



LAUREA
AMMATTIKORKEAKOULU

Uuden edellä

Virtualisointi ja Hallinta

Siivonen, Tomi

2014 Kerava



Laurea-ammattikorkeakoulu
Kerava

Virtualisointi ja Hallinta

Tomi Siivonen
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Huhtikuu, 2014

Siivonen, Tomi

Virtualisointi ja Hallinta

Vuosi 2014 Sivumäärä 40

Tässä opinnäytetyössä oli tavoitteena tutustua palvelin- ja työpöytävirtualisointijärjestelmiin sekä niiden hallintaan käyttäen ilmaisia, avoimeen lähdekoodiin perustuvia ratkaisuja. Pääpaino työssä oli Fedora Linux 17 -käyttöjärjestelmässä ja sen päälle asennetussa oVirt-virtualisointihallintajärjestelmässä. Virtualisointiteknologiana opinnäytetyössä oli käytössä KVM. Opinnäytetyön aiheen mukaista testausta varten tutkimuksen aikana luotiin oma tietokeskus, johon asennettiin Linux-käyttöjärjestelmiin perustuvia virtuaalikoneita.

Opinnäytetyön yhtenä tavoitteena oli tutkia, miten avoimen lähdekoodin virtualisointijärjestelmät voivat vastata kaupallisten tuotteiden valta-asemaan ja miten tuotteiden ominaisuudet eroavat nykyhetkellä toisistaan. Vertailun vuoksi työssä viitataan Windows Server 2012 Hyper-V-, VMware vSphere- ja VMware vSphere Hypervisor -järjestelmiin. Työn tarkoitus oli tutkia kustannustehokasta, ominaisuuksiltaan riittävää ja kilpailukykyistä ratkaisua palvelinvirtualisointiin.

oVirt-järjestelmään ja avoimen lähdekoodin virtualisointiin tutustuminen antoi paljon lisätietoa uusista mahdollisuuksista toteuttaa virtualisointia edullisemmin. Suoritetut testit osoittivat myös avoimen lähdekoodin sovellusten olevan varteenotettavia vaihtoehtoja kaupallisille tuotteille.

Siivonen, Tomi

Virtualization and Management

Year	2014	Pages	40
------	------	-------	----

The main goal of this Bachelor's thesis is to get information and get familiar with server and work station virtualization as well as managing virtual environments using open source based solutions. The test environment will be built by using Fedora Linux 17 operating system and integrating oVirt virtualization management system into it. A Data center will be built for testing purposes and it will be running Linux operating systems.

A secondary goal is to research how well can open source system compete with commercial products and how different their features are and also make notes about usability. Microsoft Hyper-V and VMware vSphere are used for making the comparisons between commercial and open source products. Cost efficiency is also a part of this thesis.

Getting familiar with the oVirt system and open source virtualization systems gave a lot of new information on the possibilities of implementing low cost virtualization systems. The tests carried out showed that the open source applications can be viable alternatives to commercial products.

Keywords: virtualization, management, oVirt, Linux

Sisällys

1	Johdanto.....	6
2	Termit ja lyhenteet.....	7
3	Tutkimusmenetelmät.....	9
4	Virtualisointi	11
5	Virtualisoinnin hyödyt	15
6	Virtualisoinnin haitat	17
7	oVirt	20
	7.1 oVirt Node	20
	7.2 oVirt Engine	21
8	oVirt järjestelmän asennus.....	22
	8.1 Fedora Linux 17 -käyttöjärjestelmän asennus ja konfigurointi	23
	8.2 oVirt All In Onen asennus ja konfigurointi	27
	8.3 oVirt Noden asennus, konfigurointi ja liittäminen hallintajärjestelmään.....	30
	8.4 Valmiin virtualisointijärjestelmän ominaisuuksien testaus	31
	8.5 Eri virtualisointijärjestelmät vertailussa	33
9	Yhteenveto	34
	Lähteet	37
	Kuvat	39
	Taulukot	40

1 Johdanto

Opinnäytetyön alkuvaiheessa oli kiinnostuksen kohteena aihe, jonka halusin olevan jotain palvelinarkkitehtuuriin liittyvää. Pohdin pitkään opinnäytetyön rajaamista, koska aiheen monipuolisuus aiheutti enemmän päänvaivaa opinnäytetyön laajuudessa kuin ensi alkuun arvasinkaan. Lopulta virtualisointi ja hallinta muodostui omaksi aiheekseni muutaman mutkan kautta ja koin virtualisoinnin ja hallittavuuden tutkimisen mielenkiintoisimmaksi osa-alueeksi palvelinarkkitehtuuriin liittyen.

Aihe on ajankohtainen palvelinpuolella, mutta yhä kasvavassa määrin myös työasema-virtualisointiprojektit ovat kasvattaneet suosiotaan. Vielä on kuitenkin havaittavissa trendi, jossa palvelinvirtualisoinnista saadut hyödyt kuvitellaan saavutettavaksi työasema-virtualisoinnissa, mutta näin asia ei välttämättä ole. Virtualisointiprojekteja viedään läpi hatarin alkutiedoin ja usein projektin tavoite on hämärän peitossa. Palvelinvirtualisointi on helpompi projekti kuin työasemavirtualisointi, mutta molemmissa on omat ongelmansa. Opinnäytetyön pääpaino on tutkia palvelinvirtualisoinnin mahdollisuuksia, mutta samoin menetelmin on mahdollista toteuttaa pienen mittakaavan työasemavirtualisointiprojekti.

Kaupallisten tuotteiden etulyöntiasema markkinoilla on johtunut erityisesti hallittavuuden ja tuen tarjonnasta. Tukipalvelut ja lisenssit tuottavat kuitenkin lisäkuluja, joita yritykset haluavat nykyisessä taloustilanteessa kaikin järkein keinoin välttää. Hallittavuus ei kuitenkaan enää ole pelkästään kaupallisten tuotteiden erikoisominaisuus, joten avoimen lähdekoodin tuotteet ovat nykyään helpompi valinta kuin aiemmin. Tutkimusta varten etsin huolellisesti eri lähteistä avoimeen lähdekoodiin perustuvia hallintajärjestelmiä, joita oli lopulta suhteellisen hankala löytää.

Pitkän etsimisen tuloksena vastaan tuli oVirt 3.1 -virtualisointihallintajärjestelmä. Se on tarkoitettu Fedora Linux -käyttöjärjestelmän päälle asennettavaksi, joten tätä kautta Fedora Linux 17 -käyttöjärjestelmä tuli mukaan opinnäytetyöhön. oVirt 3.1 on Java-ohjelmointikielen perustuva internetselaimella hallittava virtualisoidun tietokeskuksen hallintajärjestelmä, jossa on huomattavan paljon ominaisuuksia. Yleisesti ottaen avoimen lähdekoodin tai ei-kaupallisten virtualisointijärjestelmien hallintatoiminnot ovat olleet hyvin rajoitettuja tai jopa olemattomia. Taustalla oVirt-järjestelmässä on Red Hat Enterprise Virtualization (RHEV), jota on alettu tuottamaan avoimeen muotoon. Virtualisointitekniikana toimii Kernel-based Virtual Machine (KVM) hypervisor.

Tässä opinnäytetyössä tutkimus on rajattu palvelin- ja työpöytävirtualisointijärjestelmiin ja niiden hallintaan käyttäen avoimeen lähdekoodiin perustuvia ratkaisuja. Pääpaino opinnäytetyössä on Fedora Linux 17 -käyttöjärjestelmä ja sen päälle asennettu oVirt 3.1 -

virtualisointihallintajärjestelmä. Opinnäytetyössä tutkitaan myös kustannustehokasta ratkaisua virtualisointiin. Alkuosa opinnäytetyöstä keskittyy käsittelemään yleisellä tasolla virtualisointia ja sen hallintaa. Opinnäytetyön loppuosassa rakennetaan testausympäristö oVirt-hallintajärjestelmän ominaisuuksien testaamista varten.

2 Termit ja lyhenteet

AMD-V	Advanced Micro Devices (AMD) prosessoriperheen virtualisointituki.
CPU	Central Processing Unit. Prosessori eli suoritin.
Data domain	oVirt järjestelmässä myös shared storage. Tietovarasto.
FCP	Fibre Channel Protocol. Käytetään myös termiä FC.
Fedora	Linux-pohjainen käyttöjärjestelmä.
GB / TB	Gigabyte / terabyte. Käytetään muisti- ja tallennuskapasiteetin mittaamiseen.
Ghz	Gigahertz. Tässä opinnäytetyössä Ghz kuvaa prosessorin kellotaajuutta.
GPU	Graphics Processing Unit. Näytönohjain.
Guest	Termiä käytetään yleensä puhuttaessa virtuaalikoneesta.
HA	High availability eli korkea palvelutaso, jonka mahdollistaa vikasietoisuuden huomioiminen suunnittelussa.
Host	Termiä käytetään yleensä puhuttaessa isäntäkoneesta. Fyysinen palvelin.
Hyper-V	Microsoftin kehittämä virtualisointijärjestelmä.
Hypervisor	Ohjelma, joka hoitaa virtuaalikoneiden kommunikaation fyysisen laitteiston kanssa.
IBM	International Business Machines. Laite- ja ohjelmistovalmistaja.
Intel VT	Intel Virtualization Technology. Intel prosessoriperheen virtualisointituki.
IP-osoite	Internet Protocol address. Yksilöivä verkko-osoite tietoverkossa oleville laitteille.
iSCSI	Internet Small Computer System Interface. Välittää SCSI paketteja verkossa käyttäen TCP/IP protokollaa.
ISO	Tiedostoksi muunnettu levykuva.
Java	Ohjelmointikieli.
Jboss	Ohjelmistokehitysympäristö.
Kernel	Linux-käyttöjärjestelmän ydin.
Klusteri	Palvelimista koostuva yhteenliittymä, joka parantaa luotettavuutta ja tehoa.
KVM	Virtualisointitekniikka.
kWh	Kilowatt hour eli kilowattitunti. Käytetään sähköenergian määrän mittaamiseen.
Linux	Avoin ja ilmainen käyttöjärjestelmä.
Live migration	Virtuaalikoneen siirtäminen isäntäkoneelta toiselle ilman käyttökatkoksia.

NFS	Network File System. Käytetään tiedostojen jakamiseen tietoverkoissa.
NTP	Network Time Protocol. Käytetään aikatiedon synkronointiin tietoverkoissa.
oVirt Engine	oVirt-järjestelmän hallintaosa.
oVirt Node	oVirt-järjestelmän isäntäkone tai palvelin, jota hallitaan oVirt Enginen hallintajärjestelmän kautta.
Palvelin	Alusta palvelun suorittamiselle ja sen tarjoamiselle. Voi olla fyysinen laite tai virtuaalikone.
Palvelinsali	Tila jossa fyysiset palvelimet sijaitsevat. Yleensä varustettu jäähdytyksellä, sammutuslaitteistolla, kahdennetulla tietoverkolla ja muilla ominaisuuksilla korkean palvelutason säilyttämiseksi.
QEMU	Generic and open source machine emulator and virtualizer. Prosessoriemulaattori.
RAID	Redundant Array of Independent Disks. Mahdollistaa eri tasoisia keinoja varautua kiintolevyjen vioittumiseen.
RemoteFX	Microsoftin kehittämä tekniikka, joka mahdollistaa palvelimelle asennetun näytönohjaimen paremman hyödyntämisen etäyhteyden kautta.
Replikointi	Laitteiston tai tiedon kahdentaminen, jolla tavoitteena on vikasietoisuuden ja tehokkuuden parantaminen.
RHEV	Red Hat Enterprise Virtualization. Virtualisointijärjestelmä.
Root	Tässä opinnäytetyössä tällä termillä tarkoitetaan Linux-käyttöjärjestelmän pääkäyttäjää.
SAS	Serial Attached SCSI. Palvelinkäyttöön tarkoitettu liitäntävyölä kiintolevyille.
Sertifikaatti	Mahdollistaa tietojärjestelmän aitouden varmistamisen ja käytetään pääsääntöisesti salatun liikenteen varmistukseen.
Snapshot	Tilannekuva järjestelmästä. Voi tarkoittaa tilannekuvaa tallennusjärjestelmän tiedosta tai virtuaalikoneesta.
SSH	Tapa muodostaa salattu etäyhteys tietokoneiden välille.
Tietokeskus	Data center. Voi tarkoittaa fyysisistä laitteistoista muodostuvaa palvelinsaliin verrattavaa pienempää laitteistokokonaisuutta tai virtualisoitua laitteistokokonaisuutta.
UPS	Uninterruptible Power Supply. Tarkoitetaan laitteistoa, jonka tarkoitus on auttaa sähköverkon häiriötilanteita vastaan.
VDSM	Virtual Desktop and Server Manager. VDSM hallitsee ja tarkkailee virtuaalijärjestelmän resursseja fyysisissä laitteissa ja virtuaalikoneissa.
Virtuaalikone	Virtualisoitu palvelinlaitteisto ja siihen asennettu käyttöjärjestelmä.
Virtualisointi	Fyysisen laitteiston muuttamista ohjelmalliseksi.
vCenter	VMwaren kehittämä hallintajärjestelmä vSphere virtualisointijärjestelmälle.
vMotion	VMwaren kehittämä keino siirtää virtuaalikone isäntäkoneelta toiselle ilman palvelukatkoksia.
vSphere	VMwaren kehittämä virtualisointialusta.
Xeon	Prosessorivalmistaja Intelin nimitys palvelinprosessoreista.

3 Tutkimusmenetelmät

Tämän opinnäytetyön projektiluontoisuuden vuoksi tutkimusmetodiksi valittiin asiantuntija-arvioinnin ja toimintatutkimuksen yhdistelmä, jotka kuuluvat laadullisiin tutkimusmenetelmiin. Toimintatutkimuksen ja asiantuntija-arvioinnin välinen ero on lähinnä marginaalinen. Toimintatutkimuksessa asiantuntija osallistuu tutkimusryhmän toimintaan tai antaa arvion itse. Asiantuntijan omat ennakkokäsitteet ja mielipiteet saattavat helposti vaikuttaa myös lopputulemaan, joten useamman asiantuntijan käyttäminen toimintatutkimuksessa tuottaa usein tutkimuksen, jossa tutkittavaa asiaa saadaan helpommin arvioitua useammalta kantilta ja samalla vaikutetaan tutkimustuloksen validiteettiin. Määrällisillä menetelmillä saataisiin ilmiö kenties paremmin esiin, mutta laadullisella tutkimuksella voidaan ilmiötä tutkia ja ymmärtää paremmin. Nimensä mukaisesti menetelmä kuvaa ilmiön laadullisia arvoja ja eroja. (Kuula 1999, 9-16, 129-142, 218-228.)

Laadullinen eli kvalitatiivinen tutkimus on havainnointiin ja haastatteluun perustuvaa tutkimusta, joten se sopii hyvin tämän opinnäytetyön tutkimusmenetelmäksi. Kyseessä voi olla osallistuvaa tai ulkopuolista havainnointia, joiden avulla voi saada hieman erilaista tutkimusaineistoa. Havainnointi voi perustua kuva-, ääni- ja videomateriaalin keräämiseen, joilla voidaan verrata havainnoidun asian todellista toimintatapaa oletettuun toimintatapaan. Yleisin haastatteluun perustuva tutkimusmuoto on kyselylomake, jossa tutkittavaa pyydetään kuvailemaan vapaasti tai skaalalla tutkittavia ilmiöitä. Asiantuntija-arvioinnissa voidaan myös käyttää lomakepohjaan tehtyä arviointikaavaketta helpottamaan tutkimustulosten kirjaamista, vaikka asiantuntijoita olisi vain yksi. Tärkeintä on huomioida kaikki mahdolliset ongelmat tai onnistumiset, jotta tutkimuksen raportoinnissa saadaan kirjattua ja analysoitua mahdollisimman kattavasti kaikki tutkimuksen tulokset. Laadullinen tutkimus antaa tutkijalle kohderyhmän mielipiteen tai odotuksen tutkittavasta aiheesta. (Saaranen-Kauppinen & Puusniekka 2006a.)

Suunnittelu on suuri osa kaikkea toimintaa. Tutkimussuunnitelman tekemisestä on laadullisen tutkimuksen prosessissakin aloitettava. Tutkimussuunnitelma pitää sisällään aina tutkittavan aiheen ja tutkimusongelman muotoilun, tiedon keräämisen suunnitelman ja tutkimuksen analysoinnin mietinnän. Aikataulun ja resurssien kirjaaminen kuuluu hyvään tutkimussuunnitelmaan. Ensin on yksilöitävä ja rajattava tutkittava kohde ja mahdolliset ongelma-alueet, jotta tutkimus osataan kohdistaa oikeaan alueeseen. Aikaisemmat aiheen tutkimustulokset ja kirjallisuus antavat tietoa ongelmista ja mahdollisuuksista. Tutkittavat asiat ja olettamukset määritellään ymmärrettävyyden parantamiseksi ja mahdollisten ongelmien välttämiseksi. Tämän jälkeen voidaan siirtyä tiedonkeruuvaiheeseen. Tutkimuksen päätyttyä aineisto kerätään yhteen, luokitellaan ja järjestellään huomioiden laadunvalvontaan liittyvät ongelmat. Viimeisenä vaiheena on johtopäätösten tekeminen tutkimusmateriaalista,

tutkimustulosten soveltaminen ja tieteellinen päättely. (Saaranen-Kauppinen & Puusniekka 2006b.)

Laadullisen aineiston analysointi on yleensä hyvin haastavaa sen runsaan määrän ja erilaisten vastausten takia. Tulosten luotettavuuden arviointi on hankalampaa kuin määrällisessä tutkimuksessa. Laadullisessa tutkimuksessa ei yleensä ole lukuja, vaan se pyrkii olemaan kuvaustyyppinen. Aineiston keräämisen yhteydessä onkin syytä pyrkiä huomioimaan kerätyn aineiston määrä ja huomaamaan piste, jossa lisäaineisto ei tuo tutkimukselle enää lisäarvoa. Analysointi on tutkimusaineiston tulkintaa ja analysoinnin tarkoitus on antaa jonkinasteinen järjellinen yhteenveto tutkimusmateriaalista. Kyse on siis aineiston tarkasta ja huolellisesta tarkastelemisesta. Laadullisen materiaalin tarkasteleminen vaatii sen kiteyttämistä ja huolellista tulkitsemistä, minkä takia on vältettävä tutkimusaineiston tulkittamista väärin. Analysoinnilla tutkimusaineiston joukosta yritetään etsiä vastausta tutkimusongelmaan tutkijan materiaalista saamien havaintojen kautta. (Walker 2004.)

Tutkimusaineistoa itsessään ei esitetä raportissa, mutta siinä pyritään esittämään aineistosta ilmi tulleita, tutkimusongelmaa selittäviä asioita. Tutkija tuo raportissa esiin oman roolin tiedonhankinnassa, tulosten analysoinnissa ja ratkaisuiden tekemisessä ja arvioi sen huomioiden mahdolliset eettiset ongelmat. Eettisillä ongelmilla voidaan tarkoittaa tiedon hankintaan, tutkittavien suojaan ja tutkimuksen soveltamisen ongelmiin. (Kuula 2006, 11-39.)

Ongelmilta voidaan välttyä, jos tieto ei ole kohdistettavissa kovin hyvin keneenkään yksilönä, mutta tämä saattaa myös hankaloittaa tutkimusta. Eettisillä ongelmilla voidaan tarkoittaa tieteelliseltä kannalta katsottuja ongelmatilanteita, eli esimerkiksi, onko oikein tutkia tätä aihetta ja mihin tutkimustietoa voidaan käyttää. Nämä ongelmat lienevät normaalissa tutkimustilanteessa information technology (IT)-alalla hieman harvinaisia. Sovellettavuus, uskottavuus ja vahvistettavuus ovat kriteerejä laadullisen tutkimuksen validiteetin määrittelyssä. Usein näistä juuri vahvistettavuus eli useiden tutkimusten samaa tulosta osoittaminen antaa hyvän kuvan tutkimuksen luotettavuudesta. (Kuula 2006, 200-219.)

Tämän opinnäytetyön tavoite on tuottaa hyötyä itselleni osaamisen kehittymisessä ja sidosryhmälle tietotaidon lisääntymisenä. Sidoryhmäyhteistyö on IT-alalla tärkeää ja on kaikille erittäin hyödyllistä omata laaja-alainen sidoryhmä, josta voi tarvittaessa kysellä apuja ongelmiin ja tarjota omaa osaamistaan vastineeksi. Yleistä hyötyä opinnäytetyöstä voi saada tahot, jotka vertailevat oman virtuaalipalvelimen pystyttämistä erilaisia tuotteita hyödyntämällä. Tutkimus- tai kehittämisiongelman voisi tässä tapauksessa muotoilla muotoon: ”Onko avoimen lähdekoodin virtualisointijärjestelmästä kodin tai yrityksen virtualisointiratkaisuksi?” tai ”Voiko kaupallisen virtualisointijärjestelmän korvata avoimen lähdekoodin järjestelmällä?”.

Lähestymistavan ja tiedonkeruumenetelmän soveltuvuuden arvioimiseksi tarkastelin opinnäytetyön aihetta monelta kantilta. Päällimmäisenä ajatuksena oli kuitenkin projektimuotoisuus, joten laadullinen tutkimus toimintatutkimuksen ja asiantuntija-arvioinnin kautta auttaa opinnäytetyötä olemaan monipuolisempi ja kattavampi kuin määrällisen tutkimuksen avulla voitaisiin saavuttaa. Näiden tutkimusmetodien avulla opinnäytetyö ja sen tutkimustulos pystyy antamaan lukijalle tarkempaa tietoa ratkaisuvaihtoehtojen eroista, yhtäläisyyksistä ja mahdollisuuksista. On kuitenkin huomioitava, että tietotekniikan alueella laadullisen tutkimusaineiston luotettavuuteen vaikuttaa huomattavasti testaaajien henkilökohtaiset ennako-odotukset ja tottumukset.

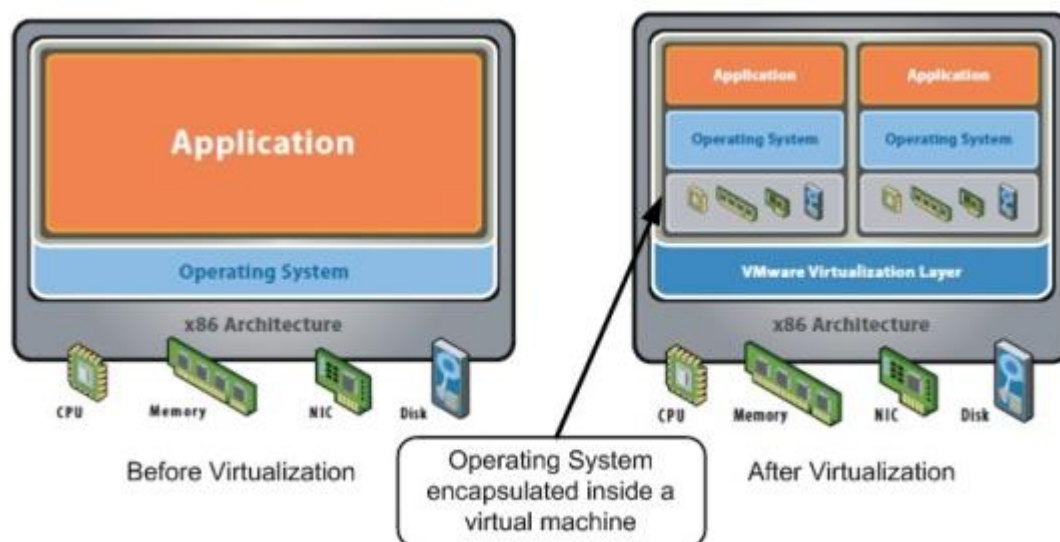
4 Virtualisointi

Virtualisoinnilla tarkoitetaan jonkin fyysisen elementin suorittamista ohjelmallisena ja sen määritelmä on vuosien varrella muuttunut tietotekniikan kehittyessä eteenpäin. Yleisimmin virtualisoinnin kohteena on palvelimet, mutta levyjärjestelmien, verkkolaitteiden ja käyttöjärjestelmien virtualisointi on mahdollista. Virtualisoinnilla voidaan yhdessä fyysisessä palvelimessa (host) suorittaa yhtä tai useampaa virtualisoitua palvelinta (guest), jolloin nykyisten tehokkaiden palvelinten resurssit saadaan paremmin hyötykäyttöön. Laitteisto-resurssien paremman hyödyntämisen lisäksi virtualisoinnilla saadaan helpotettua IT-henkilöstön työtaakkaa laitteiston hallinnan osalta. (Graziano 2011.)

Palvelimissa suoritetaan yleensä vain yhtä palvelua, joten resurssihukka saattaa olla hyvinkin suurta. Tämä on kuitenkin aina tapauskohtaista ja riippuu suuresti palvelimen tarkoituksesta. Tiedostopalvelin kuormittaa lähes pelkästään verkko- ja levykapasiteettia, mutta ei prosessoria. Tulostinpalvelin taas ei kuormita palvelinta lähes ollenkaan. Tietokannat kuormittavat keskusmuistia, prosessoria ja levykapasiteettia. Tässä esitetyt esimerkit voitaisiin hyvin todennäköisesti virtualisoida yhdelle palvelimelle resurssipulaa aiheuttamatta, vaikkakin tietokannoille yleensä suositellaan omaa palvelinta.

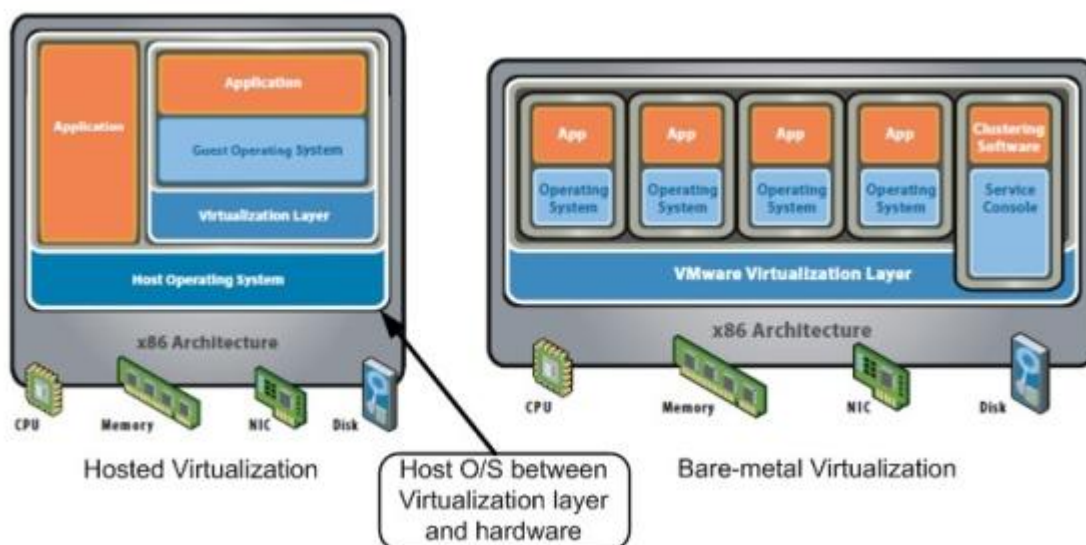
Virtualisointiohjelma (hypervisor) jäljittelee tietokoneen fyysisiä ominaisuuksia ja mahdollistaa useamman virtuaalisen tietokoneen toiminnan yhdessä fyysisessä palvelimessa. Virtuaalikone näkee resurssit virtualisointiohjelman kautta ja virtualisointiohjelma ohjaa resursseja tarpeen ja ennalta määriteltyjen sääntöjen mukaisesti virtuaalikoneille. Virtualisointiohjelma samalla varmistaa, että virtuaalikoneet eivät häiritse toistensa toimintaa. (Siebert 2009.)

Kuvassa 1 vasemmalla puolella kuvaus on perinteisestä palvelinmallista, jossa käyttöjärjestelmä on suoraan kiinni laiteresursseissa. Oikealla puolella kuvassa 1 on kuvaus siitä, miten virtualisointikerros tulee laitteiston ja käyttöjärjestelmien väliin. (Siebert 2009.)



Kuva 1. Perinteinen palvelin ja virtualisoitu palvelin (Siebert 2009.)

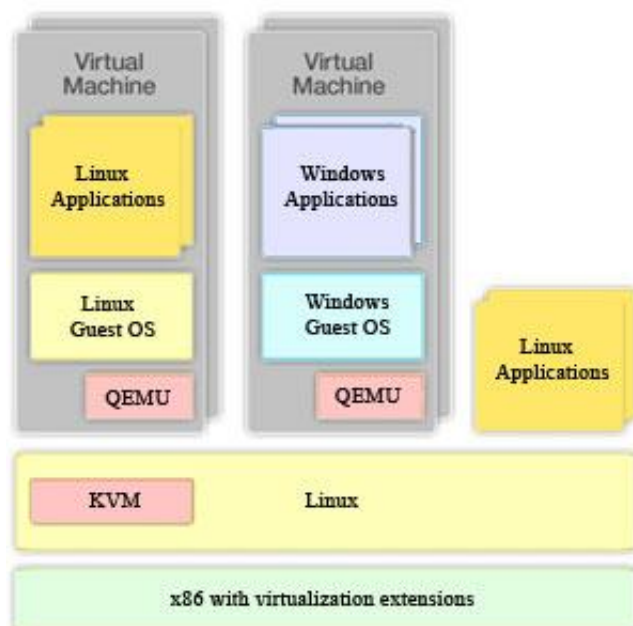
Virtualisointikerroksen luomiseen on kaksi eri lähestymistapaa. Fyysiselle palvelimelle voidaan asentaa hypervisor, joka keskustelee suoraan fyysisten komponenttien kanssa (kuvassa 2 oikealla). Näitä ovat esimerkiksi: VMware ESX, VMware vSphere, Citrix XenServer ja Microsoft Server Hyper-V. Toisessa vaihtoehdossa (kuvassa 2 vasemmalla) virtualisointikerros muodostuu käyttöjärjestelmän päälle, jolloin koneessa voidaan suorittaa virtuaalikoneiden lisäksi käyttöjärjestelmän ominaisuuksia ja ohjelmia. Yleisimmät työasemakäytössä olevat virtualisointiohjelmat ovat asennettuna käyttöjärjestelmän päälle. Näitä ovat esimerkiksi: VMware Workstation ja VMware Player, Oracle Virtualbox ja Microsoft Virtual PC. Tässä opinnäytetyössä käsitelty oVirt-virtualisointijärjestelmä käyttää Fedora Linux - käyttöjärjestelmää isäntäkäyttöjärjestelmänä, mutta sen virtualisointiosa KVM on osa Linuxin ydintä (kernel) eli se kykenee keskustelemaan suoraan laitteiston kanssa. Toteutustavasta riippuen järjestelmä on eräänlainen hybridiratkaisu ja KVM:n voi sijoittaa kumpaankin kategoriaan. (Siebert 2009.)



Kuva 2. Kaksi erilaista tapaa suorittaa virtualisointi. (Siebert 2009.)

Generic and open source machine emulator and virtualizer (QEMU) puolestaan hoitaa virtuaalikoneen keskustelun isäntäkoneen prosessorin kanssa saavuttaen lähes natiivin suorituskyvyn. Suurin hyöty saavutetaan tapauksissa, joissa käytetään samaa prosessoriarkkitehtuuria sekä isäntäkoneessa että virtuaalikoneessa. (Qemu.)

International Business Machines (IBM) esittää asian hieman selkeämmin kuvassa 3. Heidän mallissa KVM muuttaa Linuxin ytimen bare-metal hypervisoriksi, jolloin virtuaalikone saa suoraan käyttöön isäntäkoneen resurssit. KVM mahdollistaa virtuaalikoneen toiminnan Linuxin prosessina, jolloin virtuaalikone pystyy keskustelemaan suoraan laitteiston kanssa. (IBM.)



Kuva 3. Virtualisointi Linuxissa. (IBM.)

Edellisen kaltaisella ratkaisulla saadaan käyttöön Linuxin ytimen tarjoama suorituskyky, tehokkuus, skaalattavuus ja turvallisuus. Ratkaisumalli tarjoaa myös muita etuja. Linuxin tapaan KVM on kustannustehokas, sillä se ei vaadi ohjelmistolisenssiä tai kalliita tukipalveluita. Asiansa osaavan ylläpitäjän se vaatii, mutta osaavat asiantuntijat ovat vaatimuksena muissakin virtuaaliratkaisuissa, jos ei haluta ulkoistaa toimintoa ulkopuolisille tahoille. (IBM.)

Linux ja KVM antaa mahdollisuuden pysytellä itsenäisenä verrattuna kaupallisten kilpailijoiden tuotteisiin. Kaupalliset kilpailijat pyrkivät saamaan organisaation pysymään heidän asiakkaina tarjoamalla joskus jopa pakollisia tukipalveluja ja rajoittamalla yhteensopivuutta muiden virtualisointialustojen kanssa. Pienenä poikkeuksena sääntöön on VMware, joka tarjoaa työkalun virtuaalijärjestelmien siirtoon järjestelmästä toiseen. Avoimen lähdekoodin ohjelmistot lähestyvät jatkuvasti käytettävyydeltään maksullisia ohjelmistoja, mutta tuotekehityksen ja resurssien puutteellisuudesta johtuen ne tulevat todennäköisesti aina olemaan kehityksestä jäljessä. Avoimen lähdekoodin ohjelmistojen kehityssuunta on kuitenkin selkeä ja takaa-ajo on alkanut. Näyttääkin siltä, että välimatka on hieman kutistunut ainakin virtualisoinnin saralla.

5 Virtualisoinnin hyödyt

Virtualisointijärjestelmät kehittyvät jatkuvasti ja ominaisuuksia parannetaan kohtamaan laitteistojen ja ohjelmistojen vaatimukset. Eri virtualisointijärjestelmät ja niiden versiot tukevat hyvinkin erilaisia palvelinlaitteiston ja virtuaalikoneiden konfiguraatioita. Windows Server tuotteiden kehitys esitetään taulukossa 1 ja sama kehityssuunta on havaittavissa myös muissa virtualisointijärjestelmissä.

Windows Server 2012 tukee isäntäkoneessa 320 loogista suoritinta ja 4 TB fyysistä muistia. Nämä fyysiset ominaisuudet ovat jaettavissa virtuaalikoneiden kesken siten, että yhdelle virtuaalikoneelle voidaan antaa käyttöön jopa 64 suoritinta ja 1 TB muistia. Yleensä pienempikin määrä riittää. Microsoft on määritellyt aktiivisten virtuaalikoneiden määräksi 1024, jolloin jokaiselle virtuaalikoneelle olisi mahdollista määrittää kaksi virtuaalista suoritinta ja noin 1GB muistia. KVM ja RHEV kehittyvät myös jatkuvasti ja vuoden 2012 loppupuolella KVM tuki 160 loogista suoritinta ja 2 TB fyysistä muistia. RHEV tuki vielä suurempaa määrää ja suunta on jatkuvasti kasvussa uudempien versioiden mukana. Tämä suunta siis tarkoittaa, että palvelimien tehot kasvavat ja niissä voidaan suorittaa yhä suurempaa ja suurempaa määrää virtuaalikoneita ja niille voidaan tarjota entistä enemmän resursseja käytettäväksi.

Järjestelmä	Resurssi	Windows Server 2008 R2	Windows Server 2012	Kehityskerroin
Isäntä	Loogisia suorittimia	64	320	5x
	Fyysistä muistia	1 TB	4 TB	4x
	Virtuaalisia CPU:eja per Isäntä	512	2,048	4x
Virtuaalikone	Virtuaalisia CPU:eja per VM	4	64	16x
	Muistia per VM	64 GB	1 TB	16x
	Virtuaalilevyn maksimikoko	~2 TB	64 TB	32x
	Actiivisia virtuaalikoneita per isäntä	384	1,024	2.7x
	NUMA asiakas	Ei	Kyllä	-
Klusteri	Solmujen maksimimäärä	16	64	4x
	Virtuaalikoneiden maksimimäärä	1,000	8,000	8x

Taulukko 1. Windows Server tuotteiden kehitys. (Microsoft 2012.)

IT-osaston hallinnassa voi olla useita fyysisiä palvelimia ja niiden kapasiteettia voidaan jakaa siirtämällä virtuaalikoneita palvelimelta toiselle. Tämänkaltaisella toteutusratkaisulla voidaan seurata fyysisten laitteiden kuormitusta ja saada täysi hyöty laitteista. Samalla virtuaali-järjestelmän vikasetoisuus eli high availability (HA) kasvaa, koska hallintajärjestelmä yrittää

pitää virtuaalikoneet aina käynnissä jollain tarjolla olevalla toimivalla palvelimella, vaikka alkuperäinen isäntäkone menettäisi jostain syystä yhteyden hallintajärjestelmään.

Perinteisessä palvelinratkaisussa, jossa yhdessä fyysisessä palvelimessa on yksi käyttöjärjestelmä, on usein ongelmana tyhjäkäyttö. Palvelimen kaikkea kapasiteettia ei siis usein saada kokonaisuudessaan käyttöön. Tyhjäkäyttö aiheuttaa turhaa virrankulutusta, joka muodostaa merkittävän kuluosan palvelimen kokonaiskustannuksia laskettaessa. Jossain tapauksissa palvelimen koko elinkaaren eli käyttöajan sähkölasku olla suurempi kuin kyseisen laitteen hankintahinta. Virtualisointi antaa tähän ongelmaan oman ratkaisunsa. Samassa palvelimessa voidaan suorittaa samanaikaisesti useita käyttöjärjestelmiä, joissa on käynnissä palvelut, jotka olivat aiemmin käynnissä omilla fyysisillä palvelimillaan. Näin ei tarvita useita laitteita vaan yhdessä fyysisessä laitteessa voidaan ajaa tapauksesta riippuen useitakin virtuaalikoneita. (Rankin & Hill 2010, 296.)

Suuremmissa mittakaavassa voidaan myös hyötykäyttää fyysisten laitteiden hukkalämpö ja lämmittää sillä asuin- tai toimistorakennuksia. Academica Oy ja Helsingin Energia ovat yhteistyössä kehittäneet järjestelmän, jossa lämmöksi muuttunut sähköenergia saadaan kerättyä talteen. Tietokeskuksen palvelimet, levyjärjestelmät ja verkkolaitteet ovat sijoitettuna Helsingin Energian luolaan, jonne tuodaan jäähdytystä varten kylmää vettä. Jäähdytysprosessissa kylmä vesi lämpenee ja saadaan muutettua lämmitysenergiaksi kaukolämpöverkkoon. Helsingin Katajanokalla sijaitsevassa luolassa on tilaa parille sadalle palvelinkaapille ja täyden palvelinsalin hukkalämmöllä voi lämmittää jopa 500 omakotitaloa. Kaukojäähdytys on pienentänyt tässä tapauksessa sähkölaskua 150000 euroa eli taloudellinen hyöty on merkittävä. (Tuominen-Halomo 2010, 5.)

VMware esitti energiatehokkuudesta omia laskelmiaan vuonna 2008. Laskelmien mukaan 40 fyysisen palvelimen toimintojen on laskettu hoituvan neljällä fyysisellä palvelimella, joissa suoritetaan samat 40 palvelinta virtualisoituna. Fyysisien palvelinten ja niiden jäähdytykseen kuluvan sähkön yhteenlasketun kulutuksen arvioitiin vuositasolla olevan noin 41,4 kilowattituntia (kWh). Virtualisoidun järjestelmän kulutus arvioitiin olevan vain 4,6 kWh eli kulutus putoaisi noin kymmenesosaan virtualisoinnin avulla. Sähköenergian hinnan ollessa kuusi snt/kWh, olisi vuosittainen säästö sähköenergian osalta 19482 euroa. (Niemistö 2008.)

Pohjolan ilmasto sopii konesaleille erinomaisesti. Täällä on tarjolla tietokeskuksen jäähdytystarpeisiin kylmää vettä ja ilmaa. Tästä syystä yritykset kuten Facebook ja Google ovat sijoittaneet palvelinsaleja Suomeen ja Ruotsiin. Lentokonevalmistaja Airbusin tietohallintojohtaja Guus Dekkers arvioi yhtiönsä sähkölaskun vuonna 2020 olevan kymmenen miljoonaa euroa. Tästä syystä Airbus harkitsee muiden yritysten tavoin tietokeskuksen rakentamista kylmän ilmaston maihin. (Leidenius 2012, 22-23.)

Virtualisoinnilla saavutettavista hyödyistä yksi suurimpia on virtuaalikoneiden nopea ja käytännöllinen varmistus. Virtuaalikoneista voidaan ottaa snapshot eli tilannekuva, joka mahdollistaa koko virtuaalipalvelimen varmuuskopioinnin kerralla. Perinteisesti varmistuksen piirissä on vain palvelimella oleva tieto, mutta tällä metodilla voidaan varmistaa koko virtuaalikone ja ongelmatilanteessa tilannekuva voidaan palauttaa nopeasti käyttöön joko samaan tai toiseen virtualisointijärjestelmään. (Rankin & Hill 2010, 321-322.)

Isossa tietoverkossa olevan toimipisteen virtuaalikoneesta otettu varmuuskopio voidaan palauttaa vaikka toisessa maassa olevaan virtuaalipalvelimeen ja käynnistää se siellä. Samassa tietoverkossa palvelin käyttäytyy samoin kuin se olisi paikallinen. Tietoverkon viiveet vaikuttavat järjestelmän käytettävyyteen, mutta ongelmatilanteen selvittyä järjestelmä voidaan taas siirtää takaisin paikalliseen virtuaalipalvelimeen ja tilanne palautuu normaaliksi. Tämä kaltainen joustavuus mahdollistaa myös suurempien keskitettyjen tietokeskusten käyttämisen. Palvelimia voidaan pitää yhdessä tietokeskuksessa ja tarvittaessa siirtää palvelimia helposti parempien verkkoyhteyksien päähän. Siirrettävyyden lisäksi virtualisoinnin turvana on hyvä käyttää replikointia eli kahdentamista. Fyysisiä palvelimia voi lisäksi klusteroida eli määrittää ne toisilleen eräänlaisiksi varalaitteiksi. Yksi hyvä käytäntö on replikoida tallennusjärjestelmät, jolloin tieto on aina toisella tallennusjärjestelmällä vaikka toinen hajoaisi. Lisäturvaa järjestelmän varmistukselle luo myös sähköjärjestelmän varmistaminen Uninterruptible Power Supply (UPS)-järjestelmällä ja polttoainekäyttöisillä sähkögeneraattoreilla. Tietoverkon tavoitettavuus on hyvä varmistaa eli luoda varmistuksia kaapeli- ja laiterikoille käyttämällä useampia yhteyksiä tärkeiden laitteistojen välillä ja kahdentamalla laitteisto.

6 Virtualisoinnin haitat

Virtualisoinnin suurista hyödyistä huolimatta on olemassa myös muutamia ongelmakohtia, joita ei välttämättä tule otetuksi huomioon suunnitteluvaiheessa. Virtualisoinnin suunnittelu kannattaisi aloittaa virtualisoinnin kohteesta ja valita käytettävä virtualisointijärjestelmä sen mukaan. Usein kuitenkin sorrutaan teknologian valintaan ja unohdetaan tavoite eli tietojärjestelmän toiminnan varmistaminen. Suunnitteluvaiheeseen kannattaakin ottaa mukaan testivaihe, joka auttaa virtualisointijärjestelmän valinnassa. Palvelinvirtualisoinnin yksi lähtökohta voisi olla palveluiden määrittely. Palveluiden määrittelyn pohjalta voidaan arvioida virtualisointialustaan kohdistuvat vaatimukset nyt ja tulevaisuudessa.

Mikäli tarve on vain palvelimien ja varsinkin Linux palvelimien virtualisointiin, voi Linux pohjainen virtualisointijärjestelmä olla hyväkin vaihtoehto. Jos taas tarpeena on virtualisoida työasemia ja varsinkin graafista suunnittelua tai videotyöskentelyä varten tehtyjä työasemia,

on Windows Server 2012 Hyper-V hyvä ratkaisu. Windows Server 2012 Hyper-V ja RemoteFX mahdollistavat virtuaalipalvelimeen asennetun näytönohjaimen eli graphics processing unit (GPU) käyttämisen eri virtuaalikoneiden kesken. Edellä kerrotusta voidaan hyvin huomata virtualisointitarpeiden erilaisuus ja joidenkin ohjelmistotuotteiden sopimattomuus virtualisoinnin kautta käytettäviksi. Eräät ohjelmistotuotteet eivät käytännössä ole virtualisoitavissa, vaan ne vaativat fyysisen laitteiston toimiakseen kunnolla. Yhtenä esteenä tähän voi olla lisensointi eli ohjelmiston lisenssi on kiinnitetty fyysiseen laitteeseen. Täten virtualisoitu laite voi olla hyvä tai huono ratkaisu, sillä lisenssi ei välttämättä toimi virtualisoidussa koneessa tai virtuaalikoneen siirtäminen palvelimelta toiselle aiheuttaa lisensoinnin nollautumisen. Testaus hyvissä ajoin on avaintekijä onnistuneessa virtualisointiprojektissa, koska suurin osa ohjelmistoista on kuitenkin virtualisoitavissa.

Vaikkakin suuri osa ohjelmistoista on virtualisoitavissa niin oheislaitteet eivät välttämättä ole. Monissa ympäristöissä vaatimuksena on oheislaitteiden kiinnittäminen fyysiseen koneeseen ja tämä ei yleensä ole kovinkaan helppoa virtualisoidussa ympäristössä. Tietoverkkoon liitettävät laitteet on mahdollista liittää virtuaalikoneisiin, mutta usein vaatimuksena on liittää tietokoneeseen viivakoodilukija, tulostin tai jokin liitäntäkortti esimerkiksi mittalaitteen liittämiseksi. Tässä tapauksessa virtualisointi ei ole järkevä eikä kuluja säästävä vaihtoehto. Monien lisälaitteiden ohjaimet eivät mahdollisesti tue virtualisoitua laitteistoa, joka aiheuttaa ongelmia. Näissä tilanteissa ei kenties ole järkevää tehdä virtualisointia vaan pitää laitteisto erillisenä fyysisenä palvelimena. Virtualisoinnin kehittyminen ja laitteiden verkottuminen voi tulevaisuudessa mahdollistaa enemmän virtualisointia, joten virtualisointi on usein pitkäkestoinen ja kehittyvä projekti. (Shesterin 2011.)

Virtualisointi tehostaa laitteiston hyötykäyttöä, mutta osin myös kuluttaa itse tehoja. Virtualisointialusta itsessään kuluttaa muistia ja vaatii prosessoritehoa. Tämä kulutus on kuitenkin pientä ja nykyisellä palvelinlaitteistolla sen osuus on häviävän pieni. Virtualisoidussa palvelimessa ei voida toimia kuten perinteisessä laitteistomallissa eli virtuaalimaailmassa on hieman erilaiset lainalaisuudet. Tämä voi aiheuttaa ongelmia resurssien laskemisen luotettavuudessa. Virtuaalipalvelimeen on kuitenkin helppo lisätä resursseja, joten palvelinvaatimusten aliarvioinnin aiheuttamat ongelmat on helppo korjata. Tähänkin ongelmaan löytyy ratkaisu kunnollisesta testauksesta ennen palvelun julkistamista käyttöön. (Pietroforte 2008.)

Laitteiston vaurioituminen voi aiheuttaa virtualisoidussa ympäristössä suuria ongelmia. Vikatilanteen sattuessa fyysisessä palvelimessa suoritettavat virtuaalikoneet vioittuvat kaikki kerralla. Redundant Array of Independent Disks (RAID)-ohjaimen eli tallennusmedian ohjaamiseen tarkoitetun laitteen vioittuessa saattavat kaikki fyysiset kiintolevyt lakata toimimasta, joka aiheuttaa virtuaalikoneiden suorittamisen keskeytyksen. Verkkokortin

vioittuessa kaikki tai ainakin osa verkkoyhteyksistä menetetään ja virtuaalikoneet menettävät verkkoyhteyden. Tämänkaltaiset vikatilanteet todennäköisesti aiheuttavat virtuaalikoneen palveluprosessissa sellaisen katkoksen, että virtuaalikoneen hallinta menetetään sillä seurauksella, ettei sitä enää kyetä siirtämään toimivaan palvelimeen. Riskienhallinta, ongelmien välttämiseksi, on kuitenkin mahdollista varautumalla edellä mainittujen tilanteiden kaltaisiin tapahtumiin ennalta. Tämänkaltaiset ongelmat voidaan mahdollisesti ennaltaehkäistä käyttämällä varmistettua palvelinlaitteistoa, keskitettyä ja kahdennettua tallennusjärjestelmää ja mielellään lyhyen vasteajan huoltosopimusta laitetoimittajan kanssa. Klusteroinnilla saadaan varmistettua isäntäkoneiden palvelutaso, kahdennetulla virtalähteellä voidaan varautua virtaongelmiin ja useammalla verkkoliitännällä voidaan varmistaa tietoverkon toimintaa. Aina on kuitenkin olemassa mahdollisuus vioittumiseen ja virtualisoidussa ympäristössä pienikin ongelma voi moninkertaistua ja aiheuttaa pitkänkin palvelukatkoksen. (Pietroforte 2008.)

Virtualisointi tarjoaa paljon mahdollisuuksia, mutta sisältää myös vaaroja. Hyödyt ja haitat tulisi ymmärtää mahdollisimman hyvin ja ottaa ne huomioon ennen virtualisointiprojektin toteuttamista, sillä virtualisointi ei ole ratkaisu kaikkeen. Ennalta varautuminen on osa riskienhallintaa myös virtuaalipalvelinympäristössä. Osana riskienhallintaa laitteiden varmistus ja keskeytyksettömän toiminnan takaaminen tulee ottaa samalla tavalla huomioon kuin fyysisten palvelinlaitteiden kanssa toimiessa. Varmistuksen ja vikatilannetoimintojen ohessa on myös muistettava päivitysten tarpeellisuus. Usein virtuaalimaailmassa luullaan turvallisuuden olevan paremmalla tasolla ja kenties osittain näin onkin. Päivittämätön virtuaalikone voi kuitenkin olla avoin hyökkäykselle ja toimia porttina muihin palveluihin. Tietoturvapoliitikkaa tulisi kehittää ja ottaa käyttöön kaikissa tietojärjestelmissä riippumatta palvelimen rakenteesta. Virtuaalijärjestelmiä tulisi kohdella kokonaisuudessaan hyvin samoin metodein kuin perinteisiä fyysisiä järjestelmiä. (Miller 2010.)

Virtuaalialustan päivittäminen on hyvä ottaa rutiinitoimenpiteeksi. Viestintäviraston julkaisemissa tiedotteissa on useita VMware-tuotteisiin liittyviä varoituksia tietoturva-riskeistä. Viimeisin tiedote on päivätty 26.8.2013 ja se koskee VMware Workstation ja VMware Player tuotteita, joita ajetaan Linux-käyttöjärjestelmässä. Tämä haavoittuvuus mahdollisti käyttöoikeuksien laajentamisen isäntäkoneessa, mutta se ei kuitenkaan mahdollista isäntäkoneen ja virtuaalikoneen välisiä oikeusmuutoksia. Aiemmin helmikuussa 2013 VMware julkaisi päivityksen, jolla paikattiin useita haavoittuvuuksia vCenter Server 4.1, vSphere Client 4.1 ja ESXi 4.1 tuotteisiin. Nämä haavoittuvuudet olivat huomattavasti kriittisempiä, koska kyseiset tuotteet ovat tarkoitettu suurempiin virtualisointikohteisiin ja ne ovat käytössä monissa isoissa tietokeskuksissa. Käytännössä virtualisoidussa järjestelmässä yhtenä lisähaavoittuvuutena on itse virtualisointialusta, joten päivitettävää on enemmän perinteisiin ratkaisuihin nähden. (Viestintävirasto 2013a; Viestintävirasto 2013b.)

7 oVirt

oVirt on ilmainen, avoimeen lähdekoodiin perustuva virtualisointihallintajärjestelmä, joka julkaistiin vuonna 2011. Projektin aloittajana toimi Red Hat, jonka tarkoituksena oli saattaa virtualisointijärjestelmä RHEV avoimen lähdekoodin piiriin ja kerätä mahdollisimman suuri kehittäjäjoukko mukaan. Mukaan lähtikin jo heti alussa alan suuria tekijöitä kuten IBM, Intel, Cisco, SuSE, ja NetApp. Ominaisuuksien puolesta oVirt on jo verrattavissa kaupallisiin tuotteisiin kuten Microsoft Hyper-V ja VMware vSphere. oVirt sisältää monia aiemmin vain kaupallisiin järjestelmiin sisältyneitä ominaisuuksia, kuten live migration eli virtuaalikoneen siirtäminen palvelimelta toiselle ilman palvelukatkoja, hallintamodulin ja muita toiminnallisuuksia, jotka mahdollistavat palveluiden keskeytyksettömän tarjonnan. oVirt muodostuu kahdesta pääosasta, jotka ovat oVirt Node ja oVirt Engine. (Captain KVM 2013.)

7.1 oVirt Node

oVirt Node on minimaalinen ja vakaaksi rakennettu Linux versio, johon on asennettu KVM hypervisor. Hypervisor toimii eräänlaisena tulkkina palvelinlaitteiston ja virtuaalikoneiden välillä mahdollistaen usean virtuaalikoneen keskustelemisen yhden laitteiston kanssa. oVirt Node käyttää minimaalisesti resursseja, sallien näin virtuaalikoneiden mahdollisimman tehokkaan palvelimella olevien resurssien hyödyntämisen. oVirt Node asennetaan palvelimelle ja sen tehtävä on hoitaa virtualisointiprosessi kyseisellä palvelimella. Virtuaalijärjestelmän laajentaminen on hyvin helppoa oVirt Noden avulla, sillä sen voi asentaa käytännössä mihin tahansa tietokoneeseen tai palvelimeen, kunhan prosessorista löytyy virtualisointituki. Intel-proessoreissa tuki on nimellä Intel VT ja AMD-proessoreissa AMD-V. oVirt Engine rajoittaa klusterin prosessoriarkkitehtuurin käyttämään ainoastaan toista prosessoriarkkitehtuurin prosessoria, joten kahden klusterin rakentaminen on välttämätöntä, mikäli käytetään molempien arkkitehtuurien laitteistoa. oVirt Node on valmiiksi rakennettu levykuva, jonka voi vapaasti ladata internetistä. Levykuva on tiedosto, joka pitää yleensä sisällään yhden asennusmedian kaikki tiedostot. Usein levykuva tallennetaan tallennusmedialle, jollaisia ovat CD- tai DVD-levyt tai USB-muistit. Levykuvaa voi käyttää myös itsessään asennusmediaa virtuaalikoneiden asennuksessa. oVirt Noden asennuksessa uusi palvelin käynnistetään tallennusmedialle tallennetulta levykuvalta ja oVirt Node asennetaan palvelimelle. Tämän jälkeen oVirt Node liitetään oVirt Enginen hallinnan alaisuuteen. Nämä toimenpiteet käsitellään tarkemmin opinnäytetyön projektiosuudessa. Olemassa oleva Fedora Linux voidaan myös liittää oVirt Enginen alaisuuteen isäntäkoneeksi asentamalla siihen oVirt ohjelmistopaketti. Tämä mahdollistaa nykyisten palvelimien vapaiden resurssien hyödyntämisen virtuaalijärjestelmän osana. (CPU World 2013.)

7.2 oVirt Engine

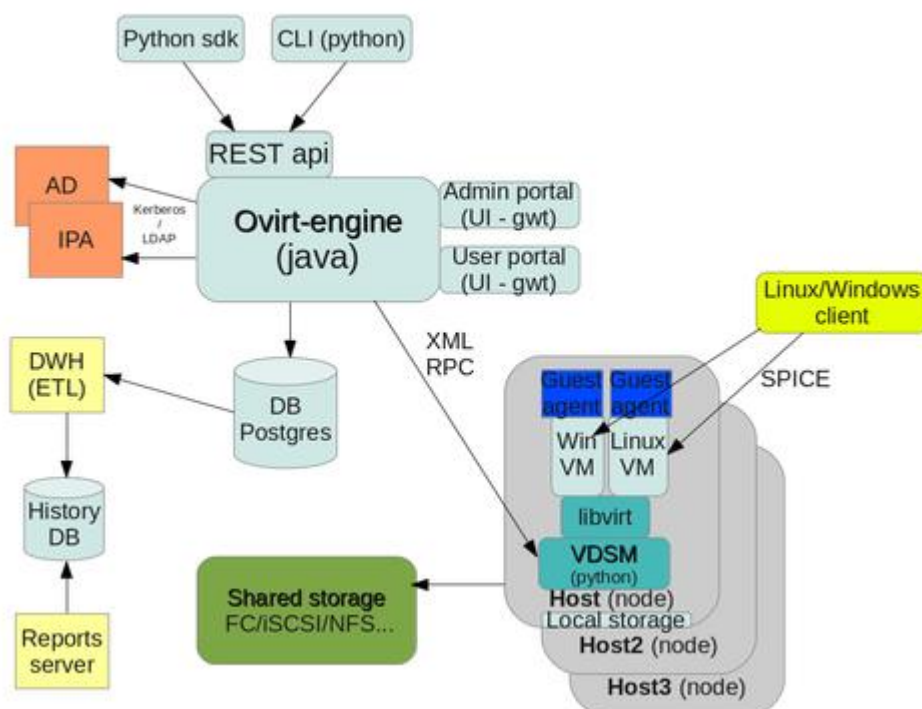
oVirt Engine on keskitetty graafinen hallintajärjestelmä, jonka tehtävä on hallita yhtä tai useampaa fyysistä palvelinta, joihin on asennettu oVirt Node tai Fedora Linux ja jotka ovat määritelty oVirt Enginen hallinnan alaisuuteen. Hallinta ulottuu virtuaalikoneisiin, tallennustilaan, virtualisoituun verkkoon ja lukuisiin muihin toimintoihin. oVirt Engine on Jboss-pohjainen Java-applikaatio, joka toimii internetselaimella web-applikaationa. oVirt Engine kommunikoi suoraan Virtual Desktop and Server Manager (VDSM) kanssa ja hoitaa virtuaalikoneiden hallinnan ja valvonnan. Kuvassa 5 on esitetty miten VDSM sijoittuu virtuaalikoneen ja isäntäkoneen välille ja miten oVirt Engine hallitsee järjestelmää käyttäen VDSM:n ominaisuuksia. oVirt Engine mahdollistaa hallinnan jakamisen käyttäjille ja ylläpitäjille, joille on omat hallintajärjestelmänsä ja erilaiset mahdollisuudet virtuaalijärjestelmän hallintaan. Ylläpitäjien hallintaportaalista voidaan tehdä käyttöoikeusmuutoksia käyttäjien hallintaportaaliin ja säädellä käyttäjien muokkausoikeuksien laajuutta. (oVirt.)

oVirt järjestelmä sisältää yleensä kolme peruskomponenttia:

- oVirt Engine -hallintajärjestelmä, jota käytetään virtuaalikoneiden luomiseen, monitorointiin, siirtämiseen toiselle palvelimelle, tallennustilan ja verkon muokkaukseen yms.
- Yksi tai useampi oVirt Node, jossa virtuaalikoneita suoritetaan.
- Yksi tai useampi tallennustila, joita käytetään keskitetysti tallennustilana virtuaalikoneille ja levykuville.

Näiden osien yksinkertaiseen asennukseen tai testikäyttöön on kehitetty oVirt All In One, joka sisältää oVirt Enginen ja oVirt Noden. Näin kaikki osat saadaan asennettua yhdelle palvelimelle ja järjestelmää päästään helpommin testaamaan. Tuotantokäytössä paras ratkaisu on asentaa oVirt Engine yhdelle palvelimelle, jonka ei tarvitse olla kovinkaan tehokas. Hallintajärjestelmän laitteistosuositus on neliydinprosessori ja 16GB keskusmuistia. oVirt Node on syytä asentaa mahdollisimman tehokkaalle palvelimelle, jotta usean virtuaalikoneen aiheuttama kuormitus pystytään hoitamaan. Hyvä perussääntö on asentaa useampi oVirt Node, jolloin yhden palvelimen vioittuminen ei vaurioita koko järjestelmää, koska virtuaalikoneet siirtyvät vikatilanteen sattuessa toiselle palvelimelle. Näin saadaan aikaan vikasietoisuutta ja palveluiden keskeytyksettömyys voidaan varmistaa. Tallennustila olisi hyvä olla vielä erillisenä yksikkönään ja sen kahdentaminen auttaa myös vikasietoisuuden parantamiseen. Käytännössä tallennustilan varmistaminen varmuuskopioilla voi olla hyvä ratkaisu, jos käytössä on RAID-varmistettu tallennustila. Tällä keinolla voi välttyä kalliilta investoinnilta kahdennettuun levyjärjestelmään. (oVirt.)

Järjestelmän pääkomponentit ja niiden liittymät on havainnollistettu kuvassa 4. Kaiken keskiössä on oVirt Engine ja oVirt Nodet, jotka on kuvassa 4 merkitty nimellä Host. Huomionarvoinen asia on myös data domain, joka tässä kuvassa on merkitty Shared storageksi. Se on keskitetty tallennustila, jossa virtuaalikoneiden levykuvat ja snapshotit eli tilannekuvat sijaitsevat. Yleensä Network File System (NFS) on helppo ja yksinkertainen ratkaisu tallennustilan muodoksi, mutta myös Fibre Channel Protocol (FCP) ja Internet Small Computer System Interface (iSCSI) ovat vaihtoehtoisina. NFS on suositeltava testiympäristössä ja kenties pienissä tuotantoympäristöissä, mutta FCP ja iSCSI mahdollistavat tehokkaamman keskitetyn ulkoisen tallennusjärjestelmän käyttämisen. (oVirt 2014.)



Kuva 4. oVirt arkkitehtuuri (oVirt.)

8 oVirt järjestelmän asennus

Omalta kohdaltani tämän työn tärkein ja mielenkiintoisin osuus oli oVirt-järjestelmän asentaminen ja vertaaminen muihin virtualisointijärjestelmiin. Testiin tuli valituksi oVirt All In One 3.1, jolla testiympäristön sai luotua kohtuullisen nopeasti ja kivuttomasti. Kuten Linux-käyttöjärjestelmien kanssa yleensä tapahtuu, tämäkään asennus ei sujunut ilman ongelmia ja pientä hienosäätöä komentoriviltä. Virtuaalijärjestelmän hallintaohjelmisto asennettiin Dell Poweredge T100 -tietokoneeseen, jossa ominaisuudet riittivät hyvin pientä testiympäristöä varten. Prosessorina Xeon X3220 @ 2.4Ghz (4 ydintä), keskusmuistia 4GB ja

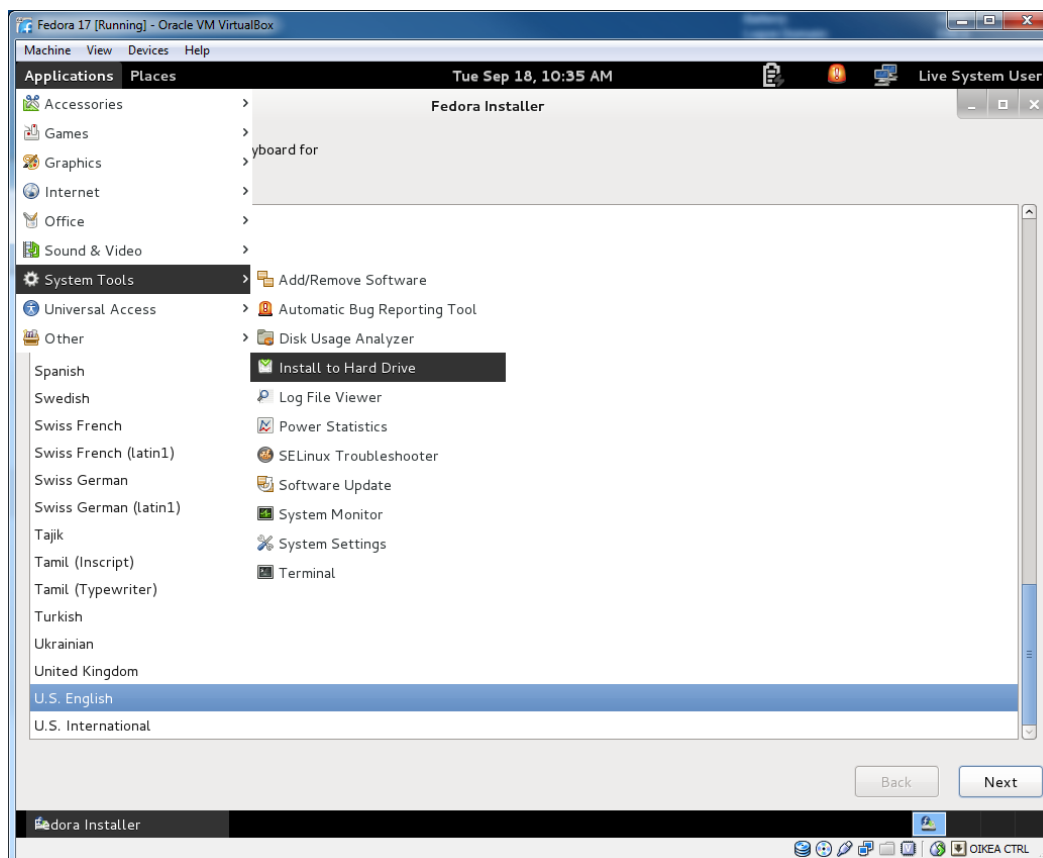
tallennustilana 146GB Serial Attached SCSI (SAS) kiintolevyt. Kiintolevyt on peilattuna RAID-1 tilassa, jolla saadaan vikasietoisuutta levyrikon varalle. Hallintakoneeseen asennettiin ensin Fedora Linux 17 -käyttöjärjestelmä, jonka päälle oVirt-hallintajärjestelmä asennettiin.

oVirt Node asennukseen löytyi kaksi käytöstä poistuvaa Dell Latitude E6410 -tietokonetta, joista rakennettiin kahden virtuaalipalvelimen järjestelmä, joita ohjataan hallintakoneen kautta. Näin saatiin aikaan hyvä testiympäristö, vaikka laitteisto ei aivan täytäkään palvelimille tyypillisiä tunnusmerkkejä.

8.1 Fedora Linux 17 -käyttöjärjestelmän asennus ja konfigurointi

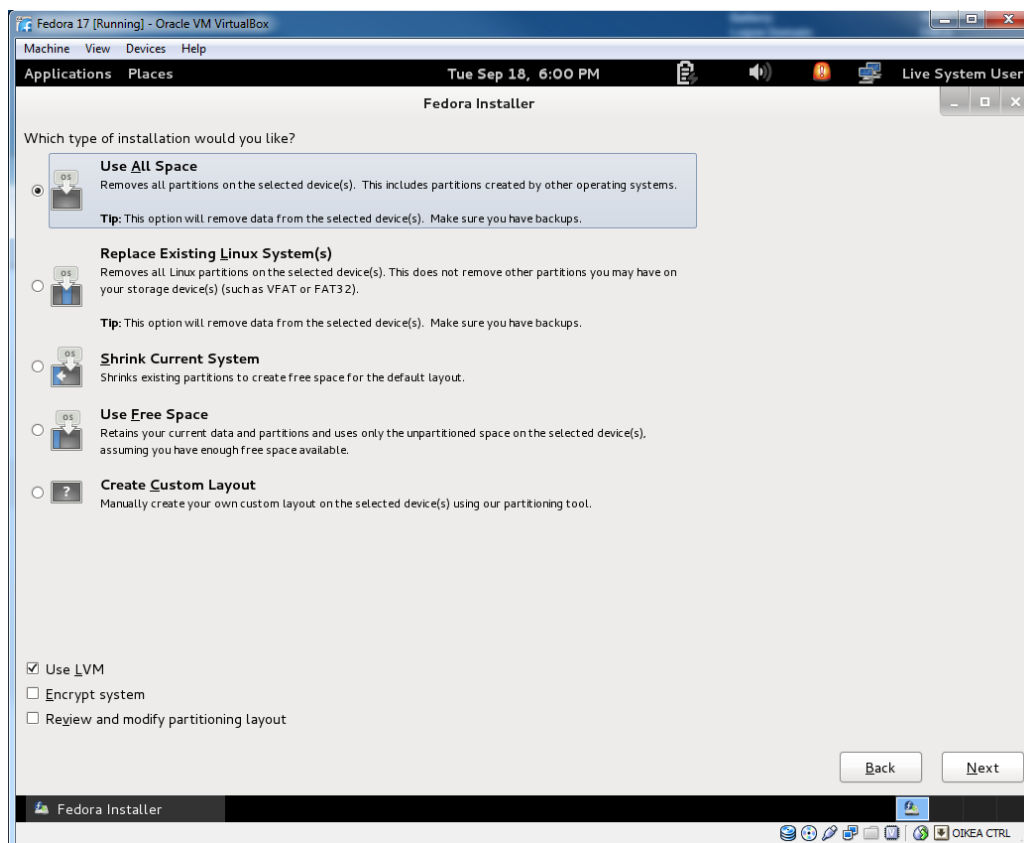
Linux-käyttöjärjestelmien asennuksessa on yleensä muutama vaihtoehto riippuen hieman versiosta ja käyttötarkoituksesta. Perinteisin asennustapa on komentorivipohjainen, mutta nykyään on jo yleistynyt graafinen asennustapa, joka on huomattavasti käyttäjäystävällisempi. Testikoneeseen asennettiin Fedora Linux 17 web install -versio, joka on komentorivipohjainen ja sisältää vain web-palvelimelle tyypilliset komponentit, mutta tässä esitetään graafinen asennustapa sen havainnollisuuden vuoksi.

Kuvassa 5 tietokone käynnistetään Fedora Live -asennusmedialta, jolloin ominaisuuksia pääsee testaamaan ilman asentamista. Helpoin tie asennukseen kulkee tätä kautta ja asennusta varten valikosta valitaan Install to Hard Drive, joka käynnistää asennusohjelman. Ensimmäisenä tulee eteen näppäimistön valinta ja valintana yleensä löytyy Finnish tai Swedish, joka kannattaa valita käytettäessä skandinaavista näppäimistöä. Seuraava vaihe asennuksessa on tallennusjärjestelmän valinta, joka yleensä tarkoittaa kohdelevyn valintaa. Valittavana on suuri määrä erilaisia hienosäätöjä, mutta usein pelkkä levyaseman valinta oletusasetuksilla on riittävä. Tämän jälkeen annetaan koneelle nimi, valitaan aikavyöhyke ja syötetään pääkäyttäjän (root) salasana.



Kuva 5. Asennuksen aloittaminen

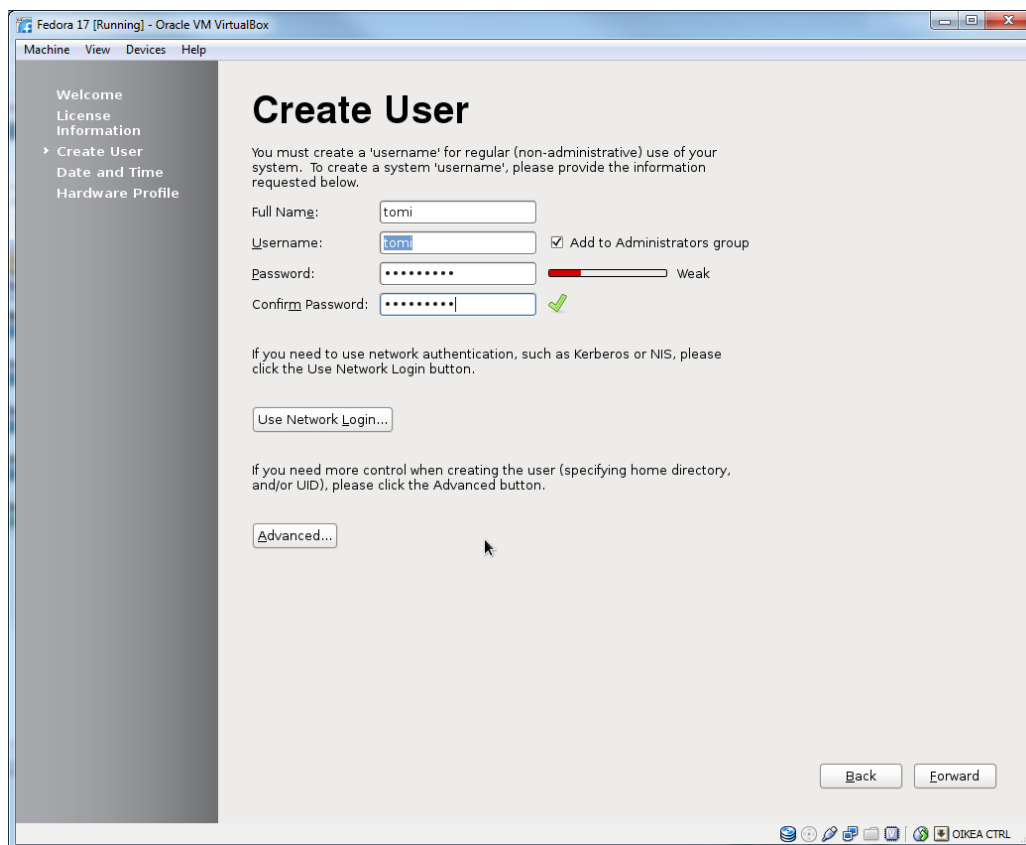
Seuraavaksi asennusohjelmalle pitää kertoa, miten levyasemaa käsitellään asennusvaiheessa. Kuvassa 6 esitetään vaihtoehtot: käytä kaikki tila tai tee levytilan muokkauksia, jos samalle koneelle halutaan asentaa useita eri käyttöjärjestelmiä. Tässä tapauksessa levyille asennettiin vain yksi käyttöjärjestelmä, joten käytettäväksi valittiin koko levy. Seuraavaksi hyväksyttiin tehdyt asetusmuutokset ja asennus pääsi alkamaan. Asennuksen päättyttyä tulee ilmoitus onnistuneesta asennuksesta ja kehoitus käynnistää kone uudestaan. Tässä vaiheessa tulee poistaa asennusmedia tietokoneesta, jotta asennusohjelma ei käynnisty uudelleen.



Kuva 6. Levyaseman käsittely asennusvaiheessa

Koneen käynnistyttyä uudelleen tulee vielä syöttää muutamia tietoja, joita asennus tarvitsee toimiakseen kunnolla. Loppuasennusta varten luetaan lisenssiehdot ja hyväksytään ne. Kuvassa 7 luodaan käyttäjätili ja yleensä lisätään tämä käyttäjä pääkäyttäjäksi (Add to Administrators group). Käyttäjätilin luomisen jälkeen tulee vielä määrittää päivämäärän ja kellon synkronointi. Oletuksena tarjolla on Fedoran Network Time Protocol (NTP) -palvelimet ja nämä yleensä riittävät, jos tarjolla ei ole omassa verkossa toimivaa NTP-palvelinta. Näiden asetusten jälkeen käyttöjärjestelmä on valmis räätälöintiä varten.

Räätälöinnillä tarkoitetaan tässä tapauksessa ohjelmistopakettien asennusta ja asetusten muuttamista käyttöjärjestelmän saattamiseksi siihen tilaan, että oVirt-hallintajärjestelmä voidaan asentaa ja että se toimisi ongelmitta. Näitä räätälöintiohjeita ei välttämättä voi sellaisenaan käyttää, mutta näistä saa hyvän kuvan mahdollisista ongelmakohtista ja tehdyistä toimenpiteistä.



Kuva 7. Käyttäjätilin luominen

Testiympäristössä räätälöinniksi riitti NetworkManager -ohjelman pysäyttämisen ja verkkoyhteyden palauttaminen komennoilla:

```
sudo systemctl stop NetworkManager.service
sudo systemctl disable NetworkManager.service
sudo systemctl restart network.service
```

Näiden toimenpiteiden lisäksi muutettiin koneen Internet Protocol (IP)-osoite kiinteäksi ja määriteltiin muut verkkoyhteyden parametrit komennolla:

```
sudo vi /etc/sysconfig/network-scripts/ifcfg-em1
```

Yksi olennainen asetusmuutos on vielä tehtävä, jotta palvelimen palomuuuri sallii yhteydet palvelimelle. Tässä tapauksessa järjestelmä on vain sisäverkossa toimiva ja liikenne ulkomaailmaan on rajoitettu erillisillä palomuuureilla. Tästä johtuen oli järkevintä vain sulkea palomuuuri ja estää sen käynnistyminen automaattisesti uudelleenkäynnistyksen yhteydessä komennoilla:

```
sudo service iptables stop
chkconfig iptables off
```

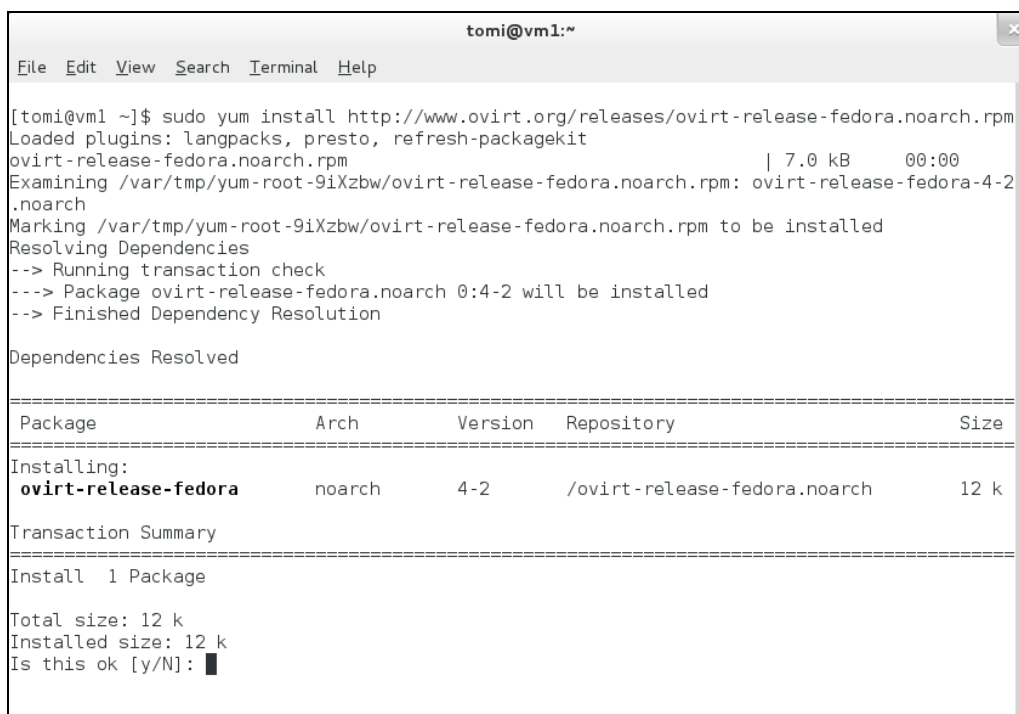
8.2 oVirt All In Onen asennus ja konfigurointi

oVirt-hallintajärjestelmän asennus aloitettiin asentamalla koneeseen ensin GNU Compiler Collection (GCC)-ohjelmapaketti, oVirt RPM Package Manager (RPM)-paketti ja oVirt Engine. Kuva 8 havainnollistaa asennuksen tapahtuvan komentorivikehötteen kautta. Asennus tehtiin komennoilla:

```
sudo yum install gcc
```

```
sudo yum install http://www.ovirt.org/releases/ovirt-release-fedora.noarch.rpm
```

```
sudo yum install ovirt-engine-setup-plugin-allinone
```



```
tomi@vml:~$ sudo yum install http://www.ovirt.org/releases/ovirt-release-fedora.noarch.rpm
Loaded plugins: langpacks, presto, refresh-packagekit
ovirt-release-fedora.noarch.rpm | 7.0 kB 00:00
Examining /var/tmp/yum-root-9iXzbw/ovirt-release-fedora.noarch.rpm: ovirt-release-fedora-4-2.noarch
Marking /var/tmp/yum-root-9iXzbw/ovirt-release-fedora.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package ovirt-release-fedora.noarch 0:4-2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
ovirt-release-fedora noarch 4-2 /ovirt-release-fedora.noarch 12 k
Transaction Summary
-----
Install 1 Package

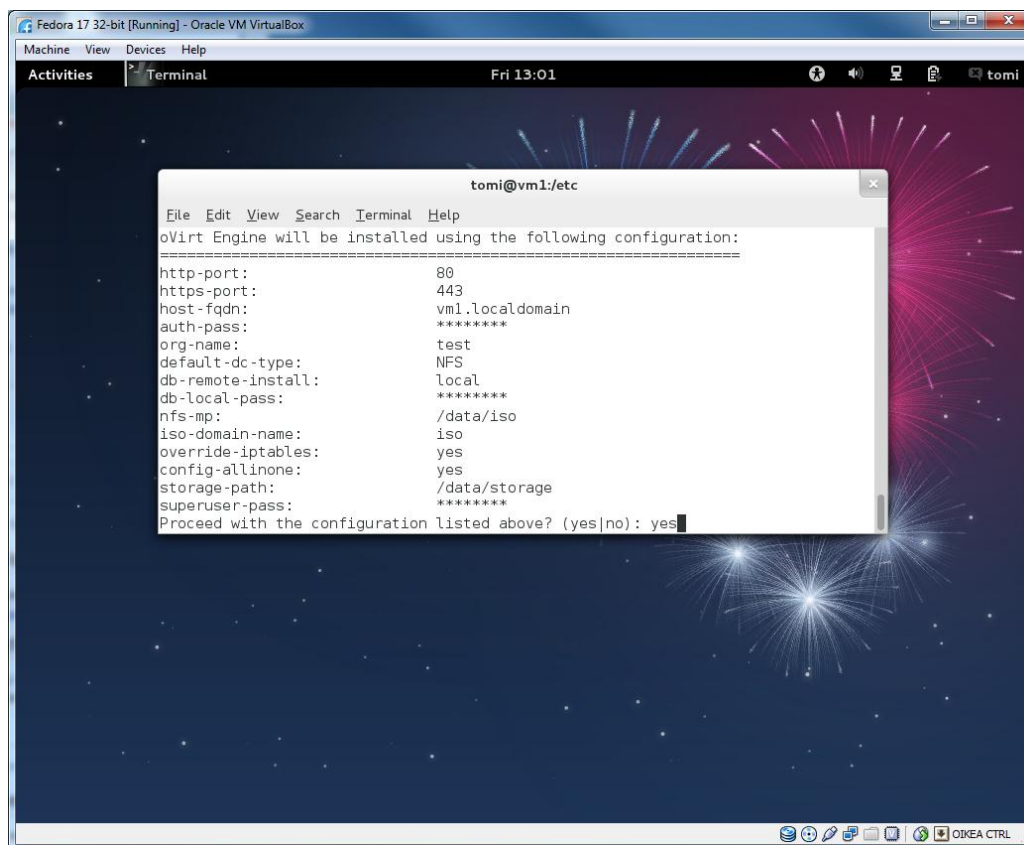
Total size: 12 k
Installed size: 12 k
Is this ok [y/N]: █
```

Kuva 8. oVirt-paketin asennus

Asennuksen seuraava vaihe on viimeistellä asennetut ohjelmistopakettit. Loppuasennus ja asetuserittelyt tehtiin komennolla:

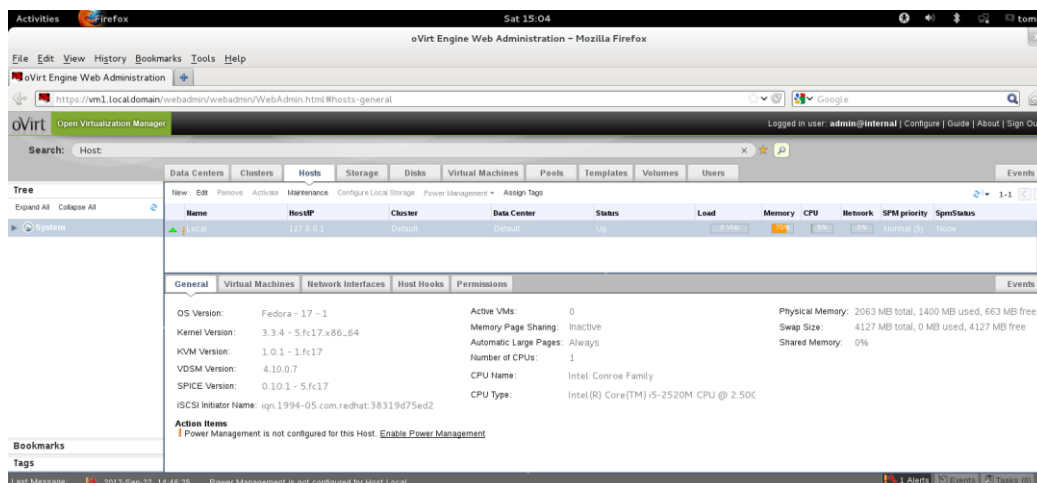
```
sudo engine-setup
```

Asetuserittelyissä, kuvassa 9, määritetään käytettävät HyperText Transfer Protocol (HTTP) ja HyperText Transfer Protocol Secure (HTTPS) portit. Näiden lisäksi määriteltiin Fully-Qualified Domain Name (FQDN), tietovaraston muoto ja polku, perustettavan tietokannan salasana ja muita määrittelyä tarvitsevia arvoja. (Rescorla 2000.)



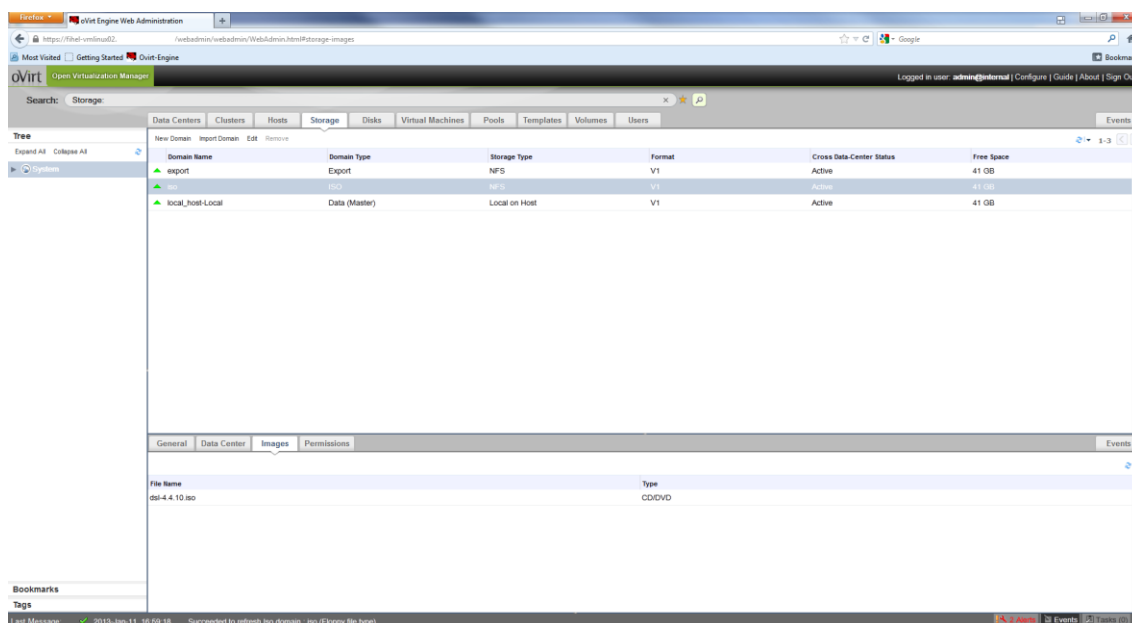
Kuva 9. oVirt Enginen asetusmäärittely

oVirt Engine Setup antoi virheen Local hostin lisäämisen epäonnistumisesta, mutta tämä on dokumentoitu virhe ja johtuu asennusohjelman virheellisestä odotusajan määrittelystä. Ongelma on luultavasti ratkaistu jo uusimmissa versioissa, koska se aiheutti paljon virheraportointia kehittäjäryhmälle. Testijärjestelmässäkin tämä aiheutti pienen ongelman, mutta se oli helposti ratkaistavissa hallintasivun kautta. Kuvan 10 mukaisesti Host piti laittaa huoltotilaan (Maintenance mode), valita Configure Local Storage, valita poluksi (path) sama tietovaraston polku (storage-path) kuin asennusvaiheessa ja hyväksyä muutos. Tämän jälkeen järjestelmä muodostaa yhteyden määriteltyyn polkuun ja Local hostin voi ottaa pois huoltotilasta. Local hostin pitäisi tämän jälkeen olla tunnistanut tietovaraston ja olla käynnissä normaalisti.



Kuva 10. Local host saatu käyntiin Local storagen lisäämisen jälkeen.

Kuvassa 11 Storage-välilehdellä on näkyvissä kaikki tietovarastot, jotka on määritelty tälle palvelimelle. Asennusvaiheessa määritelty /data/iso -tietovarasto näkyy todennäköisesti statuksella Unattached, koska sitä ei ole liitetty vielä Data Centeriin. Liittäminen tapahtuu valitsemalla tietovarasto, alaikunasta valinta Data Center ja sieltä Attach. Valitaan Data Center, johon tietovarasto halutaan liittää ja hyväksytään valinta. Tämän jälkeen pitää vielä valita alaikunasta valinta Activate, jonka jälkeen tietovarasto liittyy valitun Data Centerin tietovarastoksi ja aktivoituu.



Kuva 11. Tietovarastot ja iso-tietovarastoon lisätty levykuva.

Iso-tietovarasto on varattu ISO-levykuville, joista virtuaalikoneita asennetaan. Tällaisia ovat esimerkiksi Windows- ja Linux-versioiden asennusmedian levykuvat. Local-tietovarasto taas on

ns. Master-tietovarasto, jossa virtuaalikoneiden käyttölevyt sijaitsevat. Export-tietovarasto on tarkoitettu virtuaalikoneiden siirtämiseen.

oVirt All In One loi oletuksena local_datacenter Data Centerin, joka on käytännöllinen vain yhden palvelimen ratkaisussa. Uuden Data Centerin ja Clusterin luominen on pakollista, jos halutaan liittää useampi palvelin hallintajärjestelmään. Tässä tapauksessa myös tietovarastot on liitettävä uuden Data Centerin alaisuuteen ja alkuperäinen local_datacenter jää tyhjäksi. Tästä on hyvin huomattavissa helppokäyttöisyyden ja yhdistelyn tuomat hyödyt ja haitat. Testaus helpottuu, mutta testiympäristön vieminen tuotantoon ei käytännössä ole järkevää.

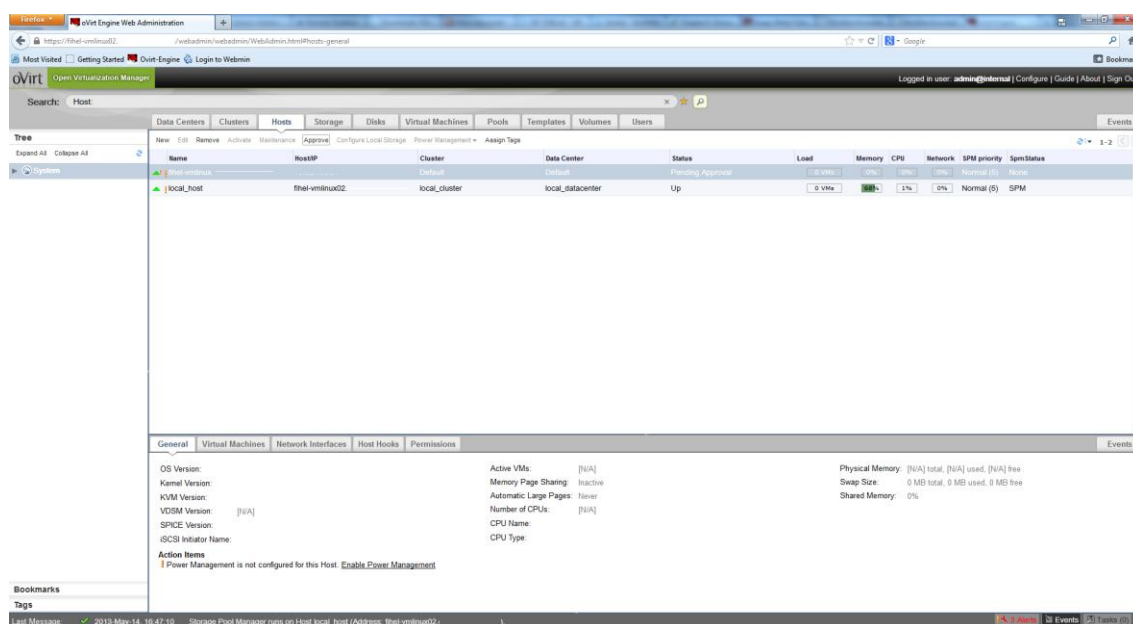
8.3 oVirt Noden asennus, konfigurointi ja liittäminen hallintajärjestelmään

oVirt Node Hypervisor on erillisenä projektina oVirt-projektista. Vaikka kehittäjäryhmä onkin erillinen niin sitä kehitetään mahdollistamaan helppo oVirt-järjestelmän laajennettavuus. Testattu versio oli oVirt Node Hypervisor 2.5.1 (1.0.fc17). oVirt Node -levykuva ladattiin verkosta ja tallennettiin CD-levylle. Tämän jälkeen palvelimen roolia esittävä Dell Latitude E6410 käynnistettiin CD-levyltä ja päästiin asennuksen alkuun valitsemalla Start Ovirt Node.

Seuraavassa vaiheessa valitaan Hypervisor asennettavaksi tai vaihtoehtoisesti vanhemman version päivitys, mikäli palvelimella on jo aikaisempi versio olemassa. Tämän jälkeen vuorossa on näppäimistön kieliasetusten valinta sekä käynnistyslevyn ja asennuslevyn valinta. Nämä olivat testitapauksessa se yksi ja ainoa levy, joka koneesta löytyi. Seuraavaksi asetettiin Hypervisorille salasana ja aloitettiin asennus. Järjestelmä muokkasi tietokoneen kiintolevyn tarvitsemaansa muotoon ja teki käynnistyssektorin muutokset. Asennuksen valmistuttua asennusohjelma pyysi käynnistämään tietokoneen uudestaan.

Uudelleenkäynnistyksen jälkeen koneelle pääsee kirjautumaan ja sen jälkeen on tarpeen tehdä muutamat asetusmuutokset. Verkkokortille kannattaa asettaa kiinteä IP-osoite, jotta dokumentointi helpottuu ja IP-osoite ei vaihdu satunnaiseksi, esimerkiksi virtakatkoksen yhteydessä. Etäyhteys kannattaa laittaa päälle, jotta koneeseen päästään verkon kautta yhteyteen käyttämällä secure shell (SSH)-yhteyttä. oVirt Engine valinnasta pääsee liittämään Hypervisorin oVirt hallintajärjestelmän alaisuuteen ja osaksi luotua Data Centeriä. Kirjoita annettuun kenttään hallintapalvelimen osoite kohtaan Management Server ja valitse oletusportti tai vaihda se, mikäli se on oVirt Enginen asennusvaiheessa vaihdettu. Tämän jälkeen hyväksy asetusmuutokset ja hyväksy annettu sertifikaatti. Sertifikaatin avulla voidaan varmistaa oikeaksi todettujen laitteiden lisääminen hallintajärjestelmään ja jatkossa näiden välinen tietoliikenne luottaa luotujen sertifikaattien aitouteen.

Näiden asetusmuutosten jälkeen oVirt Node Hypervisor on hyväksyttävissä hallinta-järjestelmän Hosts-listassa. Kuvassa 12 esitetään näkymä, josta oVirt Node saadaan hyväksytyä osaksi virtualisointijärjestelmää. Valitse hyväksyttävä oVirt Node Hypervisor, valitse Approve ja haluttu Data Center. Näin hyväksyntä menee läpi, status eli tilatieto muuttuu tilasta Pending Approval statukseen Up ja laitteesta tulee uusi virtuaalipalvelin Data Centeriin.



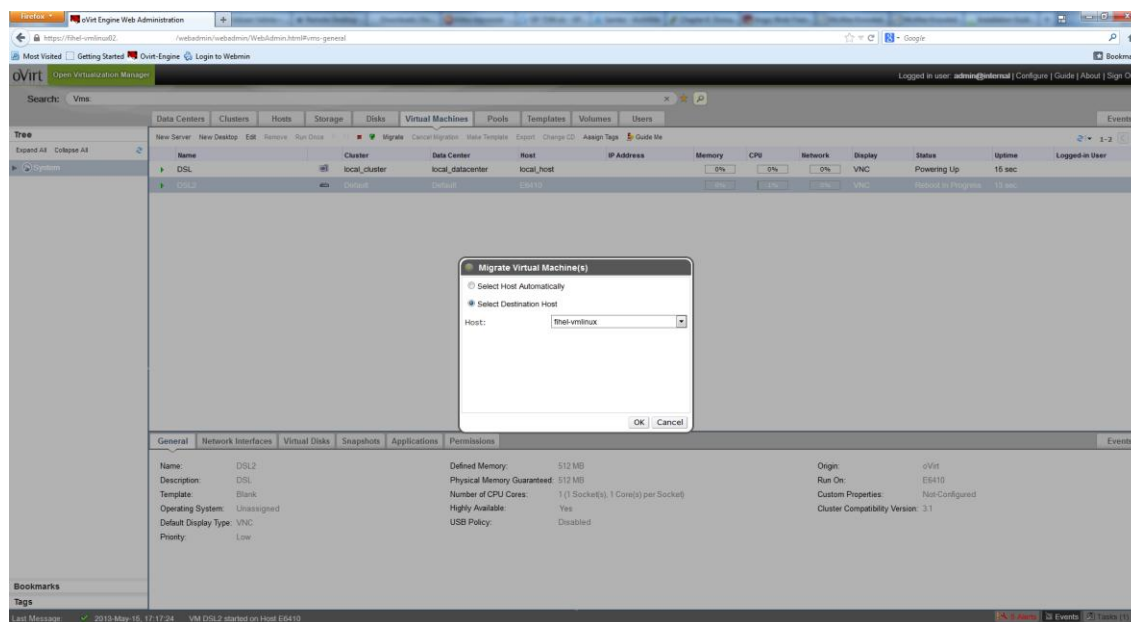
Kuva 12. Uusi hypervisor lisätty ja hyväksyttävissä

8.4 Valmiin virtualisointijärjestelmän ominaisuuksien testaus

Ominaisuuksien testaaminen painottuu lähinnä hallittavuuteen ja laajennettavuuteen, koska ne ovat juurikin olleet kompastuskiviä avoimen lähdekoodin tuotteissa ja vahvuuksia kaupallisissa tuotteissa. Ilmaisia virtuaalijärjestelmiä on ollut jo pitkän aikaa, mutta vasta nyt niihin on saatu mukaan myös hallinta. Tämä avaa uusia ovia ja antaa uusia mahdollisuuksia IT-ammattilaisille valittaessa uutta virtuaalijärjestelmää tai laajennettaessa jo olemassa olevaa järjestelmää.

Testausta varten testijärjestelmään luotiin yksi virtuaalikone, jonka käyttöjärjestelmänä oli DSL Linux. Isäntäkoneita (host) liitettiin testausta varten perustettuun uuteen klusteriin kaksi, joiden avulla pystyttiin testaamaan virtuaalikoneen siirtoa isäntäkoneelta toiselle. Testit suoritettiin siirtämällä virtuaalikone manuaalisesti toiselta isäntäkoneelta toiselle isäntäkoneelle ja automaattisen siirron toimivuutta testattiin katkaisemalla käynnissä olevan virtuaalikoneen isäntäkoneesta virta.

Ensimmäisessä testissä virtuaalikone siirrettiin manuaalisesti toiselta isäntäkoneelta toiselle isäntäkoneelle. Tämä suoritettiin valitsemalla virtuaalikone ja hallintapaneelista toiminto Migrate. Tämän jälkeen valittiin uusi isäntäkone ja hyväksyttiin siirto kuvan 13 mukaisesti valinnalla OK. Virtuaalikoneen siirto käynnistyi välittömästi ja muutaman minuutin kuluttua virtuaalikone oli käynnissä uudella isäntäkoneella. Siirto tapahtui reaaliaikaisesti ja virtuaalikone pysyi käynnissä koko siirtämisen ajan. Tästä metodista käytetään yleisesti nimitystä live migration, mutta VMware käyttää tästä omaa vMotion-termiä.



Kuva 13. Virtuaalikoneen siirto manuaalisesti toiselle isäntäkoneelle

Automaattisen isäntäkoneen vaihdon testaus oli mielenkiintoisempi, koska siinä saattoi odottaa tuloksien vaihtelevan suuresti. Käytännön testaus osoitti kuitenkin sen, että hallintajärjestelmä huomasi kohtuullisen nopeasti isäntäkoneen kadonneen hallinnasta ja hallintajärjestelmä käynnisti virtuaalikoneen toiselle isäntäkoneelle automaattisesti. Tässä tapauksessa tulos oli ennakoidun kaltainen eli virtuaalikone käynnistyi uudelleen toisella isäntäkoneella ja käyttökatkos oli muutaman minuutin mittainen. Tämä on ominaisuutena myös kaupallisissa tuotteissa, joten testi ei tarjonnut mitään poikkeuksellista yllätystä.

Ongelmattomia järjestelmät ei vielä ole ja tämän ongelmakohdan testit osoittivatkin muutamaan kertaan. On kuitenkin otettava huomioon, että järjestelmään tehtiin jatkuvasti muutoksia, sitä käynnisteltiin ja sammuteltiin välillä oikein ja välillä väkisin, jotta saadaan mahdolliset ongelmat esiin. Tuotantokäytössä järjestelmät ovat enemmän tai vähemmän rauhassa asetusmuutoksilta ja niiden aiheuttamilta toimenpiteiltä, joten järjestelmään ei kohdistu samanlaista rasitusta. Noin puolen vuoden jälkeenkin testijärjestelmä oli täydessä kunnossa eikä mitään huoltotoimenpiteitä ole tarvittu. Testauksen perusteella tuotantokäytössä ei tulisi olemaan suuria ongelmia odotettavissa.

Testijärjestelmä meni testin aikana useamman kerran vikatilaan virhekoodeilla: host non-responsive tai guest unknown state. Nämä ongelmat saatiin kuitenkin korjattua helposti komentorivin komennolla:

```
sudo systemctl restart vdsmd.service
```

Satunnaisesti ilmenevä ongelma käynnistysten yhteydessä oli VDSM-palvelun jääminen käynnistymättä. Näissä tapauksissa ongelmasta päästiin eroon käynnistämällä VDSM-palvelu komennolla:

```
sudo service vdsmd start
```

8.5 Eri virtualisointijärjestelmät vertailussa

Opinnäytetyön vertailuosuudessa on mukana oVirt-virtualisointijärjestelmän lisäksi Microsoft HyperV ja VMware vSphere. Näitä kolmea tuotetta on koko opinnäytetyön ajan vertailtu keskenään. Tutkimusosuudessa esitettiin kysymyksiin ”Onko avoimen lähdekoodin virtualisointi-järjestelmästä kodin tai yrityksen virtualisointiratkaisuksi?” tai ”Voiko kaupallisen virtualisointijärjestelmän korvata avoimen lähdekoodin järjestelmällä?” voisi helposti vastata kyllä. Käytännössä kuitenkin suurten yritysten voi olla turvallisempaa käyttää kaupallisia ratkaisuja, sillä niihin on helpommin saatavilla tukea konsulteilta ja ohjelmistojen valmistajilta. Päivitettyvyys on helpompaa kaupallisissa tuotteissa, mutta vain tietyin rajoituksin. Virtualisointijärjestelmän soveltuvuus tilanteeseen ja yritykseen on harkittava aina tapauskohtaisesti. Ilmaisen järjestelmän käyttöönotto on usein se hankalin ja se vaatii tietynlaista osaamista. Osa ominaisuuksista vaatii paljonkin työtä ja saadaanko niistä siltikään käytäntöön soveltuvia? Mikäli palvelinjärjestelmiä ja osaamista aletaan rakentamaan alkutilanteesta, on avoimen lähdekoodin järjestelmä mahdollisesti houkutteleva vaihtoehto. Yksi harvoin ajateltu vaihtoehto voi olla kahden järjestelmän käyttöönotto. Ensi- ja toissijaisten tai muuten vaan eri prioriteetin järjestelmät voivat vaatia erilaisia ratkaisuja, joista toisen voi hoitaa kaupallinen tuote ja toisen avoimen lähdekoodin tuote. Tärkein kuitenkin on miettiä vaihtoehtoja, tutustua niihin eikä vain pitäytyä yhdessä vanhassa tutussa järjestelmässä pelkästään sen takia, että sitä on ennenkin käytetty.

VMware-virtualisointijärjestelmät ovat myös rakennettu Linuxin kaltaiselle pohjalle, mutta niiden käytettävyyteen on panostettu huomattavasti oVirt-järjestelmää enemmän. Microsoftin Hyper-V on täysin oma järjestelmänsä, joka jostain syystä ei ole saanut suurempaa jalansijaa markkinoilta. Kaikissa tutkituissa järjestelmissä on lähes samat ominaisuudet, toiminnallisuudet ja mahdollisuudet. Pienemmässä yrityksessä ilmainen oVirt-järjestelmä on varmasti houkutteleva vaihtoehto. Varsinkin verrattuna kaupallisten tuotteiden kohtuullisen suureenkin hintalappuun ja vuosittaisiin lisenssimaksuihin. Kaikilla testatuilla tuotteilla laajennettavuus on samantasoista ja integroiminen eli liittäminen osaksi pilvipalveluita on

myös mahdollista. Suurin eroavaisuus on hallintaan käytetyissä työkaluissa. VMware ja Microsoft luottavat erillisiin hallintasovelluksiin, mutta oVirt-järjestelmässä hallinta tapahtuu internetselaimen avulla tai komentoriviltä. VMware on kehittänyt hallintasovellusta internetselaimella käytettäväksi ja lähitulevaisuudessa selainpohjainen hallinta on ensisijainen myös VMwaren tuotteissa. Molemmissa tavoissa on omat hyvät ja huonot puolensa. Erillisessä sovelluksessa on yleensä helpompi logiikka ja ne pysyvät hyvin muuttumattomina versioiden välillä. Sovellukset on kuitenkin asennettava erikseen tietokoneille, joilta virtuaalijärjestelmää halutaan hallita. Internetselaimella hallinta onnistuu kaikilta koneilta, jotka ovat samassa verkossa ja joissa on jokin järjestelmän tukema selain.

Tulevaisuuden arviointi onkin vaikeampi tehtävä. Kaikki järjestelmät kehittyvät ja kopioivat toimintoja toisistaan. Yleensä myös käytettävyys kehittyy ajan kuluessa. Avoimen lähdekoodin sovelluksia usein vaivaa projektin vetäjien poistuminen tai projektin hajaantuminen erillisiksi projekteikseen. Tässä tapauksessa oVirt-projekti kuitenkin kulkee kaupallisen valmistajan vanavedessä, joten samanlaisia riskejä ei ehkä ole tämän tuotteen kanssa. Kaupallisilla valmistajilla on kova kilpailu markkinaosuuksista ja tuotteet kehittyvät vauhdilla, mikäli markkinoilla on kiinnostusta heidän tuotteilleen. Tämä edistää kehitystä kaupallisten ja myös avoimen lähdekoodin tuotteiden keskuudessa. Erilaiset ohjelmisto-, palvelin- ja työasema-virtualisointijärjestelmät ovat kuitenkin tulossa kovasti. Mielestäni suurimpana esteenä yleistymiselle on tällä hetkellä kuitenkin epäselvyydet lisenssiasioissa ja kalliit lisenssit. Moni ohjelmistovalmistaja on kuitenkin muuttanut lisenssipolitiikkaa suuntaan, joka tukee virtualisoinnin kehittymistä.

9 Yhteenveto

Toimintatutkimuksellinen asiantuntija-arvioinnin suorittaminen vaati yllättävänkin paljon aikaa ja työtä. Toiminnallinen osuus oli huomattavasti aikaa kuluttavampaa kuin arvioinnin tekeminen, sillä sitä pystyi tekemään samanaikaisesti toiminnallisen osuuden kanssa. Käytetty oVirt-virtualisointijärjestelmä on kohtuullisen tuore ja jatkuvasti kehittyvä, joten dokumentaation löytäminen tuotti satunnaisesti yllättävänkin paljon haasteita. Tästä syystä opinnäytetyössä käytettiin suurimmaksi osin sähköisiä lähteitä. Myös pohjalla olevan Fedora Linux 17 -käyttöjärjestelmän virheet ja oikut tulivat satunnaisesti ongelmiksi, mutta näihin odotinkin törmääväni. Opinnäytetyön kirjoittamisen aikana on tullut jo useampiakin versioita työssä esitetystä käyttöjärjestelmästä ja virtualisointijärjestelmästä, joten kaikki ongelmat eivät välttämättä toistuisi tutkielman mallin mukaisesti. Uusia ongelmia on kuitenkin odotettavissa, mutta IT-alan nopean muutosrytmin mukaisesti olisi yllättävämpää, jos ongelmia ei projektin aikana ilmenisi.

Yksin tehdyn toimintatutkimuksen validiteetin osalta törmäsin ongelmaan, jota en ennalta osannut arvioida. Validin tutkimuksen edellytyksenä on, että tutkimustulos olisi yhdenmukainen muiden alan asiantuntijoiden mielipiteiden kanssa. Tässä opinnäytetyössä käsiteltiin havaintoja ja kokemuksia vain yhden asiantuntijan näkökulmasta, eikä tuloksia käyty läpi muiden asiantuntijoiden kanssa. Olen kuitenkin ollut IT-alan kanssa tekemisissä jo lähes 20 vuotta, joten tietynlainen asiantuntijuus ja yleisen käsityksen omaksuminen alalla on kehittynyt vahvuudekseni. Täten tutkimustulos on heijastettavissa yleiseen käsitykseen ja tutkimustulosta voidaan pitää validina. Tutkimustulos oli kuitenkin hieman kaksijakoinen, joten näin ollen tutkimustulosta voidaan pitää myös epävalidina, mutta opinnäytetyön alussa esitettyyn tutkimusongelmaan saatua tulosta voidaan pitää validina. Validiteetin arviointi jää myös jokaisen lukijan arvioitavaksi, sillä he voivat reflektoida tutkimuksen tuloksia omiin kokemuksiinsa. (Anttila 1998).

Reliabiliteetin osalta tutkimus on kaksijakoinen. Täysin samalla laitteistolla ja samoilla ohjelmistoversioilla toistettuna voidaan tutkimustulosta pitää reliabelina ja tutkimustulos on toistettavissa. Pienenkin muutoksen takia tutkimustulos saattaisi muuttua hyvinkin paljon. Eri laitteistot ja ohjelmistoversiot toimivat hyvinkin eri tavoin, joten osa ongelmista saattaisi jäädä pois kokonaan tai korostua merkittävästi. Tässä opinnäytetyössä ei esitelty ohjelmistoversioiden tarkkoja versionumeroita, joten siltä osin toistettavuus voi kärsiä, mutta toisaalta se on hyvin yleinen tilanne tietotekniikan alueella.

Opinnäytetyön rajaus muuttui hieman alkuperäisestä suunnitelmasta ja kohdentui enemmän oVirt-järjestelmän ominaisuuksien tutkimiseen. Suurin työmäärä oli odotetusti tutkitussa virtualisointijärjestelmässä. Vertailukohtina toimivat järjestelmät ja niiden ominaisuuksien tutkimus jäi osittain syrjään ja niihin viitataan lähinnä vertailukohtina oVirt-järjestelmälle. Vertailussa olevia virtualisointijärjestelmiä käytettiin ja tutkittiin kuitenkin sen verran, että vertailua pystyttiin suorittamaan. Toinen vaativa tehtävä opinnäytetyön kehitymisessä oli tutkimustulosten dokumentointi ja auki kirjoittaminen. Opinnäytetyön aikana ilmenneiden ajankäytön haasteiden ja työkiireiden vuoksi työ jäi osittain hieman pinnalliseksi kaupallisten järjestelmien osalta. Näistä syistä sainkin työnantajalta suostumuksen osallistua koulutuksiin, jotta syvällisempi ymmärrys erilaisten virtualisointijärjestelmien eroista ja mahdollisuuksista korostuisi henkilökohtaisessa osaamisessani.

Opinnäytetyön lähtökohtana oli työelämälähtöisyys. Työssäni pääsen jatkuvasti tutkimaan uusia toteutusmahdollisuuksia ja arvioimaan järjestelmien ja ohjelmistojen soveltuvuutta käytäntöön, joten tästä opinnäytetyöstä oli suuri hyöty itselleni järjestelmien teknisen tutkimuksen kautta, mutta myös käytettävyyden tutkimisen puolelta. Työnantajalle tästä opinnäytetyöstä on hyötyä monella tasolla. Ensimmäisenä esiin nousee yleinen tietoisuus

virtualisointijärjestelmistä ja sen ohessa osaaminen opinnäytetyössä käytetyistä virtualisointijärjestelmistä. Osaaminen kehittyi näiden lisäksi myös tutkimuksen ja dokumentoinnin alueella, jotka ovat hyvin tärkeitä osaamisalueita tietotekniikan alalla.

Järjestelmäarvioinnissa keskityttiin hankaluuksien, helppokäyttöisyyden ja luotettavuuden kautta saavutettaviin hyötyihin tai haittoihin. Opinnäytetyössä tekemäni asiantuntija-arvio perustuu toimintatutkimuksessa tehtyjen havaintojen tulkintoihin. Tässäkin tutkimuksessa asiantuntijan ennakoasenteet ja kokemus tuotteista vaikuttaa varmasti tutkimuksen lopputuloksiin. Linuxin ja asiantuntijalle ennen tuntemattoman virtualisointijärjestelmän integroiminen yhteen toimivaksi kokonaisuudeksi tuottaa aina haasteita, joita kaupallisissa sovelluksissa ei välttämättä ole niiden osien sulautuessa paremmin yhteen. Kaikki virtualisointijärjestelmät kehittyvät jatkuvasti, kuten myös asiantuntijat ja asiantuntemus. Tästä syystä on erittäin suositeltavaa kokeilla uusia järjestelmiä ja oman osaamisen rajoissa auttaa avoimia projekteja etenemään tarjoamalla omaa osaamista kehittäjäyhteisön saataville. Osaaminen voi olla ohjelmointia, mutta lähes yhtä tärkeää on tarjota osaamista käytettävyyden tai toiminnallisuuden testauksen alueella. On erittäin hyödyllistä olla tietoinen erilaisista mahdollisuuksista ja järjestelmistä, jotta pystyy paremmin arvioimaan eri järjestelmien hyötyjä tai haittoja.

Lähteet

Kirjalliset lähteet

Kuula, A. 1999. Toimintatutkimus: Kenttätyötä ja muutospyrkimyksiä. Tampere: Tammer-Paino Oy.

Kuula, A. 2006. Tutkimusetiikka: Aineistojen hankinta, käyttö ja säilytys. Jyväskylä: Gummerus Kirjapaino Oy.

Leidenius, K. 2012. Tietokonesalit vaeltavat pohjoiseen. Tietokone 8/2012, 22-23.

Rankin, K. & Hill, B. M. 2010. The official Ubuntu server book. 2nd edition. Boston: Pearson Education.

Tuominen-Halomo, A. 2010. Tietokoneiden lämpö lämmittää kohta koteja. Helsingin Uutiset 9.-10. lokakuuta 2010, 5.

Sähköiset lähteet

Anttila, P. 1998. Tutkimisen taito ja tiedonhankinta. Viitattu 24.3.2014.

http://www.metodix.com/fi/sisallys/01_menetelmat/01_tutkimusprosessi/02_tutkimisen_tai_to_ja_tiedon_hankinta/10_tutkimuksen_luotettavuus/10_3toimintatutkimuksen_luotettavuus

Captain KVM. 2013. Recap of the January 2013 oVirt Workshop. Viitattu 27.10.2013.

<http://captainkvm.com/2013/01/recap-of-the-january-2013-ovirt-workshop/#more-438>

CPU World. 2013. Virtualization Technology and AMD-V. Viitattu 5.3.2014.

http://www.cpu-world.com/Glossary/V/Virtualization_Technology.html

Graziano, C. D. 2011. A performance analysis of XEN and KVM hypervisors of hosting the XEN Worlds Project. Viitattu 28.2.2014.

<http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=3243&context=etd>

IBM. Open virtualization with KVM. Viitattu 30.11.2012.

<http://www-03.ibm.com/systems/virtualization/infrastructure/open/>

Miller, W. 2010. Virtualization: Top 10 Virtualization Best Practices. Viitattu 13.10.2013.

<http://technet.microsoft.com/en-us/magazine/gg131921.aspx>

Niemistö, K. 2008. Ympäristöystävällinen IT. Viitattu 28.9.2012.

<http://www.ttlry.fi/sites/ttl.ttlry.mearra.com/files/file-uploads/Tapahtumat/2008/Ymp%C3%A4rist%C3%B6yst%C3%A4v%C3%A4llinen%20IT-Keijo%20Niemist%C3%B6,%20VMware.pdf>

oVirt. oVirt Architecture. Viitattu 26.10.2013. <http://www.ovirt.org/Architecture>

oVirt. 2014. Quick Start Guide. Viitattu 5.3.2014. http://www.ovirt.org/Quick_Start_Guide

Pietroforte, M. 2008. Seven disadvantages of server virtualization. Viitattu 12.10.2013.

<http://4sysops.com/archives/seven-disadvantages-of-server-virtualization/>

Qemu. Viitattu 30.11.2012. <http://wiki.qemu.org/KVM>

Rescorla, E. 2000. HTTP Over TLS. Viitattu 6.3.2014. <https://tools.ietf.org/html/rfc2818>

Saaranen-Kauppinen, A. & Puusniekka, A. 2006a. KvaliMOTV. Havainnointi. Viitattu 25.3.2014.
http://www.fsd.uta.fi/menetelmaopetus/kvali/L6_4.html

Saaranen-Kauppinen, A. & Puusniekka, A. 2006b. KvaliMOTV. Tutkimussuunnitelma. Viitattu 25.3.2014. http://www.fsd.uta.fi/menetelmaopetus/kvali/L2_4.html

Shesterin, D. 2011. The Downsides To Virtualization. Viitattu 12.10.2013.
<http://www.faronics.com/news/blog/the-downsides-to-virtualization/>

Siebert, E. 2009. What is virtualization? Viitattu 1.10.2012.
<http://itknowledgeexchange.techtarget.com/virtualization-pro/what-is-virtualization/>

Viestintävirasto. 2013a. CERT-FI haavoittuvuustiedote 112/2013. Viitattu 13.10.2013.
<http://www.cert.fi/haavoittuvuudet/2013/haavoittuvuus-2013-112.html>

Viestintävirasto. 2013b. CERT-FI haavoittuvuustiedote 025/2013. Viitattu 13.10.2013.
<http://www.cert.fi/haavoittuvuudet/2013/haavoittuvuus-2013-025.html>

Walker, R. 2004. Handbook A4: Getting and analysing qualitative data. Viitattu 25.3.2014.
<http://www.col.org/SiteCollectionDocuments/A4.pdf>

Julkaisemattomat lähteet

Microsoft. 2012. Konesali ilman rajoja. Azlan Offshore 2012 esityskalvot.

Kuvat

Kuva 1. Perinteinen palvelin ja virtualisoitu palvelin (Siebert 2009.).....	12
Kuva 2. Kaksi erilaista tapaa suorittaa virtualisointi. (Siebert 2009.).....	13
Kuva 3. Virtualisointi Linuxissa. (IBM.).....	14
Kuva 4. oVirt arkkitehtuuri (oVirt.).....	22
Kuva 5. Asennuksen aloittaminen	24
Kuva 6. Levyaseman käsittely asennusvaiheessa	25
Kuva 7. Käyttäjätilin luominen	26
Kuva 8. oVirt-paketin asennus	27
Kuva 9. oVirt Enginen asetusmäärittely.....	28
Kuva 10. Local host saatu käyntiin Local storagen lisäämisen jälkeen.	29
Kuva 11. Tietovarastot ja iso-tietovarastoon lisätty levykuva.	29
Kuva 12. Uusi hypervisor lisätty ja hyväksyttävissä.....	31
Kuva 13. Virtuaalikoneen siirto manuaalisesti toiselle isäntäkoneelle	32

Taulukot

Taulukko 1. Windows Server tuotteiden kehitys. (Microsoft 2012.)	15
------------------------------------------------------------------------	----