytes 88.9 Mbits/sec 1356
[ 4] 48.00-49.00 sec 10.6 MBytes 88.9 Mbits/sec 1356
[ 4] 49.00-50.00 sec 11.3 MBytes 94.8 Mbits/sec 1447
[ 4] 50.00-51.00 sec 10.1 MBytes 84.6 Mbits/sec 1291
[ 4] 51.00-52.00 sec 11.3 MBytes 94.7 Mbits/sec 1448
[ 4] 52.00-53.00 sec 10.5 MBytes 88.4 Mbits/sec 1346
[ 4] 53.00-54.00 sec 10.9 MBytes 91.8 Mbits/sec 1401
[ 4] 54.00-55.00 sec 10.5 MBytes 88.3 Mbits/sec 1347
[ 4] 55.00-56.00 sec 10.6 MBytes 88.7 Mbits/sec 1354
[ 4] 56.00-57.00 sec 10.8 MBytes 90.3 Mbits/sec 1378
[ 4] 57.00-58.00 sec 10.6 MBytes 88.9 Mbits/sec 1356
[ 4] 58.00-59.00 sec 10.8 MBytes 90.6 Mbits/sec 1382
[ 4] 59.00-60.00 sec 10.6 MBytes 88.7 Mbits/sec 1354
-----
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[ 4]  0.00-60.00 sec  643 MBytes  89.9 Mbits/sec  0.409 ms   15/82316 (0.018%)
[ 4] Sent 82316 datagrams

iperf Done.
C:\share\iperf>

```

Kuva 23. Suoritettu UDP-testi

Tässä työssä käytetty lähiverkko oli pieni, joten testien tulokset olivat hyvin ta-  
saisia ja niistä ei ilmennyt mitään ongelmia. Tulosten perusteella voitiin aina-  
kin todeta, ettei millään virtualisointialustalla ole ongelmia pienen verkon  
kanssa, eikä virtuaalilaitteiden määrä vaikuta sen suoritukseen.

#### 8.4 Prosessorin suorituskykytestit

Prosessorin suorituskykyä virtualisointialustojen välillä vertailtiin kolmella eri-  
laisella suorituskykytestillä. Ensimmäinen suorituskykytesti jota suoritettiin, oli  
NovaBench. Tämä suorituskykytesti antaa testin tuloksen kolmen erilaisen  
laskentaoperaation perusteella. Toinen suorituskykytesti oli CineBench, joka  
on 3D-renderöintiin perustuva suorituskykytesti. Kolmas suorituskykytesti oli  
SuperPi, joka laskee piin desimaaleja yhdellä prosessoriytimellä.

### 8.4.1 Prosessorin suorituskykytestit ensimmäisellä ohjelmalla

Ensimmäisenä ohjelmana käytettiin NovaBench-suorituskykytestiä. Suorituskykytesti suoritettiin ensin kymmenellä virtuaalilaitteella, kun virtualisointialustan rasitus oli mahdollisimman alhainen. Tuloksissa suurempi tulos oli parempi:

- **ESXi:** 363
- **Proxmox:** 350
- **Hyper-V:** 296

ESXi sekä Proxmox suoriutuivat tästä testistä melko samoilla tuloksilla, mutta Hyper-V jäi tuloksista taka-alalle. Kahdellakymmenellä virtuaalilaitteella tulokset olivat odotettavasti kaikilla hieman huonommat:

- **ESXi:** 356
- **Proxmox:** 344
- **Hyper-V:** 294

Seuraavaksi sama suorituskykytesti suoritettiin, kun virtualisointialustan prosessorin rasitus oli nostettu maksimiin. Tällä imitoitiin palvelimen todella raskasta käyttöä. Prosessorin rasitus nostettiin usealla virtuaalilaitteella, joissa suoritettiin Prime95-ohjelmaa taustalla. Tulokset olivat seuraavat, kun virtuaalilaitteita oli alustalla kymmenen:

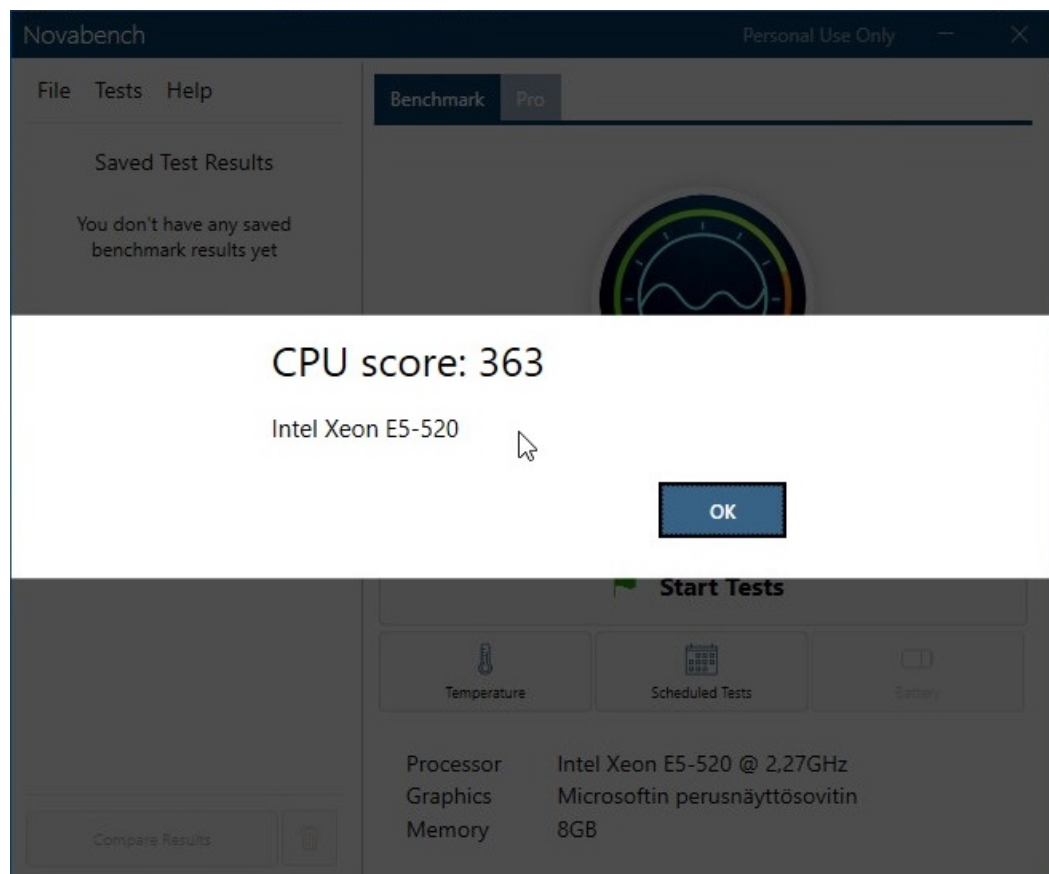
- **ESXi:** 279
- **Hyper-V:** 266
- **Proxmox:** 264

Vaikka tulokset olivat tasaiset, ESXi:n sekä Proxmoxin tulokset laskivat kolme kertaa enemmän kuin Hyper-V:llä, jos tuloksia verrataan ensimmäisiin arvoihin. Tästä voitiin päätellä, että Hyper-V suoriutuu muista alustoista paremmin rasituksen alla. Seuraavaksi sama testi suoritettiin kahdellakymmenellä virtuaalilaitteella:

- **Proxmox:** 279
- **ESXi:** 278
- **Hyper-V:** 220

Useammalla virtuaalilaitteella ei näyttänyt olevan vaikutusta ainakaan tässä skaalassa muilla kuin Hyper-V:llä. Proxmox suoriutui tästä paremmin kuin kymmenellä virtuaalilaitteella, ja ESXi pysyi samoissa arvoissa.

ESXi suoriutui ensimmäisen ohjelman testeistä parhaimmilla tuloksilla, mutta Hyper-V kärsi eniten raskaasta rasituksesta testin aikana. Kuva 28 näyttää suoritettua suorituskykytestin NovaBenchillä.



Kuva 24. NovaBench-suorituskykytesti

#### 8.4.2 Prosessorin suorituskykytestit toisella ohjelmalla

Toisena ohjelmana käytettiin CineBench-ohjelmaa. Suorituskykytesti suoritettiin ensin kymmenellä virtuaalilaitteella, kun virtualisointialustalla oli mahdollisimman matala rasitus. Tuloksissa suurempi tulos oli parempi:

- **ESXi:**            280
- **Hyper-V:**        262
- **Proxmox:**        251

Tästä testistä alustat suoriutuivat tasaisesti, ESXi jälleen tulosten kärjessä. Kuten ensimmäisenkin ohjelman testeissä, kahdenkymmenen virtuaalilaitteen tuloksissa nähtiin pientä laskua, mutta järjestys on sama:

- **ESXi:**            273
- **Hyper-V:**        241
- **Proxmox:**        235

Seuraavassa testissä käytettiin taas Prime95-ohjelmaa taustalla, imitoiden palvelimen todella raskasta käyttöä. Kymmenellä virtuaalilaitteella tulokset olivat:

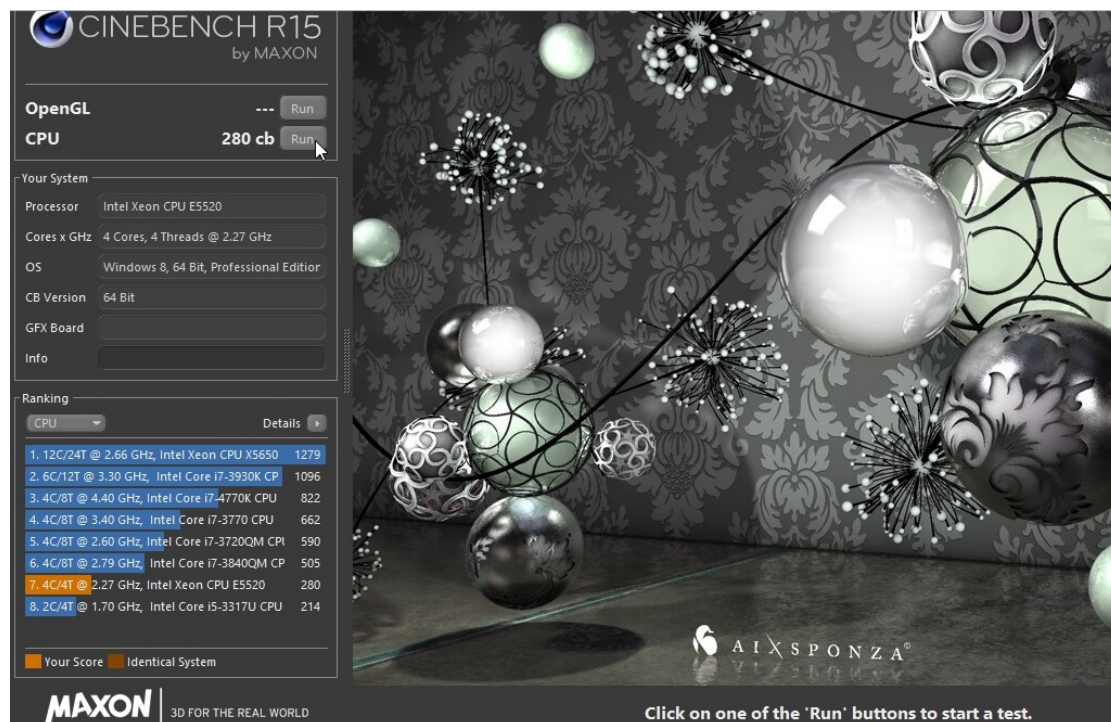
- **ESXi:**            200
- **Hyper-V:**        156
- **Proxmox:**        154

ESXi:n tulos oli edelleen kärjessä, ja ensimmäisiin testeihin verraten Hyper-V:n ja Proxmoxin tulokset laskivat yhtä paljon. Kahdellakymmenellä virtuaalilaitteella tulokset laskivat taas kaikilla:

- **ESXi:**            163
- **Proxmox:**        145
- **Hyper-V:**        96

Proxmox suoriutui useammalla virtuaalilaitteella lähes yhtä hyvin, kun taas Hyper-V:n sekä ESXi:n tulos laski melkein yhtä paljon.

ESXi sai parhaimmat tulokset ja näytti kärsivän keskimäärin rasituksesta yhtä paljon kuin Proxmox. Hyper-V suoriutui tästäkin testistä huonoiten. Kuva 29 näyttää suoritettua CineBench-suorituskykytestin.



Kuva 25. Cinebench-suorituskykytesti

### 8.4.3 Prosessorin suorituskykytestit kolmannella ohjelmalla

Viimeinen ohjelma, jota tässä opinnäytetyössä käytettiin, oli SuperPi. Vaikka kyseessä onkin vanha suorituskykytesti, se antaa helposti ymmärtävän tuloksen testin jälkeen. SuperPi laskee piin desimaaleja esimerkiksi 4 megatavun verran tekstitiedostoon, jonka jälkeen tulostaa ruudulle ajan, kuinka kauan prosessissa meni. Tässä opinnäytetyössä käytettiin neljän megatavun testiä. Ensimmäiseksi testi toteutettiin, kun alustoilla oli kymmenen virtuaalilaitetta päällä. Tulokset olivat:



- **Proxmox:** 88 sekuntia
- **ESXi:** 92 sekuntia
- **Hyper-V:** 102 sekuntia

Proxmox ja ESXi suoriutui tästä testistä lähes yhtä hyvin, ja Hyper-V osoitti pientä eroa tuloksessa. Kahdellakymmenellä virtuaalilaitteella erot suurenivat:

- **Proxmox:** 93 sekuntia
- **ESXi:** 99 sekuntia
- **Hyper-V:** 112 sekuntia

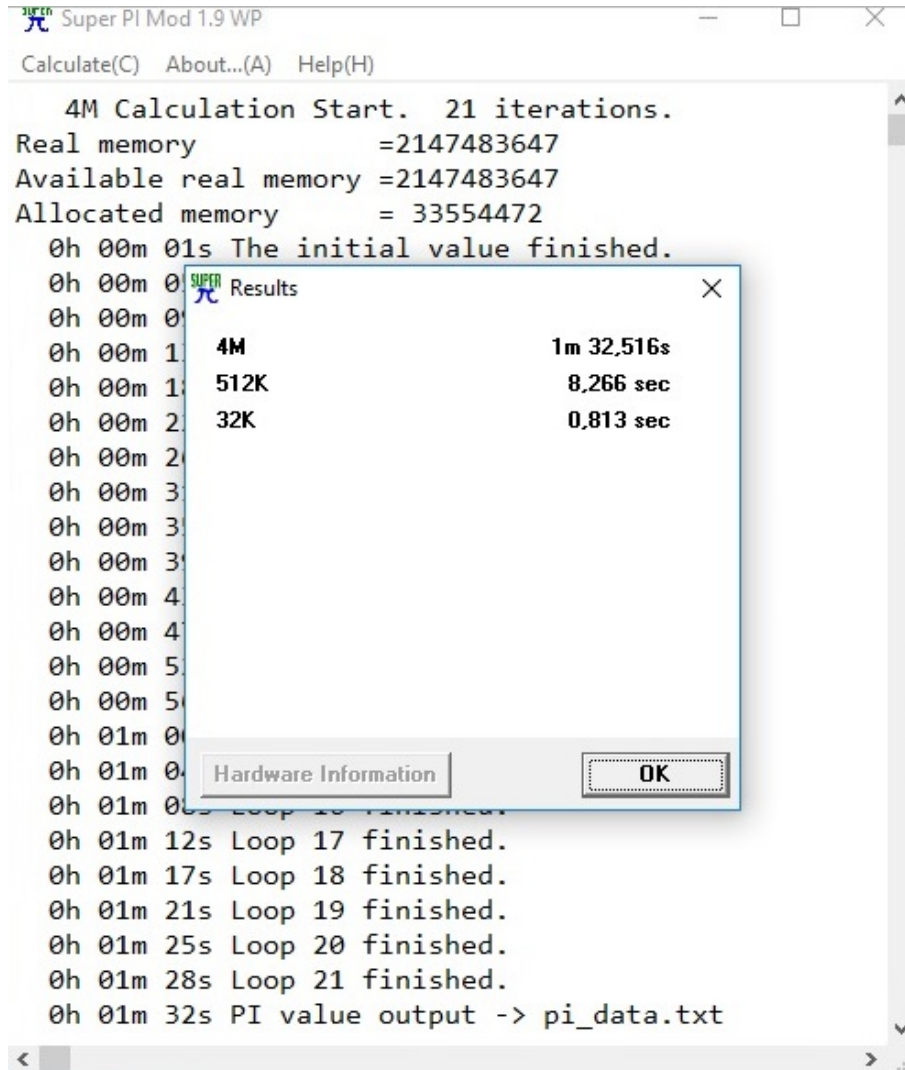
Kuten muillakin ohjelmilla, seuraavassa testissä imitoitiin samalla raskasta palvelimen käyttöä Prime95-ohjelmalla, ja tulokset olivat kymmenellä virtuaalilaitteella seuraavat:

- **ESXi:** 119 sekuntia
- **Hyper-V:** 128 sekuntia
- **Proxmox:** 150 sekuntia

ESXi ja Hyper-V vaikuttivat kärsivän rasituksesta saman verran, kun taas Proxmoxin tulos heikkeni melkein kaksi kertaa huonommaksi. Viimeisenä tulokset, kun virtuaalilaitteita oli käynnissä kaksikymmentä:

- **ESXi:** 135 sekuntia
- **Proxmox:** 151 sekuntia
- **Hyper-V:** 210 sekuntia

Vaikka ESXi suoriutui tästä nopeammin kuin muut, voidaan Proxmoxin tulos luokitella paremmaksi koska se suoriutui testistä vain sekunnin huonommin kuin kymmenen virtuaalilaitteen testillä. Hyper-V:n tulos huononi jälleen runsaasti, ja suoriutui paljon huonommin kuin muut alustat. Tuloksista voidaan todeta, että pääsääntöisesti ESXi suoriutuu yhden ytimen presessoinnista parhaiten. Kuvassa 30 nähdään SuperPi ja siinä suoritettujen testien tulokset.



Kuva 26. SuperPi-suorituskykytesti

## 9 YHTEENVETO

Suoritettut testit osoittivat virtualisointialustojen heikot ja vahvat puolet. Eroja huomattiin jo testausvaiheessa, mutta alustojen vahvuudet saatiin selvästi esille syöttämällä tulokset Excel-ohjelmaan. Vaikka erot olivat pääosin pieniä, kaikista testeistä huonoiten suoriutui Hyper-V. Pääosin Proxmoxin sekä ESXi:n tulokset olivat hyvin tasaisia, mutta vahvuuksia havaittiin yksittäisissä testeissä. Esimerkiksi joissain IOPS-testeissä Proxmox suoriutui ESXi:tä huomattavasti paremmin, kun taas ESXi loisti prosessorin suorituskykytesteissä.

Suorituskykymielessä ei nähty ainakaan tässä skaalassa niin suuria eroja, että valintaa kannattaisi tehdä niiden perusteella. Eroavaisuudet olivat niin pieniä,

että normaalikäytössä eroavaisuuksia ei huomaa. Kuitenkin jos joku kolmesta pitäisi valita tulosten perusteella, se ei ainakaan olisi Hyper-V.

Jos valintaan ottaa muitakin tekijöitä kuin suorituskykytestit, valintani osuisi ESXi:n kohdalle. Proxmox tuntui aluksi helpommalta käyttää kuin ESXi, mutta työn loputtua ESXi:stä jäi sellainen kuva, että sitä olisi miellyttävinnä käyttä näistä kolmesta työympäristössä. ESXi:stä jäi mielikuva, että siinä oli paljon hyödyllisiä ominaisuuksia joita tässä opinnäytetyössä ei käytetty. Hyper-V oli työläämpi asentaa kuin muut alustat, siitä puuttui ominaisuuksia kuten virtuaalilaitteen kloonaus, sekä en henkilökohtaisesti pitänyt sen käyttöliittymästä. Hyper-V:n hallinnointikin tuntui kankealta, ja että siitä ei ole tehty yhtä käyttäjäystävällistä ja yksinkertaista kuin muista alustoista.

Olisi mielenkiintoista nähdä suorituskykytestien tulokset suuremmassa skaalassa, esimerkiksi neljänkymmenen virtuaalilaitteen ollessa käynnissä. Myös hitaamman tai nopeamman palvelimen kanssa olisi helppo verrata tämän opinnäytetyön tuloksiin. Koska tässä opinnäytetyössä ei käytetty mitään erikseen asennettavia paravirtualisointityökaluja, voitaisiin niitäkin verrata keskenään mahdollisessa jatkotutkimuksessa.

## LÄHTEET

Client. s.a. Client Access Licenses and Management Licenses. WWW-dokumentti. Saatavissa: <https://www.microsoft.com/en-us/licensing/product-licensing/client-access-license.aspx> [viitattu 29.3.2018].

Conroy, S. 2018. History of Virtualization. WWW-dokumentti. Päivitetty 25.1.2018. Saatavissa: <https://www.idkrtn.com/history-of-virtualization/> [viitattu 19.3.2018].

Finn, A. 2016. Understanding Windows Server 2016 Licensing. WWW-dokumentti. Päivitetty 13.10.2016. Saatavissa: <https://www.petri.com/understanding-windows-server-2016-licensing> [viitattu 29.3.2018].

Giri, B. 2012. Difference between vSphere, ESXi and vCenter. WWW-dokumentti. Päivitetty 24.8.2012. Saatavissa: <http://www.mustbegeek.com/difference-between-vsphere-esxi-and-vcenter/> [viitattu 23.3.2018].

Golden, B. 2007. Virtualization for Dummies. PDF-dokumentti. Saatavissa: [https://ssl.www8.hp.com/de/de/pdf/virtuallisation\\_tcm\\_144\\_1147500.pdf](https://ssl.www8.hp.com/de/de/pdf/virtuallisation_tcm_144_1147500.pdf) [viitattu 15.3.2018].

Heikkilä, T. 2014. Kvantitatiivinen tutkimus. PDF-dokumentti. Saatavissa: <http://www.tilastollinentutkimus.fi/1.TUTKIMUSTUKI/KvantitatiivinenTutkimus.pdf> [viitattu 12.6.2018].

Hirt, T. 2010. KVM – The kernel-based virtual machine. PDF-dokumentti. Saatavissa: <https://www.cs.hs-rm.de/~linn/fachsem0910/hirt/KVM.pdf> [viitattu 19.3.2018].

Hämäläinen, P. 2007. Verkkovoimaa virtuaalisesti. WWW-dokumentti. Päivitetty 28.11.2007. Saatavissa: <https://www.tivi.fi/Arkisto/2007-11-28/Verkkovoimaa-virtuaalisest-3086464.html> [viitattu 22.3.2018].

Microsoft. 2013. Competitive Advantages of Windows Server 2012 R2 Hyper-V over VMware vSphere 5.5. PDF-dokumentti. Saatavissa: <https://download.microsoft.com/download/E/8/E/E8ECBD78-F07A-4A6F-9401-AA1760ED6985/Competitive-Advantages-of-Windows-Server-Hyper-V-over-VMware-vSphere.pdf> [viitattu 23.3.2018].

Microsoft. 2017. Windows Server 2016 System Requirements. WWW-dokumentti. Päivitetty 17.10.2017. Saatavissa: <https://docs.microsoft.com/en-us/windows-server/get-started/system-requirements> [viitattu 21.3.2018].

Moonsoft. s.a. Microsoft Windows Server 2016. WWW-dokumentti. Saatavissa: <http://www.moonsoft.fi/products/000831.aspx> [viitattu 21.3.2018].

Mäntylä, J.-M. 2008. Virtualisointi mullistaa tietotekniikan. WWW-dokumentti. Päivitetty 30.11.2008. Saatavissa: <https://www.tivi.fi/CIO/2008-11-30/Virtualisointi-mullistaa-tietotekniikan-3158514.html> [viitattu 14.3.2018].

Portnoy, M. 2012. Virtualization Essentials. PDF-dokumentti. Saatavissa: <http://www.tu-varna.bg/tu-varnaknt/images/tutorials/vt/ve.pdf> [viitattu 15.3.2018].

Pricing. s.a. Windows Server 2016 Licensing & Pricing. WWW-dokumentti. Saatavissa: <https://www.microsoft.com/fi-fi/cloud-platform/windows-server-pricing> [viitattu 29.3.2018].

Proxmox. 2018. Main Page. WWW-dokumentti. Päivitetty 24.5.2018. Saatavissa: [https://pve.proxmox.com/wiki/Main\\_Page](https://pve.proxmox.com/wiki/Main_Page) [viitattu 24.3.2018].

Rouse, M. 2017. VMware vSphere. WWW-dokumentti. Saatavissa: <https://searchvmware.techtarget.com/definition/VMware-vSphere> [viitattu 29.3.2018].

Serre, R. 2016. Why moving from Windows Server 2012 R2 to 2016 for Hyper-V. WWW-dokumentti. Päivitetty 16.8.2016. Saatavissa: <https://www.starwindsoftware.com/blog/why-moving-from-windows-server-2012-r2-to-2016-for-hyper-v> [viitattu 21.3.2018].

Tulloch, M. 2010. Understanding Microsoft Virtualization Solutions. PDF-dokumentti. Saatavissa: <https://download.microsoft.com/DOWNLOAD/5/B/4/5B46A838-67BB-4F7C-92CB-EABCA285DFDD/693821EBOOK.PDF> [viitattu 21.3.2018].

VMware. 2009. VMware ESX 4.0 Update 1 Release Notes. WWW-dokumentti. Saatavissa: [https://www.vmware.com/support/vsphere4/doc/vsp\\_esx40\\_u1\\_rel\\_notes.html](https://www.vmware.com/support/vsphere4/doc/vsp_esx40_u1_rel_notes.html) [viitattu 23.3.2018].

VMware. 2018. vCenter Server Appliance Overview. WWW-dokumentti. Päivitetty 26.2.2018. Saatavissa: <https://docs.vmware.com/en/VMware-vSphere/6.5/com.vmware.vsphere.vcsa.doc/GUID-223C2821-BD98-4C7A-936B-7DBE96291BA4.html> [viitattu 23.5.2018].

vSphere-Land. 2009. vSphere Release date. WWW-dokumentti. Päivitetty 17.5.2009. Saatavissa: <http://vsphere-land.com/news/vsphere-release-date.html> [viitattu 23.3.2018].

## KUVALUETTELO

Kuva 1. Tyypin 2 virtualisointialusta.

Kuva 2. Tyypin 1 virtualisointialusta

Kuva 3. Lisenssien hintaerot

Kuva 4. Käyttäjän Client Access License

Kuva 5. Laitteen Client Access

Kuva 6. vSphere tuoteryhmä.

Kuva 7. VMwaren eri lisenssien hintaluokkia.

Kuva 8. Palvelimen POST-screen.

Kuva 9. Hallintaverkon asetukset.

Kuva 10. Proxmoxin komentorivi.

Kuva 11. Virtuaalilaitteen pudotusvalikko, josta virtuaalilaitteet kloonataan.

Kuva 12. Hyper-V Server Managerissa.

Kuva 13. Uuden virtuaalilaitteen asennustyökalu.

Kuva 14. Virtuaalilaitteen kopiointi kiintolevyllä.

Kuva 15. ESXi:n asennuksen varoitus prosessorista.

Kuva 16. ESXi Web Clientin virtuaalilaitteen asennustyökalu.

Kuva 17. vCenter Server Appliance:n graafinen asennustyökalu.

Kuva 18. Suoritettu IOmeter testi.

Kuva 19. Testiprofiilin luonti-ikkuna.

Kuva 20. IOmeterin aloitusruutu.

Kuva 21. Advanced memory testiin syötetyt asetukset.

Kuva 22. Muistin lukunopeustesti ESXi:n sisällä olevalla virtuaalilaitteella.

Kuva 23. Suoritettu UDP testi.

Kuva 24. NovaBench suorituskykytesti.

Kuva 25. Cinebench suorituskykytesti.

Kuva 26. SuperPi suorituskykytesti.