LAUREA

Kriittisten järjestelmien virtualisointi

Luoma Sami

Nieminen AKi

2010 Leppävaara

Laurea-ammattikorkeakoulu
Laurea Leppävaara

Kriittisten järjestelmien virtualisointi

Sami Luoma, Aki Nieminen
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Marraskuu, 2009

Laurea-ammattikorkeakoulu
Laurea Leppävaara
Tietojenkäsittelyn koulutusohjelma
Yritysten tietoverkot

Tiivistelmä

Sami Luoma, Aki Nieminen

Kriittisten järjestelmien virtualisointi

Vuosi 2010 sivumäärä 76

Tutkimme tässä työssä, miten virtualisoinnin ja kriittisen järjestelmän yhteensopivuudesta voidaan yleisellä tasolla varmistua. Rakensimme asiakkaalle virtualiympäristön VMware tuotteilla. Projektin päätavoitteena oli konvertoida fyysisistä palvelimista virtuaaliseksi Timecon-kulunvalvontajärjestelmä, kaksi domain controller- järjestelmää, DHCP-palvelin, UPS-valvontapalvelin ja tiedostopalvelin. Tarkoituksena oli myös rakentaa alusta tuleville uusille palvelimille. Tulevat järjestelmät voidaan asiakkaan kannalta käsittää kriittisiksi järjestelmiksi. Käytimme tätä asiakasprojektia myös Laurea-ammattikorkeakoulun opinnäytetyö prosessina.

Opinnäytetyön teoriaosuudessa esittelemme yleisesti virtualisoinnin historiaa ja VMware-virtualisointitekniikoita, sen tuoteperhettä sekä lyhyesti tämän järjestelmän valinnan perusteluita alustan valinnassa. Tämän jälkeen esittelemme yleisesti kriittisten järjestelmien määritelmiä ja tutkimme enemmän kulunvalvontaan liittyviä elementtejä ja erityispiirteitä.

Yhteenvetona yritimme koota yhteen niitä asioita, jotka tulee ottaa huomioon suunniteltaessa ja rakennettaessa vastaavanlaista järjestelmää käyttöönottoa varten. Keskitymme opinnäytetyön kirjoittamisessa menetelmiin, joilla järjestelmät ja ympäristöt voidaan turvallisesti ja joustavasti ottaa käyttöön. Tuomme työssä ilmi, mitä pitää erityisesti huomioida tämän muutosprosessin aikana. Käymme lisäksi läpi millaisia valmisteluita projektin toteuttaminen itse ympäristöltä, laitteilta, sovelluksilta ja käyttäjiltä vaatii. Lisäksi mietimme vastauksia kysymykseen, milloin järjestelmä kannattaa virtualisoida.

Lopputuloksena kirjoitimme asioista, joilla vastaavanlainen projekti voidaan mutkattomasti ja turvallisesti suorittaa loppuun. Kokoamme tässä työssä ennakoivia ja täydentäviä toimia, joita kannattaa ja pitää ottaa huomioon vastaavanlaisen projektin toteutusta tehdessä. TimeCon-sovelluksen asennusta virtuaaliseksi voidaan pitää onnistuneena, koska sovellus toimi virtuaalisessa alustassa saumattomasti. Kuvasimme projektin eri vaiheet asennusvaiheesta aina lopputulokseen. Kulunvalvonta järjestelmän muutosprosessin jälkeen projektissa käytettiin vielä aikaa järjestelmän tutkimiseen, seurantaan ja kehitykseen, jossa etsimme ongelmia, virheitä ja parannuskohteita.

Asiasanat virtualisointi, kriittiset järjestelmät, vmware, muutosprosessi

Laurea University of Applied Sciences
Laurea Leppävaara
Information Technology Programme
Business Information Networks

Sami Luoma, Aki Nieminen

Critical system virtualization

Year 2010 pages 76

Abstract

This thesis presents a solution how to implement virtual environment and critical services on the generic level. A virtual environment was built with VMware products for an existing customer. The main objective of the project was to convert physical servers (Timecon access control system, two domain controllers, DHCP server, UPS monitoring server and the file server) virtual. The secondary objective is to build a platform for these individual new servers. In the customer's environment these new systems can be defined as very critical.

The theory section presents a general history of virtualization and VMware virtualization technologies. Different VMware products and brief justification for choosing this platform is also presented. In addition, a general definition of critical systems is introduced and access control elements and their special features are investigated in some more detail.

In summary, an attempt is made to assemble things that should be considered when users are designing and constituting a similar system for deployment. The focus is on methods and aids with which this system and environment can be deployed safely and smoothly to production. In addition, aspects are mentioned that must be particularly considered during this process and what preparations implementation requires from devices, applications and users.

As a final result a list of matters was compiled that shows how similar projects can be completed. Proactive and supplementary actions have been gathered which should be taken into account when creating similar implementation. TimeCon application implementation can be considered a success because the application worked perfectly in a new virtualized environment. The project was described the installation phase to the final results. After the implementation some time was used for the testing, monitoring and developing the environment, problems, errors and improvements were searched

Key words virtualization, critical systems, vmware,

SISÄLTÖ

1 Johdanto

Tämän opinnäytetyön aiheeksi valittiin kriittisten järjestelmien virtualisointi, jossa rakensimme oikeassa tuotantoprojektissa asiakkaalle virtuaaliympäristön VMware-tuotteilla. Tähän rakennettuun ympäristöön asensimme muutamia järjestelmiä, jotka tietojenkäsittelyssä ja asiakkaan näkökulmasta katsoen voidaan käsittää kriittisiksi järjestelmiksi.

Itse otsikko kattaa laajan aihepiirin, mutta keskityimme opinnäytetyössä lähinnä Timecon kulunvalvontajärjestelmän toimintaan fyysisessä ja virtuaalisessa ympäristössä. Muista työhön liittyvistä kriittistä järjestelmistä tuomme esille ainoastaan virtualisoinnissa tapahtuvan konvertointi vaiheen.

Tämä projekti toteutettiin toisen opiskelijan työpaikalla loppuasiakkaalle, asiakkaan haluamien tarpeiden mukaan. Opinnäytetöitä tai muita tietolähteitä suoranaisesti kriittisten järjestelmien virtualisoinnista emme projektin kuluessa löytäneet, joten aihe on sinällään uusi ainakin omalla ammattikorkeakoulutasollamme.

Asiakkaan ilmoittamat vaatimukset järjestelmiltä ja itse virtuaalialustan rakentamisessa käytettävät tekniikat mahdollistivat käyttää tätä projektia myös Laurea-ammattikorkeakoulun opinnäytetyöprosessissa. Ohjaavan opettajan ja koululta tulleiden tiettyjen vaatimusten pohjalta halusimme syventyä erityisesti kriittisten tietojärjestelmien ja virtualisoinnin näkökulmaan.

Työssä käsiteltiin virtualisointia (VMware) ja kriittisiä järjestelmiä (mm. Timecon-kulunvalvontajärjestelmä) teoriassa sekä käytännössä. Lisäksi opinnäytetyössä esitellään VMware-virtualisointitekniikoita ja VMwaren tuoteperhettä, sekä lyhyesti tämän järjestelmän valinnan perusteluita alustana. Tämän jälkeen esittelemme yleisesti kriittisten järjestelmien määritelmiä ja tutkimme enemmän kulunvalvontaan liittyviä seikkoja ja erityispiirteitä. Itse projektissa teimme alustan ja VMware-asennukset ensin valmiiksi, minkä jälkeen asensimme Timecon kulunvalvontajärjestelmän virtuaaliseen ympäristöön.

Esittelemme asennusvaiheet VMwarelle vasta kriittisen järjestelmän yliheiton jälkeen. Oikeassa projektissa rakensimme virtuaaliympäristön palvelimista, jotka toimivat kriittisten järjestelmien alustoina. Tämän jälkeen esittelemme työssä rakennetun kokonaisuuden ja pohdimme järjestelmän toimivuutta sekä tulevaisuutta.

Yhteenvetona yritämme koota yhteen niitä asioita, jotka tulee ottaa huomioon suunniteltaessa ja rakennettaessa vastaavanlaista järjestelmää käyttöönottoa varten. Yritämme opinnäytetyön sisällössä ja ydinaiheena keskittyä menetelmiin, joilla järjestelmät ja ympäristöt voidaan

turvallisesti ja joustavasti ottaa käyttöön. Lisäksi kerromme menetelmistä, joita pitää erityisesti huomioida tämän muutosprosessin aikana. Projektissa on otettu huomioon ainoastaan asiakkaan tarpeita ja vaatimuksia, joiden kautta yritimme luoda itsellemme kuvan siitä, mitä tämänkaltaisen projektin kautta voidaan uudessa järjestelmässä kehittää ja saada aikaan. Lisäksi mietimme millaisia valmisteluita projektin toteuttaminen vaatii itse ympäristöltä, laitteilta, sovelluksilta ja käyttäjiltä.

2 Projektin tarkoitus ja tavoitteet

Projektin päätavoitteena oli konvertoida fyysisestä palvelimesta virtuaaliseksi Timecon-kulunvalvontajärjestelmä, kaksi domain controller-järjestelmää, DHCP-palvelin, UPS-valvontapalvelin ja tiedostopalvelin. Tarkoituksena oli myös rakentaa alusta tuleville asiakkaan uusille palvelimille. Asiakkaalla oli käytössään kaksi toimipistettä, joihin asennettiin omat klusteriympäristöt, joita hallinnoidaan samalla Virtualcentersovelluksella. Toisessa näistä toimipisteistä on yhteensopimaton levyjärjestelmä ESX 3.5:n kanssa, joten se päivitettiin uuteen versioon, joka on yhteensopiva uuden alustan kanssa.

VMware-alustat päivitettin migraation jälkeen uusimpaan 3.5 versioo, koska asennuksen aikana viimeisin versio oli 2.0, Samalla Virtualcenter päivitettiin 2.0 versiosta 3.0:aan. Virtualisoinnin kohteina olivat Windows NT 4, Windows 2000 ja 2003 Server-palvelimet. Ympäristöön asennettiin myös Red Hat Enterprise Linux ja Windows 2008-palvelimia. Lisäksi tarkoituksena oli tuottaa asiakkaalle toimiva palvelinympäristö, vapauttaa lisäresursseja palvelintilaan, sekä tuottaa asiakkaalle kustannussäästöjä. Näiden ohella myös yksinkertaistettiin palvelimien hallittavuutta. Kokonaisuudessaan virtualisoimme noin 70 palvelinta.

Aiheen ja projektin edetessä otimme kantaa muun muassa virtuaaliympäristön ja konsolidoinnin tuomiin etuihin. Yksi opinnäytetyön tärkeistä osa-alueista oli vastata kysymykseen: Miksi ja koska kannattaa konsolidoida ja virtualisoida. Tavoitteenamme oli myös ottaa kantaa laadun ja riskien ymmärtämiseen. Yritimme kuvata myös asiakkaan käyttämät yhtenäiset menetelmät, tekniikat ja käytännöt, jotka tukivat virtualisointiprojektin toteuttamista. Opinnäytetyön teoriaosuudessa ideana oli tuoda esille lähinnä yleisesti virtualisointia, kuvastaa sen käyttöä ja sen tuomia mahdollisuuksia. Lisäksi teimme kriittisten järjestelmien virtualisoinnista teknisemmän tutkimuksen, joka käsitti yleisen katsauksen ESX 3.5:sta sekä Virtualcenter 2.5:n ominaisuuksista. Tätä aihe-analyysiä kirjoittaessa VMware oli julkaissut jo beta- version ESX 4.0 sovelluksesta.

2.1 Lähtötilanne

Asiakkaallamme oli selkeä ongelma sähkönkulutuksessaan, lukuisat palvelimet palvelintilassa käyttivät liikaa sähköisiä voimavaroja pelkkään laitteiden ylläpitoon. Tarkoituksena oli ratkaista tämä ongelma virtualisoimalla suurin osa asiakkaan palvelimista. Tämän lisäksi useasta koneesta oli takuu- ja huoltosopimus menemässä umpeen, mikä aiheutti sen, että asiakkaalle tuli paljon ylimääräisiä kuluja, kun laitteille joudutaan tilaamaan ulkopuolista huoltoa.

Asiakkaan ympäristö oli vuosien saatossa laajentunut huomattavasti, joten heillä oli jäänyt palvelimien vaatimat perusresurssit huomioimatta. Kustannussäästöjen, hallinnan helpottamisen ja vikasietoiduuden kannalta virtualisointi valittiin asiakkaan ympäristöön sopivimmaksi ja parhaimmaksi ratkaisuksi. Ensimmäisenä projektin vaiheena selvitimme kriittisten järjestelmien virtualisointimahdollisuudet. Tämän jälkeen tavoitteena oli aluksi asentaa ESX-palvelimet sekä selvittää niiden laitevaatimukset ympäristöltä (levyjärjestelmä, kytkimet ja itse palvelimet). Aikaisempien projektien kokemuksella pidimme ensisijaisen tärkeänä, että laitteistot olivat yleisesti tuettuja VMware-ympäristön kanssa.

Virtualcenter-palvelin asennettiin myös virtuaaliseksi samaan ympäristöön, jolloin saimme sille korkean käytettävyysasteen. ESX-palvelimen, Virtualcenterin ja oheislaitteiden asennuksen jälkeen aloitimme konsolidoinnin testikoneilla. Seurasimme testikoneiden toimintaa pari viikkoa, jonka läpimenon jälkeen aloitimme tuotanto koneiden ja kriittisen järjestelmien konsolidoinnin.

Molemmat osallistuivat tiedonkeruuseen, mutta karkeajakoisesti Aki Nieminen vastasi pääosin käytännön toteutuksesta itse asiakkaalla. Sami Luoma dokumentoi projektin etenemistä sekä etsi lähde-aineistoa. Ensimmäisten fyysisten laitteiden asennusten jälkeen pystyimme tekemään testejä mistä tahansa VPN-etäyhteyden avulla. Perimmäisenä tarkoituksena projektissa pidimme sitä, että yritys saisi ongelmaansa hyvän ja toimivan ratkaisun konsolidoinnin avulla.

Rajasimme karkeasti ydinaiheen VMware ESX 3.5- ja Virtualcenter 2.5- virtualisointialustoihin, konsolidoinnin tuomiin etuihin ja Timecon kulunvalvontajärjestelmään. ESX-alusta valittiin projektiin, koska se oli selkeästi tuotantorajapintaan tarkoitettu palvelu sekä vastasi asiakkaan tarpeita ympäristössä. Virtualisoinnista teoreettinen osuus rajattiin vain VMware-tuoteperheeseen, koska muuten aiheesta tulisi liian laaja.

2.2 Työn taustat

Tuottavuutensa kannalta virtualisointi valittiin asiakkaan nykyiseen ympäristöön sopivimmaksi ja parhaimmaksi ratkaisuksi. Sopimuksessa tehtiin selväksi, että asiakkaan kaikki palvelimet

halutaan virtualisoida. Senhetkinen fyysinen ympäristö käsitti myös muutamia kriittisiä palveluja ja toimintoja, jotka tultaisiin projektissa myös virtualisoimaan. Kummallakaan projektin tekijöistä ei ollut vielä kokemusta kriittisten palvelimen asennuksesta, testaamisesta tai käytöstä virtuaaliympäristössä.

Olimme mukana projektissa vastaamassa lähinnä virtualisointiin liittyvistä kysymyksistä teknisestä näkökulmasta. Kriittisten järjestelmien asennuksessa uuteen ympäristöön konsultoimme projektin tai lähinnä asiakkaan puolen asiantuntijaa. Yritimme myös tutkia tässä työssä kriittisten järjestelmien (Timecon-kulunvalvontajärjestelmä) implementointia uuteen virtuaaliseen ympäristöön kirjallisuuslähteiden avulla.

Virtualisoinnin osalta tietomme perustuivat projektin yhteenvetoihin asiakkaan puolelta, testauksiin, asennuksiin ja uusien järjestelmien käyttöönottoon ja niiden seuraamiseen. Yritimme työn lopputuloksena muun muassa tuottaa keinoja, joilla vastaavanlainen järjestelmä voidaan helposti ja joustavasti ottaa käyttöön uudessa ympäristössä. Tässä työssä emme ottaneet kuitenkaan kantaa esimerkiksi sovelluksien tai muiden vastaavien toimintojen suunnitteluvaiheessa ilmi tulleisiin virheisiin, koska ne yleensä huomataan vasta järjestelmän käytön aikana.

2.3 Tutkimusmenetelmät

Tutkimme tässä työssä sitä, miten virtualisoinnin ja kriittisten järjestelmien yhteensopivuudesta ja toimivuudesta voidaan muutosprosessin aikan varmistua. Tutkimus kriittisten järjestelmien osalta pohjautui lähes kokonaan kirjallisiin lähteisiin ja virtualisoinnin osalta lähteisiin, omiin kokemuksiimme ja tietoomme sekä virtualisointia tuottavan yrityksen aiempiin onnistuneisiin projekteihin.

Projektin alussa keräsimme lähinnä kriittisten järjestelmien toiminnasta yleistä tietoa ja sen avulla yritimme käsittää sen toimintaa fyysisessä ja virtulaisoidussa ympäristössä. Käytännön toimenpiteenä ja tutkimuskohteena toimi rakennettava virtuaaliympäristö ja siihen asennettava kulunvalvontajärjestelmä. Opinnäytetyön lähteinä käytimme erilaisia kirjallisia ja sähköisiä tietolähteitä, omaa tietoamme ja alan asiantuntijoita. Kokeellisen tutkimuksen perustana toimivat testit testiympäristössä, minkä jälkeen järjestelmät implementoitiin tuotanto ympäristöön. Projektin lopussa keräsimme havaintoja ja tietoja toteutuneen ympäristön tutkimisella. Lopuksi teimme tuloksista yhteenvedon sekä pohdimme sitä, mitä olisimme voineet tehdä toisin, tai saavutimmeko asiakkaalle hyödyllisen ja tavoitetun tuloksen.

Yritimme tehdä työssä listan niistä käytännön toimista, joita vastaavan projektin toteuttamisessa tulisi noudattaa. Lisäsimme listassa myös ohjeita näiden toimintojen soveltamisesta.

Tutkimuksessa ja kehityksessä pyrimme jakamaan ongelman soveltavaan tutkimukseen, sekä yritimme mahdollisuuksien mukaan käyttää omien resurssiemme ja tietämyksemme mukaan kehitystutkimusta. Tässä työssä oli luonteenomaista uuden todellisuuden rakentaminen olemassa olevan tutkimuksen pohjalta. (Järvinen & Järvinen 2000, 102.)

Pyrimme tietenkin mahdollisimman hyvään ja onnistuneeseen lopputulokseen asiakkaan näkökulmasta. Vastasimme siihen, miten virtualisoitu ympäristö voidaan joustavasti suunnitella sekä toteuttaa ja mitä erityisesti tulee huomioida kriittisten järjestelmien käyttöönotossa virtuaaliympäristössä. Koska aikaisempaa kokemusta kriittisten järjestelmien (Timecon-kulunvalvonta) asentamisesta virtuaalisiksi ei meillä ollut, lähdimme liikkeelle kuin mistä tahansa fyysisen kriittisen järjestelmän asennuksesta, koska oletimme järjestelmän toimivan täysin samalla tavalla myös uudessa ympäristössä.

Kriittisten järjestelmien osalta toimimme samalla tavoin kuin aikaisemmin ja itse kulunvalvonta järjestelmän toimivuudesta mietimme onnistuvaa teoriaa ja teimme oletuksia siitä, kuinka yliheitto tulisi toteuttaa. Jotkut kulunvalvonta järjestelmän erityispiirteet muuttivat käsitystämme olemassa olevasta teoriasta, jota jouduimme työn edetessä muuttamaan.

Rakentavan tutkimuksen tavoitteena tässä työssä oli uuden tiedon lisääminen jo olemassa olevan tutkimustiedon ja tehtyjen projektien avulla. Projektin tutkiminen soveltui mielestämme hyvin tähän tilanteeseen, koska siinä perehdyttiin oikeisiin ongelmiin oikeassa tuotanto ympäristössä. Toimintatutkimuksessa, me tutkijoina osallistuimme tutkittavan kohteen toimintaan konsultin tehtävässä muutosagenttina (Järvinen & Järvinen 2000, 129). Teimme osallistuvaa havainnointia, jolloin osallistumme ryhmän toimintaan sen yhtenä jäsenenä. (Uusitalo 2001, 90.)

Aloitimme tutkimuksen määrittelemällä ja tutkimalla perusongelmaa, joka juontaa alkunsa palvelintilan sähkönsaannin ongelmasta. Tästä syystä aloitettu muutaman järjestelmän virtualisointi asetti meille lähtökohdat tutkimaan ja löytämään seikkoja, Jotka tukivat kriittsten järjestelmien muutosprosessin toteuttamista.

Tilaava yritys muodosti meille tutkittavan sekä muutettavan järjestelmän. Lähdeaineistona käytimme virtualisoinnin osalta muutamaa kirjaa, oppimateriaaleja sekä omaa tietotaitoamme. Kriittisistä järjestelmistä meillä ei ollut aikaisempaa kokemusta, joten etsimme siitä lähinnä sähköisiä ja kirjallisia lähteitä. Kyselimme myös kollegoiltamme oliko heillä mahdollista kokemusta vastaavista projekteista. Alustavasti yritimme etsiä taustoja ja syitä, jotka voisivat muutoksessa aiheuttaa ongelmia virtuaali-alustan kanssa.

Kun olimme saaneet rakennettua suunnitelman, tunnistaneet suurimmat mahdolliset ongelmat muutosprosessissa ja arvioineet niiden vaikutukset, aloitimme konkreettisen virtuaalialustan rakentamisen.

Tutkimusongelmassa etsimme oikeita tapoja, joilla voidaan parantaa kriittisen järjestelmän muutosprosessia niin, että se voidaan suorittaa joustavasti ja turvallisesti. käytimme työssä ns. 5-vaiheista suunnittelua. Ensimmäisessä vaiheessa määrittelimme ongelman. Toinen vaihe oli suunnittelua, jossa ongelma pyrittiin ratkaisemaan. Toteutusvaiheen jälkeen, työlle tehtiin arviointi, jossa tutkittiin seurauksia ja tuloksia. Viimeisenä tarkastelimme tuloksia, joita on saatu aikaan tutkimuksessa. (Järvinen & Järvinen 2004, 129-132.)

3 Virtualisointi

Virtualisointi on osoittautunut teknologiaksi, mikä muuttaa nopeasti koko informaatio teknologian maailmaa ja sen keskeisimpiä toimintoja ja perusteita tavalla, joka näkyy alalla päivittäin. Tämän päivän tehokkaat x86 laitteet suunniteltiin alun perin toimimaan yhdellä käyttöjärjestelmällä ja useilla sovelluksilla, mutta virtualisoinnin myötä tätä perinteistä käytäntöä on voitu muuttaa niin, että laitteilla voidaan pyörittää useita eri käyttöjärjestelmiä ja sovelluksia yhdessä tai samassa koneessa samaan aikaan. Tämä lisää laitteistojen hyötykäyttöä ja niiden joustavuutta toimia erilaisissa ympäristöissä. (VMware education services WMware Infrastructure 3 2008, 8-10.)

Virtualisoinnista tietojenkäsittelyssä voidaan käyttää termiä, joka viittaa resurssien "abstraktimaisuuteen". (Wikipedia 2010.)

Yksi hyvä määritelmä virtualisoinnille on tekniikka, millä fyysisten resurssien tekniset piirteen voidaan piilottaa muilta järjestelmiltä, sovelluksilta ja loppukäyttäjiltä, jotka käyttävät näitä samoja resursseja. Yksi tällainen resurssi voi toimia monena loogisena resurssina tai yksi fyysinen resurssi voi näkyä monena loogisena resurssina (käyttöjärjestelmä, sovellus, tallennus kapasiteetti)(Wikipedia 2010.). Perustuntomerkiksi voidaan sanoa toiminnallisuutta jossa käyttäjä voi muuttaa käyttöjärjestelmän kokonaan omaksi ohjelmakseen.(VMware education services WMware Infrastructure 3 2008, 11-12.)

Tällä hetkellä on olemassa monia erilaisia laitevalmistajan ohjelmistoja, joilla voidaan jakaa laite (keskusyksikkö, muistit, kovalevyt, ja verkonhallinta) moneksi itsenäisesti toimivaksi laitteeksi. Laitteista muodostuu ns. virtuaalikoneita, missä voidaan käyttää omaa käyttöjärjestelmää ja sovelluksia kuten "tavallisessa" koneessa. Lähestulkoon kaikki virtuaalikoneet jakavat laitteistoja ja sovelluksia ilman, että se aiheuttaa häiriöitä tai katkoksia ympäristössään tai toisissa laitteissa.

Virtualisointi mahdollistaa usean käyttöjärjestelmän tai sovelluksen käyttämisen itsenäisissä järjestelmän osioissa tai "varastoissa". Ne voidaan mukauttaa yksilöllisiin tarpeisiin, kuten IT-hallintapalveluihin tai verkkoresursseihin. Palvelinteknologiassa virtualisointi tarkoittaa fyysisen palvelintietokoneen erottamista käyttöjärjestelmästä ja palvelinohjelmistosta niin, että samassa koneessa voi toimia useampia käyttöjärjestelmän ja palvelinohjelmiston muodostamia itsenäisiä kokonaisuuksia eli virtuaalipalvelimia. Itse virtualisointi tehdään virtualisointiohjelmistolla, mikä toimii virtuaalipalvelimien kannalta fyysisen tietokoneen kaltaisena alustana. Tämä alusta jakaa virtuaalipalvelimille tietokoneen resursseja normaalien laiterajapintojen kautta. (VMware education services WMware Infrastructure 3 2008, 13-14.)

3.1 Virtualisoinnin historiaa

Tietojärjestelmien virtualisointi ei sinänsä ole kovinkaan uusi keksintö, jos osaa hahmottaa kuinka aiemmat vanhat suuremmat järjestelmät ovat toimineet. Näissä niin sanotuissa Mainframe ympäristöissä ”virtualisointi” otettiin ensimmäisenä käyttöön jo 60-luvulla. Ja vasta vuosikymmeniä myöhemmin, alettiin miettiä ratkaisuja resurssien tehokkaampaan hyödyntämiseen, yleisiin hallinnan ongelmiin ja järjestelmien haavoittuvuuteen edullisemman x86 -raudan päällä. (VMware inc. 2010.)

Virtualisointi itse teknologiana on ollut läsnä jo pitkään, vaikkakin se on ollut niin sanottua näennäisvirtualisointia, mitä ei suoranaisesti voida pitää IT teknologian virtuaalisointina. Tämän tekniikan perinteisempiä ja ymmärrettävämpiä muotoja voidaan esimerkiksi nähdä perinteisessä puhelinverkossa. Näissä verkoissa puhelimet on liitetty toisiinsa omilla liitännöillään, mitkä on tehty puhelinoperaattorien toimesta perinteisellä kytkennäisellä menetelmällä. Näissä PSTN (public switched telephone network) puhelinverkoissa standardisoituneet puhelinjärjestelmät ottavat yhteyden puhelinkeskukseen, mikä vastaa kutsuun ja ilmoittaa linjan olevan auki omalla valintaäänellään. Tämän jälkeen soittaja valitsee numeron perinteisellä soittosarjalla ja valintakeskus yhdistää puhelun oikeaan paikaan. Yhteys avautuu ja toimii automaattisesti linjojen välillä ilman sitä, että puhelun välittäjä joutuu yhdistämään puhelut, niin kuin ensimmäisissä puhelinverkoissa tapahtui. (VMware education services WMware Infrastructure 3 2008, 1-3.)

Tämä uusi kytkennäinen tekniikka automatisoi puhelun jolloin voidaan jo puhua palvelun virtualisoimisesta, koska puhelu voidaan yhdistää oikeaan paikkaan, kunhan tekniikka, kytkeminen ja automaatio toimivat yhdessä halutulla tavalla. Tästä toiminteesta käytettiin aikaisemmin termiä virtualisointi. Jos pystyt kuulemaan kytkentäisessä puhelinverkossasi valinta äänen, niin se tuo mahdollisuuden käyttää maailmanlaajuisesti kaikkia puhelinverkkoja ja myös mahdollisuuksien mukaan sen erilaisia palveluja. (VMware education services WMware Infrastructure 3 2008, 3-6)

Informaatio teknologian alati muuttuvassa maailmassa virtualisointi on kasvanut nopeasti, ensimmäiseksi virtualisoitiin tietoverkkoja (tietoverkot ovat lähinnä tekniikaltaan puhelinverkkoja), joissa avainasemassa olivat reitittimet ja kytkimet. Nykypäivänä verkkoon kytketty työasema voi helposti ottaa yhteyden toiseen koneeseen, riippumatta siitä missä kohde osoitteen kone fyysisesti sijaitsee. Viimeisimpinä muutamina vuosina virtualisoituja osa-alueita on esitelty ja kehitetty useita, kuten verkkoja (VPN – Virtual Private Network), tallennuskapasiteettia (Storage), sekä itse tietokoneita ja käyttöjärjestelmiä. Viimeisestä, sekä tietokoneen että käyttöjärjestelmän virtualisointi tekee molemmista riippumattomia toisistaan. (Lintala 2006, 22.)

Virtualisoinnissa koneen fyysistä prosessointitehoa voidaan hyödyntää useille eri työkuormille, kuten useille eri käyttöjärjestelmille ja niiden erilaisille sovelluksille. Lyhykäisyydessään virtualisoinnissa on siis pyrkimys poistaa riippuvaisuutta fyysisten resurssien väliltä ja tehdä niistä loogisia. (Lintala 2006, 23.)

3.2 Virtualisoinnin mahdolliset edut

Kasvavat vaatimukset tuovat uusia ongelmia ja haasteita palvelimien joustavuudelle. Käytännössä tämä edellyttää erilaisten järjestelmien helppoa käyttöönottoa, toimivia palveluja ja työkaluja niiden ylläpitoon. Jotta näiden uusien palveluiden käyttöönotto olisi kitkatonta, palvelinarkkitehtuurin on tuettava virtualisointia. Virtualisoimalla levy-, palvelin- ja verkkoresurssit sekä verkon tarjoamat lisäarvopalvelut, kuten palomuurit ja palvelimien kuormanjakopalvelut, on mahdollista ottaa käyttöön uusia palveluita nopeasti, kustannustehokkaasti ja laiteriippumattomasti. (Cisco systems 2009.)

Sen lisäksi, että virtualisointi säästää paljon kustannuksia, se myös mahdollistaa järjestelmien kykyä muuttua nykyaikaisten teknologioiden mukana nopeasti ja saumattomasti. (Soyinka 2009, 621.)

Nykyaikaisella laitteella voidaan esimerkiksi ajaa useampaa yhtäaikaista Windows/Linux palvelinta, sekä useita järjestelmiä ja sovelluksia voidaan ajaa samalla fyysisellä laitteella. Virtuaalikoneiden hallinta voidaan suorittaa keskitetysti (esim. Virtualcenter). Uuden palvelimen asentaminen tapahtuu parhaimmillaan alle puolessa tunnissa, lisäksi palvelimia voidaan kahdentaa vikasietoisuuden saavuttamiseksi. Tällä tavalla virtualisointi parantaa tuottavuutta ja reagointikykyä poistamalla sovellusympäristöjen laitesidonnaisuuden ja niihin liittyviä rajoituksia. Palvelin-, verkko- ja tallennusresursseja voidaan jakaa sovelluksille käyttäjän ja ylläpitäjän kannalta helpoimmalla tavalla. (Saarinen 2006, 25.)

Esimerkiksi internet-palveluntarjoajilla ja suurehkoilla teknologia yrityksillä siirtyminen virtuaaliympäristöihin voi merkitä senhetkisten resurssien tehokkaampaa käyttöä ja resurssitarpeiden kuten koneiden, tilojen, lämmityksen ja jäähdytyksen vähentymistä. Pienemmissä ympäristöissä virtualisointi luo erilaisia mahdollisuuksia laitekannan joustavaan ja monikäyttöiseen käyttöön sekä palvelujen parempaan tietoturvaan ja saatavuuteen kustannustehokkaan kahdennuksen kautta. (Kinnunen 2009.)

Virtualisointi voi myös tuottaa käyttäjilleen kustannussäästöjä. Pidemmällä aikavälillä esille tulevat virtualisoinnin muut hyödyt: parantuva tietoturva, mahdollisuus useamman käyttöjärjestelmän hallintaan samassa koneessa, ja palvelujen aiempaa helpompi ulkoistaminen palveluntarjoajien hoidettavaksi. Virtuaalitietokoneet parantavat tietoturvaa sekä fyysisesti, että arkkitehtuurin osalta. Palvelimien käyttöaste kasvaa, koska samaa palvelinta voidaan mahdollisesti hyödyntää useamman sovelluksen käyttöön. Levyjärjestelmien käyttöaste kasvaa, koska keskitetyt levyjärjestelmät ovat useamman sovelluksen käytettävissä. Verkkojen tarjoama kapasiteetti ja vapaat portit ovat kaikkien sovellusten käytettävissä ja helposti allokoitavissa eri VLAN:eihin ja VSAN:eihin aina tarpeen vaatiessa. (Cisco systems 1992-2010.)

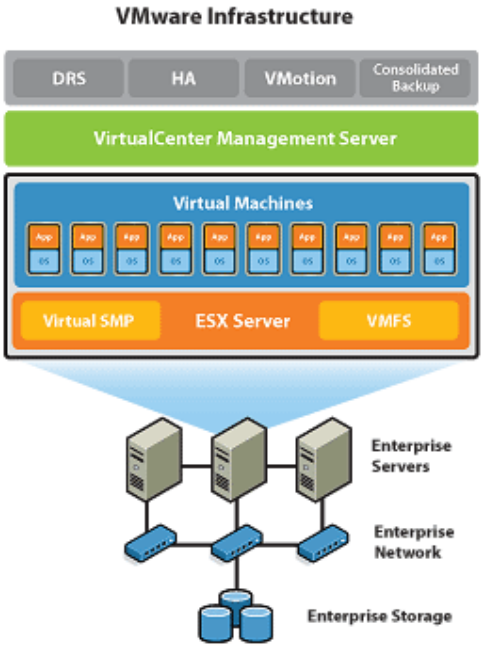
Virtualisoidussa ympäristössä verkkoon integroidut palvelut, kuten esimerkiksi välimuistipalvelimet, sovellusten kuormanjako, SSL-kiihdytys, sovellusoptimointi ja keskitetty tietoturva ovat helposti käsiteltävissä ja saatavilla. Keskitetty hallinta, valvonta ja projisiointi mahdollistavat yksinkertaisemman näkymän datakeskuksen palveluihin. Lisäksi se mahdollistaa uusien palveluiden nopeamman käyttöönoton, sekä mahdollisuuden vastata liiketoiminnan tuomiin haasteisiin entistä nopeammin ja kustannustehokkaammin. Laitekannan käyttöasteen kasvaessa uusia laitteita tarvitaan vähemmän, mikä vähentää sähköntarvetta, konesalien jäähdytystä ja tätä kautta alentaa kustannuksia. (Cisco systems 1992-2010.)

4 VMware

Perinteisillä fyysisillä menetelmillä rakennetuissa datakeskuksissa on tiettyjen koneiden, levyjärjestelmien, verkkoporttien, ohjelmistojen ja sovelluksien välillä tiukka sidos, jota ympäristö joutuu omilla parametreillään ja ominaisuuksillaan noudattamaan. VMware tuoteperheen infrastruktuurin avulla näitä tiukkoja laitteiden ja ohjelmistojen sidoksia voidaan löysätä ja myös kokonaan poistaa.

VMwaren virtualisointiin tehtyä tuoteperhettä kutsutaan nimellä VMware Infrastructure 3 (Vi3). Tähän kokonaisuuteen tuoteperheessä kuuluvat ESX Server 3 sekä palveluiden hallintaan ja käyttöön tehty tuote VMware Virtualcenter. ESX-käyttöjärjestelmän perustana toimii organisaation kehittämä VMkernel, jolla voidaan valvoa ja ohjata järjestelmän erilaisia toimintoja ja keskeisimpiä ajoja. Tuoteperheeseen kuuluu myös erilaisia lisäarvoja tuovia omi-

naisuuksia kuten Vmotion. Näillä tuotteilla käyttäjät voivat virtualisoimalla mm. hallita, optimoida resursseja, nostaa sovelluksien käytettävyyttä ja operatiivista kyvykkyyttä. (Knuutinen, 2008, 27.)



Kuva 1. Yleinen kuvus VMwaren rakenteesta (WMware 2010)

4.1 VMWare ESX Server

ESX Server on lähinnä yritysmaailman tarpeita varten luotu virtualisointityökalu. Sen tarkoituksena on hyödyntää useiden yksittäisten virtuaalikoneiden tuottamia palveluita niin, että koneista saadaan luotua ympäristölle parempaa käytettävyyttä, luotettautta ja tehokkutta, kuin perinteisimmistä VMware-tuoteperheen palvelintuotteista. ESX Server toimii suoraan virtualisointikerroksella (ESX Server hypervisor) omana käyttöjärjestelmänään, joten se ei tarvitse toimiakseen mitään muuta erillistä käyttöjärjestelmää.

Järjestelmän hallitsemiseen käytetään VMkerneliä, joka pohjautuu Linuxin ytimeen. Tämä ydin säästää huomattavasti resursseja eliminoimalla ylemmällä tasolla normaalisti tarvittavan ja toimivan käyttöjärjestelmän, millä hallittaisiin myös virtuaalikoneita. Työkalu pystyy näin suoraan jakamaan hierarkiassa resursseja eteenpäin fyysisiltä laitteilta ja simuloimaan monia klooneja ja kopioita sovelluksista tai resursseista virtuaalikoneiden käyttöön. Järjestelmään on kehitetty myös ”over-commit memory”-ominaisuus, jolla virtuaalikoneet voivat turvallisesti saavuttaa jopa saman muistin käytön prosessien ja suorituskyvyn tasossa, mikä on itse fyysisellä koneellakin. (VMware infrastructure education 2008 install and configure student manual ESX server 3.5 and VC 2.5, 31-32.)

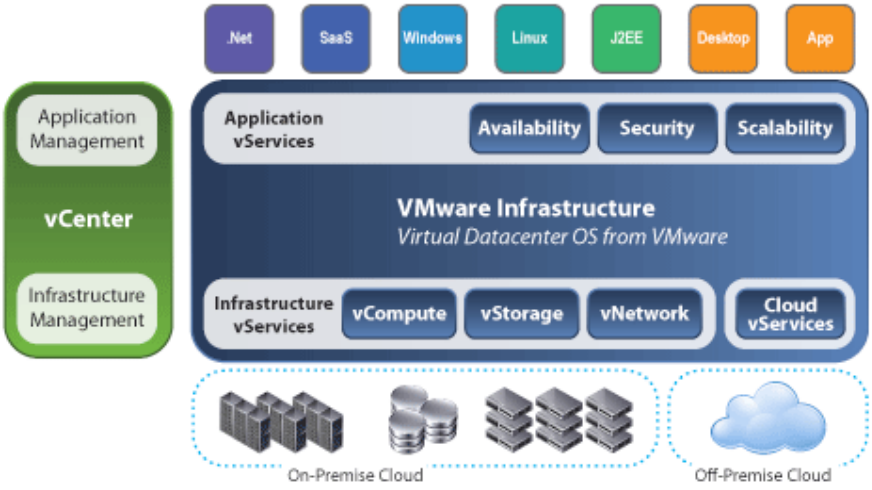
Muistin käyttöä voidaan käyttää ja hyödyntää erillisellä Service Consolilla, mikä toimii samalla hallintaohjelmistona ESX-Serverille ja sen käyttöjärjestelmälle. ESX-serveri on tarkoitettu käytettäväksi ympäristöihin, missä on tarvetta virtaviivaistaa palvelimia, laitteistoja ja sovelluksia resurssien ja kapasiteetin äärirajoille. Sitä voidaan lisäksi käyttää ympäristöissä alusta-, palvelin- ja työasema-asennuksiin.

Käytössä ja ajossa oleville laitteille voidaan myös suorittaa ylläpitotoimenpiteitä niin, etteivät ne kärsi lainkaan katkoista. ESX-palvelin on kokonaan riippumaton laitteistolle, sovelluksille ja käyttöjärjestelmille, joten niitäkin voidaan päivittää tai siirtää uusiin virtuaalipaikkoihin tai ympäristöihin aiheuttamatta itse palveluille minkäänlaista katkosta. Toimiakseen minimivaatimuksilla ESX-palvelin vaatii Intelin tai AMD:n (vähintään 1500Mhz) suorittimen ja muistia vähintään 1GB. Lisäksi siihen tarvitaan pysyvä tallennusratkaisu, mikä kerää informaation virtuaalikoneiden virtuaalilevyiltä (SCSI, RAID, IDE, ATA, SATA), sekä yksi ethernet-rajapintaan sopiva verkkokortti. (Davids 2007.)

4.2 VMWare-virtuaalikeskus

VMware-virtuaalikeskus on keskitetty hallinta työkalu ESX-palvelimille ja virtuaalikoneille, jonka yksilöllisestä graafisesta käyttöliittymästä voidaan ohjata ja hallinnoida koko ympäristön toimintaa. Käytännössä Virtualcenter asennetaan yhdelle palvelimelle, mistä se kytketään kaikkiin ympäristön virtuaalilaitteisiin, mikä mahdollistaa hallinnan yhdestä ja samasta pisteestä. Työkalu tulee tarpeelliseksi lähinnä suurempien virtuaalikoneiden ympäristöissä.

Pääasiallinen funktio on tarjota hallintaa lähinnä konsolisointiin, joka sisältää esimerkiksi ympäristön kaikkien tapahtumien seurannan ja hallinnan, suorituskykyjen ja käyttäjien tekemät hälytykset, kulun/pääsynvalvonnan ja virtuaalisen resurssikartan ympäristöstä, jonka suorituskyvyistä voidaan tehdä vaikka graafisia esityksiä. Järjestelmään voidaan lisäksi luoda erilaisia tehtäviä, jotka tukevat toimintaa tai ilmoittavat hälytyksinä kriittisten toimintojen ajoa. Tällä voidaan esimerkiksi mitata prosessorin käyttöä ja ilmoittaa käyttäjälle sähköpostilla, kun tietty teho/suorituskyky saavutetaan. (VMware infrastructure education 2008 install and configure student manual ESX server 3.5 and VC 2.5, 163-168.)



Kuva 2. Yleinen kuvaus VMwaren palveluista (Clark 2008)

4.3 VMware infrastructure Client

Vmware Infrastructure Clientistä käytetään yleisemmin nimeä VI Client. Tämä graafinen käyttöliittymä on tarkoitettu hallintatyökaluksi virtuaalikoneille. Sen avulla voidaan konfiguroida itse ESX Serveriä. Lisäksi sitä voidaan käyttää Virtualcenterin määrittelyssä, millä voidaan hallita konfiguraatioita ja monitoroida ESX Servereitä ja virtuaalikoneita koko ympäristössä. (Päivinen 2008, 16.)

4.4 VMware client

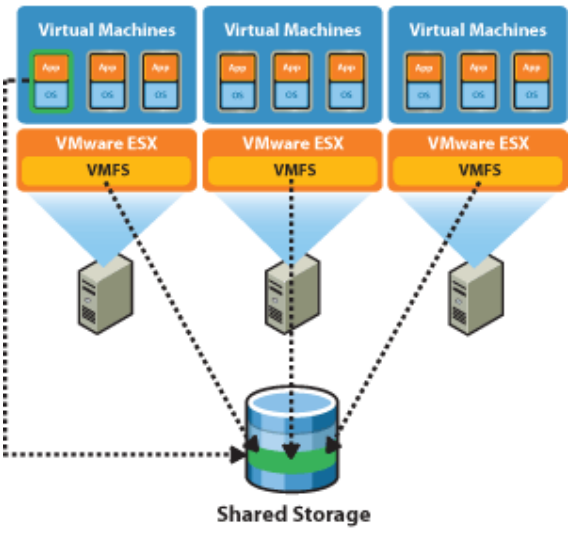
Virtuaalikoneessa ohjelmisto toimii alustana itse koneelle, niin kuin fyysisessä koneessakin, missä pyörii käyttöjärjestelmä ja sen päällä eri sovelluksia. Käyttöjärjestelmä, joka on virtualisoitu, kutsutaan ”guest operating system”:ksi. Yksi tällainen ”questkone” pyörii siis jokaisessa luodussa virtuaalikoneessa, kuitenkin niin, että joka virtuaalikone on täysin itsenäinen, sisältäen omat sovelluksensa, käyttöjärjestelmänsä ja tietoturvansa.

Jos virtuaalikonetta tarkastellaan ESX palvelimen kannalta, niin se voidaan määritellä erittäin luottamukselliseksi kokonaisuudeksi, missä itse laitteeseen yhdistyy konfiguraatio- virtuaalilevy-, ja NVRAM- tiedostot, sekä omat loki hakemistonsa. Virtuaalikoneet ovat helposti siirrettäviä ja käyttäjän on esimerkiksi helppo ottaa niistä varmuuskopioita tai klooneja. Tiedostojen sijainti, sekä helppo käytettävyys tekee virtuaalikoneesta myös helposti hallittavan. Fyysisessä koneessa käyttöjärjestelmä on asennettu suoraan koneen levylle. Tämä vaatii fyysiseltä laitteelta erillisen ajurin laitteelle, mikä tukee järjestelmän toimivuutta. Jos laite päivitetään isompaan tai nopeampaan myös ajurit pitää laitteelle päivittää. Normaalilla käyttäjällä

tämä vaatii myös yleensä asiaan erikoistuneen henkilön konsultaatiota. (VMware infrastructure education 2008 install and configure student manual ESX server 3.5 and VC 2.5, 182.)

Virtuaalikoneet ovat käytännössä sataprosenttisesti ohjelmistoja. Virtuaalikone ei ole tavallaan muuta, kuin ”pino” tiedostoja, mitkä sisältävät mm. tiedostoja virtuaalilevyistä, mitkä korvaavat fyysisen laitteen kovalevyn. Kaikki tiedostot tietylle virtuaalikoneelle sijaitsevat yhdessä ja samassa hakemistossa. Virtuaalikone käyttää standardoituja laite ajureita, jolloin laitteiden päivittäminen onnistuu ilman mitään muutoksia itse virtuaalikoneelle.

Moninkertaiset (multiple) virtuaalikoneet ovat eristettyinä toisistaan, jolloin yhdessä fyysisessä koneessa voidaan ajaa samaan aikaan esimerkiksi tietokanta ja sähköpostipalvelinta. Eristäminen tarkoittaa virtuaalikoneissa sitä, että ohjelmistoriippuvaisuuden ja tai suoritustason kysymykset ja ristiriidat eivät järjestelmässä aiheuta ongelmia. Koska virtuaalikone oli siis periaatteessa vain ”pino” tiedostoja se tarkoittaa sitä, että on hyvin helppoa ja yksinkertaista siirtää koko virtuaalikone uudelle palvelimelle ja toteuttaa laitteistojen päivitykset. Tämä myös tekee katastrofisesta tilanteesta järjestelmän palauttamisen, testaamisesta ja etukäteen suunnittelusta huomattavasti helpompaa. (VMware infrastructure education 2008 install and configure student manual ESX server 3.5 and VC 2.5, 182.)



Kuva 3. Jaettu levyjärjestelmä ESX-koneille, NAS ja SAN-levyt
(Nextor 2008.)

5 Kriittisen järjestelmän määritelmä

Kriittiset järjestelmät ovat tietojenkäsittelyssä ympäristöjä tai laitteita, joiden hajoaminen tai toiminnan pysähtyminen johtaa menetyksiin, joita ei voida missään nimessä järjestelmän, tuotteen tai palvelun elinkaaren aikana sallia. Tällaisia menetyksiä ovat alalla, esimerkiksi

palvelinhuoneessa syntyvä tulipalo ja siitä johtuvat seuraukset. Kriittiset tietojärjestelmät ovat ihmisistä, atk-laitteistoista, tiedonsiirtolaitteista ja sovelluksista koostuvia järjestelmiä, joiden tarkoitus on dataa käsittelemällä helpottaa ja tehostaa toimintaa tai tehdä se ylipäätään mahdolliseksi. (Ridley 1983, 55.)

Kun tietojenkäsittelyssä tutkitaan kriittisiä järjestelmiä, ei voida vain ottaa kantaa itse järjestelmän tai sovelluksen koodiin, vaan siinä tulee ottaa huomioon kokonaisuus, jossa ohjelmisto tai sovellus on käytössä ja sen ympäristötekijät, jotka voivat vaikuttaa järjestelmän toiminnallisuuteen. (Ridley 1983, 55.)

Järjestelmän käyttövarmuus (dependability) on yksi seitsemästä keskeisimmistä termeistä, kun puhutaan tietojärjestelmistä. Ne voidaan jakaa käyttövarmuuden mukaan edelleen kuuteen ominaisuuteen: Luotettavuus, (reliability), saatavuus (availibility), käyttöturvallisuus (safety), luottamuksellisuus (confidentiality), eheys (integrity) ja ylläpidettävyys (maintainability).

Tietoturvallisuus (security) kuuluu olennaisena osana käyttövarmuuteen. Yllä olevien ominaisuuksien olemassaolo järjestelmässä ei ole läheskään aina selvyys. (Ridley 1983, 66, 68, 69.)

Esimerkiksi meidän toteutuksessamme hetkellinen katkos kulunvalvonnassa aiheuttaa kaikkien ovien lukkiutumisen, mutta jos yhteyden palatessa ovet toimivat jälleen normaalisti, se ei välttämättä ole vielä ongelma, jos laitteiden uudelleenkäynnistys ei tule aiheuttamaan muuta ongelmaa itse järjestelmässä tai sen ylläpidossa. Vaikkakin kulunvalvonta on asiakkaalle tuotantokriittinen järjestelmä, se voisi olla myös ei-kriittinen, mutta jos se taas toimii kriittisessä ympäristössä, niin se on kriittinen. Itse työasemat tai palvelimet hyvin harvoin räjähtävät tai syttyvät tuleen, joten tietokoneita voidaan yleisesti pitää kriittisyyden kannalta melko turvallisina laitteina. Kuitenkin esimerkiksi työasemat ja vastaavat sähkölaitteet osana kriittistä kokonaisuutta voivat vaikuttaa merkittävästi onnettomuuksien syntyyn, joten niitä pitää tarkastella ja käsitellä näkökulmasta, jossa niitä pidetään kriittisinä järjestelminä.

Jos halutaan hieman yksinkertaistaa tällaisia mahdollisia vaaratilanteita tai vikoja, voidaan ajatella kolmea erityyppistä osakomponenttia, mitkä näitä vaaroja aiheuttavat. Näitä ovat itse fyysiset osat laitteissa, ohjelmistot ja käyttäjien tekemät virheet. (Sommerville 2000, 78.) Jos halutaan siis parantaa kriittisten järjestelmien turvallisuutta, niin tulee niiden toiminnassa tarkastella kaikkia näitä kolmea osa-aluetta. (Tanhumäki 2006.)

5.1 Kulunvalvonta

Nykyään tietojärjestelmien keskeinen rooli ja sen luomat mahdollisuudet joustaviin ratkaisuihin esimerkiksi toiminnanohjausjärjestelmänä, mahdollistavat tehokkaan järjestelmien käytön yhdessä yritysten kriittisten toimintojen ja erilaisten palvelujen kanssa.

Järjestelmillä on tehtäviensä hoitamiseen kriittisten tilanteiden toiminnallisuuksien hallintaan ja valvontaan räätälöity ympäristönsä ja siihen tarvittavat laitteet ja ohjelmistot. Perinteisesti kriittisiä järjestelmiä on käytetty voimalaitoksissa, terveydenhuollossa ja ilmailu ja avaruusteknologiassa, mutta nykyään yhä useammin niitä löydetään ja käytetään myös muissa "tavallisemmissa" ympäristöissä. Uusien tietojärjestelmien käyttöönoton lisäksi vanhat järjestelmät tarvitsevat uusia ominaisuuksia muuttuvan ympäristön painostaessa sujuvampaan ja tehokkaampaan asioiden hoitamiseen. Järjestelmän muutokset edellyttävät myös ympäristön ja käyttäjien sopeutumista uuteen tilanteeseen. (Tanhumäki 2006, 9.)

Nykypäivän kulunvalvontatekniikka on lähes täysin edistynyttä tietotekniikkaa, jota voidaan hallita ja ylläpitää tietoverkkojen yli. Erilaisten järjestelmien vaativuudet ja nopeat muutokset vaativat kulunvalvonta järjestelmiltä erityisesti joustavuutta ja integroitumiskykyä erilaisiin sovelluksiin ja järjestelmiin. (Peippo 2008, 12.)

Peruskulunvalvonnan lisäksi nämä järjestelmät mahdollistavat muun muassa työaikaseurannan, käyttäjien läsnäolo palvelun ja ruokailuseurannan. Järjestelmän tiedonkeruu mahdollistaa myös käyttäjien tarpeet ja ne ovat integroitavissa helposti esimerkiksi palkanlaskentaan ja henkilöstöhallintoon. Lisäksi se on helposti sulautettavissa rakennuksen turvajärjestelmiin (rikos, palo ja video).

Timecon kulunvalvonta järjestelmän idea on vastata oman ympäristönsä sovelluksista, joissa käyttäjä ei tarvitse kuin yhden tunnisteen saadakseen kaikkien palvelujen edut käyttöönsä. Palveluiden edut ovat selkeästi esillä ja helposti haettavissa. Järjestelmästä voidaan viiveettä esimerkiksi tuottaa erilaisia tarkkoja raportteja ylläpitäjien tai käyttäjien käyttöön. (Peippo 2008, 18.)

Toimiakseen kunnolla Timecon sovellus vaatii alustakseen Windows 2000, XP tai 2003 käyttöjärjestelmän. Tietokantana tulee olla SQL-pohjainen tietokanta esim. DB2, Oracle, MySQL tai Solid. Järjestelmän toiminnan varmistamiseksi on hyvä ottaa kopio itse tietokannasta ja kahdentaa myös sähkönsyöttö. Lisäksi järjestelmän ohjaaminen vaatii samassa verkossa olevan selainpohjaisen client-koneen, mikä mahdollistaa sen, että järjestelmien käyttöön ei tarvitse ostaa erillisiä ohjelmistoja tai tiedonsiirtomekanismeja.

Timecon järjestelmän hallinta, valvonta ja sovelluksien käyttäminen voidaan mahdollistaa yhtenäisellä graafisella käyttöliittymällä, mikä helpottaa sovelluksien toimintaa ja käyttöä.

Käyttöliittymällä voidaan esimerkiksi, ohjata kulunvalvonnan piirissä olevia ovia, määritellä kulkuoikeuksia ja aikaohjauksia, seurata hälytyksiä tapahtumalokista, sekä huoltojen ajaksi kytkeä laitteita tai palveluita irti ympäristöstä. (Peippo 2008, 22.)

5.2 Kulunvalvonnan tarkoitus ja hyödyt

Kulunvalvonnan tarkoitus on ensisijaisesti estää asiaan kuulumattomien pääseminen paikkoihin, jossa siitä voisi aiheutua haittaa yrityksen toiminnalle. Näin voidaan taata samalla omalle henkilökunnalle kulku työtehtävien edellyttämällä tavalla. Lisäksi kriisitilanteen ollessa päällä, mm palo-ovia voidaan lukita pysyvästi tai vastaavasti tiettyjä lukittuja ovia voidaan avata henkilökunnan tai asiakkaiden poistumisreittejä varten.

Sovellusten ja eri järjestelmien integroiminen uusiin alustoihin, sekä tekniikan hyödyntäminen vähentää erillisten järjestelmien määrää ja hallintatarvetta. Toimintojen automatisoiminen sovelluksiin tarjoaa luotettavaa suodatettua järjestelmien tietoa käyttäjän, organisaation tai ylläpitäjän tueksi. Järjestelmiin kirjautuminen, hälytys-, vika- ja vastaavat tapahtumat voidaan joustavasti liittää lokeihin ja asiakkaan vaatimiin informaatiohallinta työkaluihin. Näin kaikkia järjestelmässä syntyviä tapahtumia ja hälytyksiä voidaan tarkastella ja hallita helposti yhdestä sovelluksesta reaaliaikaisen tiedonsiirron, tietokantojen ja käyttöliittymien kautta. (Niscayah 2010.)

5.3 Kulunvalvonnan tulevaisuus

Teknologioiden edelleen kehittyessä erityisesti langattomuus, Internet, TCP/IP -protokolla, selainkäyttö, OPC -liittymät ja PC -pohjaiset järjestelmät tuovat jatkuvasti uusia käyttömahdollisuuksia kulunvalvonnalle ja ovat helpommin sopeutettavissa järjestelmän käyttötarkoituksien muutoksiin (Jussila 2008.)

Nykypäivän laitteistot ja järjestelmät kulunvalvotussa tilassa pitävät sisällään juuri näitä ”älykkäämpiä” järjestelmiä, mille ominaista ja hieman pakollistakin on, muunneltavuus, avoimuus ja integroitavuus. Nämä ominaisuudet mahdollistavat erillisten valvonta ja ohjauslaitteiden kommunikaation ja medioiden yhtenäistämisen tavalla, jota voidaan hyödyntää myös kulunvalvonnassa. (Jussila 2008.)

6 Case: Kulunvalvonnan toteutus asiakasyrityksessä

Timecon kuluvalvonta järjestelmä oli toteutettu asiakasyrityksessä yhdellä palvelimella, sekä sen käyttämä tietokanta oli asennettu lokaalisti yhteen ja samaan palvelimeen. Itse tietokannasta löytyy aina ajan tasalla oleva kopio, mitä päivitetään automaattisesti tietokannan kopi-

oimisella toiselle palvelimelle kerran päivässä. Kuitenkin suurena riskinä tässä voidaan pitää palvelimen hajoamista, koska itse fyysistä järjestelmää ei oltu asiakkaan toimesta klusteroitu, eikä varapalvelinta ollut olemassa. Tällä palvelimella ohjattiin asiakasyrityksen ovia ohjaavia reitittimiä, mitkä olivat kulunvalvonnan kannalta tärkeässä asemassa.

Virtualisoinnin ideana oli lisätä ennen kaikkea vikasietoisuutta järjestelmään ja mahdollisessa katastrofi tilanteessa. Pyrimme tekemään järjestelmän, jossa yhteydet oviin, tietokantaan ja haluttuihin muihin laitteisiin ja sovelluksiin toimivat nopeasti ja varmasti myös järjestelmän palauttamisen jälkeen. Timecon järjestelmän toiminnassa ja ylläpidossa yritimme tunnistaa järjestelmiin liittyvät erittäin kriittiset tekijät ja laadimme häiriötilanteita varten palautumissuunnitelman.

Mahdollisuuksien mukaan testasimme valmiissa virtuaaliympäristössä kriittisten järjestelmien (Timecon) toipumisen ja varajärjestelmien automaattisen toiminnan häiriötilanteessa. Toimivuuden varmistamista varten itse kulunvalvonta ohjelmistoille (Timecon) ja laitteistoille oli tehty tuki-, huolto- ja ylläpitosopimukset asiakkaan toimesta. Ennen päätöstä virtualisoinnista ajoimme palvelin ympäristössä VMware Capacity Plannerin, minkä raportin mukaan Timecon järjestelmä oli täysin virtualisoitavissa. Raportin tulos perustui Timecon järjestelmän suhteellisen pieniin suorituskyky vaatimuksiin. Suorituskyky sovellusta ajettiin 2 viikon ajan asiakkaan tuotanto ympäristössä ennen raportin tekoa.

6.1 TimeCon kulunvalvonta järjestelmän käyttöönotto virtuaaliympäristössä

Tutkimme palvelua siis noin 2 viikkoa ja päädyimme tekemään niin, että otimme fyysisestä koneesta päivää aiemmin varsinaista yliheittoa kloonin, palvelimen ollessa päällä. Tässä oli riskinä se, että itse palvelu ei voinut olla poissa käytöstä yli 30 minuuttia pidempään, koska muuten se automaattisesti sulki kaikki järjestelmään liitetyt ovet. Seuraavana päivänä nostimme virtuaalisen palvelimen pystyyn ja asetimme sen uudella nimellä ja ip-osoitteella verkkoon, jonka jälkeen se myös pudotettiin ulos domainista. Sitten Pysäytimme fyysisen koneen tietokannan otimme olemassa olevasta tietokannasta kopion, joka siirrettiin uuteen virtuaalikoneeseen. Tämän jälkeen ajoimme fyysinen koneen alas ja annoimme virtuaalikoneelle saman ip-osoitteen sekä nimen ja liitimme sen domainiin.

Timecon palvelu lähti toimimaan heti virtuaalipalvelimella, mutta ovien päivitykset ei. Päädyimme uudelleenkäynnistämään ovimekanismien reitittimet, mikä herätti ja päivitti tiedot Timecon palvelulle. Testasimme ovia eripuolelta rakennusta ja ovet toimivat halutusti. Seurasimme vielä loppupäivän järjestelmää, emmekä havainneet virheitä konsolin lokeissa.

Järjestelmän käyttöönotto sujui mielestämme hyvin. Laitteiden (reitittimet, kytkimet, palvelin ja sovellus) tiedot ja asetukset päivitettiin järjestelmään jo ennen itse yliheittoa. Yritimme kartoittaa järjestelmän eri osien vikaantumisherkkyyttä sekä mahdollisten ympäristön muutosten vaikutusta järjestelmään. Uuden ympäristön osalta tällaisia voisivat olla esimerkiksi turvaratkaisujen, sähkönsyötön, laitteiden tai tietoverkkojen häiriöt. Tätä varten teimme asiakkaalle lyhyen ohjeet, jolla voitiin antaa arvio sille, kuinka viasta toipuminen tapahtuu ja kuinka kauan se kestää.

Itse järjestelmässä ei tullut lainkaan fyysisiä laitevikoja, mahdolliset reitittimien toimintahäiriöt tai ovien sähkönsyöttö ongelmat olivat itse järjestelmästä riippumattomia ja tällaisista selvittäisiin todennäköisesti vain fyysisen laitteen tai vioittuneen osan vaihtamisella uuteen, joita on tavarantoimittajalta tilattavissa. Asiakkaalle ilmoitimme, että mahdollisia varareitittimiä tulisi pitää säilössä muutama kappale valmiiksi konfiguroituna, jos mahdollisia ongelmia kohdataan, niin vioittuneet laitteet voidaan vaivattomasti ja nopeasti vaihtaa tuotantoympäristön käyttöön.

Tulimme siihen tulokseen aikaisempien projektien kannalta, että useimmat ongelmat järjestelmän käytössä ovat käytännössä käyttäjä,- lähdekoodi,- tietoliikenne,- ja kantayhteysongelmia, joista selvitään usein sovelluksien ja laitteiden uudelleenkäynnistyksellä tai kannan tarkistuksella. Listasimme lyhyesti mahdolliset yleisimmät ongelmat, joita yliheitossa, valmistelussa ja seurannassa voisi ilmetä:

- verkkoliikenteen latenssit tai resurssien loppuminen järjestelmässä
- itse ohjelmistoviat (lähdekoodivirheet)
- etäyhteyksien käyttäminen ja tietokanta yhteydet.
- järjestelmän itsensä generoimat ongelmat.
- käyttäjien generoimat ongelmat

Emme listanneet tähän kaikkia mahdollisia ongelmia, mutta aikaisempien kokemusten mukaan tämä luokittelu antaa tarpeeksi hyvän lähtökohdan vakavuuden määrittelystä, jossa esim. vika kyseenalaistaa koko järjestelmän toiminnan, mahdolliset viiveet ympäristössä voivat esimerkiksi aiheuttaa ovien avaamisessa odottelua ja viivettä tai kokonaan toimimattomuuden. (Tanhumäki 2009.)

Usein viat ovat myös riippumattomia itse järjestelmästä, mutta verkon aktiivilaitteiden tai ohjelmistopalvelun päivitys voi myös poistaa ongelman. Kahdesta viimeisestä yleensä selvitään laitteiden uudelleenkäynnistyksellä. Itse kulunvalvontaohjelmistossa on valvontajärjestelmä, mikä ilmoittaa ovista, reitittimistä, palvelimesta ja koko ympäristön peruskäytössä

ilmenevistä ongelmista. Samasta käyttöliittymästä käyttäjä näkee myös halutessaan vaikka palvelimen prosessoritehot Timecon sovelluksessa.

Timecon palvelimelta nähdään clientillä avoinna olevat tietokantayhteydet, niiden statukset ja levytilat. Itse tietokannan ylläpito on ulkoistettu kolmannelle osapuolelle. Palvelimelta voidaan myös tutkia eri sovellusten ja laitteiden kuten ovien, reitittimien ja lokitiedostoja. Toiminta mahdollisessa vikatilanteessa riippuu itse ongelman laajuudesta ja laadusta. Ulkoistettu kolmas osapuoli on vastuussa tietokanta- tai tietoliikenne ongelmissa korjaamisesta sovitun vasteajan puitteissa. Käyttäjän on toisin ilmoitettava näistä ongelmista aina eteenpäin itse. Mahdollisia päätelaitteen, sovellusten tai itse palvelimen uudelleenkäynnistämistä tulee välttää viimeiseen asti, koska se saattaa myös joissain tapauksissa tyhjentää järjestelmän palvelimen eri komponenttien lokitiedostoja, joka vianselvitys tilanteessa on ratkaisun kannalta mahdollisesti oleellinen tekijä.

Timecon kulunvalvontajärjestelmän ylläpito on loppukädessä asiakkaan ja käyttäjän vastuulla. Tämä tarkoittaa sitä, että Timecon palvelimelle tehtävät tarkastukset (levytilat, prosessit, tietokanta, varmuuskopiot ja lokit) otetaan itse käyttäjien toimesta. Olemme suositelleet asiakkaalle, että koko järjestelmästä (Timecon palvelin, reitittimet, kytkimet ja sovellukset) tehtäisiin toinen täysin identtinen ympäristö, jolloin mahdollisessa katastrofitilanteessa, koko järjestelmä voidaan vaihtaa uuteen nopeasti ja tehokkaasti. Tällaisen järjestelmän rakentaminen vie asiaan perehtyneeltä henkilöltä asennuksineen noin 4 tuntia.

6.2 Arvio ja tulokset TimeCon järjestelmän yliheitosta

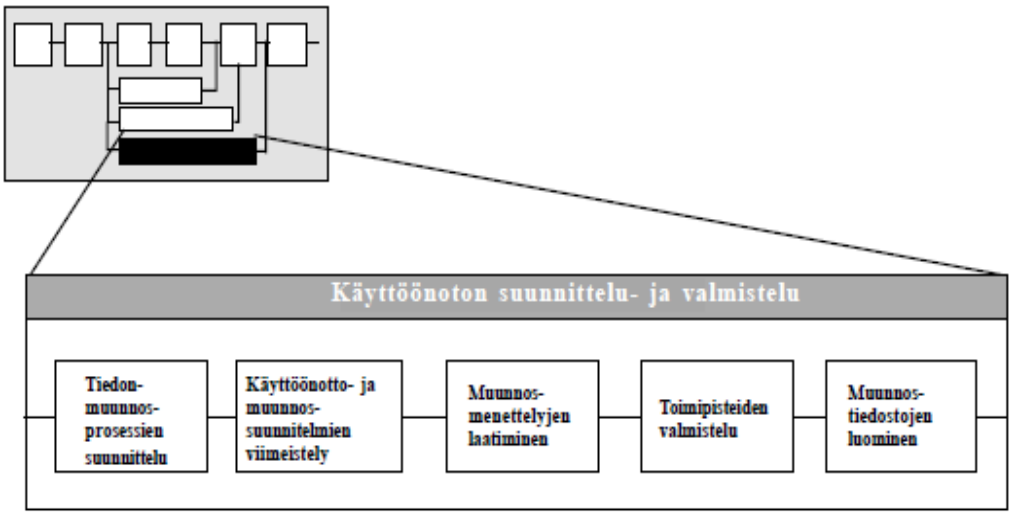
Itse Timecon järjestelmän kannalta tehty muutos ei ollut suuri, koska sitä ei rakennettu täysin tyhjästä, vaan järjestelmä rakennettiin toimimaan samoilla toiminnallisuuksilla vanhan alustan päälle. Muutoskohteena ollut vanha järjestelmä toteutettiin kahdesta pääsyystä, jotka olivat asiakkaalle ennen kaikkea strategiset ja taloudelliset. Eri järjestelmien, sovellusten ja palvelinten hallinta ja palauttaminen helpottui virtualisoinnin myötä huomattavasti, sekä alun perin projektin aloittamisen syy, eli sähkön riittävyys ja niiden kustannukset saatiin minimoitua virtualisoinnin myötä halutulle tasolle. Kriittisten järjestelmien virtualisoinnissa on kyse lähinnä alustan muutoksesta, jolloin ns. ”korvattava” tekniikka ei sinällään aiheuta ongelmia uudessa alustassa, kunhan laitteet, alusta ja sovellukset ovat yleisesti yhteensopivia keskenään. Loppukäyttäjän näkökulmasta ei kulunvalvonta järjestelmällä ole väliä toimiiko se fyysisessä vai virtualisoidussa ympäristössä.

Uusi virtualisoitu alusta vaati myös jonkin verran uutta osaamista. Järjestelmän toiminnot pysyivät Timeconin käyttäjien osalta samana, mutta meidän piti ottaa kantaa tekniikan toimivuuteen ja yleisiin kulunvalvonnan määräyksiin sekä normeihin. Itse yliheitto tapahtui luonte-

vasti, olosuhteet olivat tälle hyvät, testaamisen ja alkutoimenpiteiden jälkeen. Tulimme siihen lopputulokseen että perinpohjaisella ympäristön testaamisella voidaan löytää muun muassa koodi virheitä, jolloin ohjelmisto tai järjestelmän laitteet esimerkiksi reitittimet toimivat eri tavalla kuin on määritelty. Yliheitolle varattiin aikaa virka-aikojen ulkopuolelta, jotta se häiritsisi asiakkaan järjestelmän käyttäjiä mahdollisimman vähän.

Ensimmäisenä tuotantokriittisenä toimenpiteenä mietimme kuinka järjestelmän ja alustan joustavuuden ja toiminnan jatkuvuus voidaan varmistaa myös yliheiton aikana. Pitkiä liikenteen tai järjestelmän katkoksia ei voitu missään nimessä sallia, jolloin käyttöönoton, versiopäivitykset ja tekniikoiden yhteensopivuuden hallinta loivat meille muutamia haasteita.

Tämän casen avulla etsimme vastaisuuden varalle niitä seikkoja, jotka tekivät järjestelmien käyttöönotosta uudessa alustassa sujuvaa ja turvallista. Kirjallisuuden osalta nämä ennakkovalmistelut keskittyvät suurelta osin suunnitteluprosessiin, teoriaan sekä valmisteluun. Yliheitosta ja sen vaikutuksen seurauksesta Timecon järjestelmään emme kirjallista teoriaa löytäneet.



Kuva 4. Yleinen kuvaus projektin käyttöönotosta.

Mietimme järjestelmän käyttöönoton suunnittelussa ja valmistelussa tavoittelussa seuraavia lähtökohtia. Ensimmäisenä tärkeimpänä seikkana täytyi varmistua siitä, että siirtyminen vanhasta järjestelmästä uuteen on yleensä mahdollinen. Laadimme suunnitelman mahdollisia muutoksia varten ja varmistimme, että loppukäyttäjien järjestelmän käyttöönotto lennossa ei aiheuta katkoksia tai generoi ongelmia ympäristöön. Varmistimme, että kaikki tarvittavat informaatiot ja hallintatiedot (konfiguraatiot ja sovellukset) olivat vapaasti saatavilla. Teimme käyttöönotolle aikataulun, jossa näkyi selkeä järjestys sille, kuinka yliheiton tulee tapahtua.

Tässä otimme myös huomioon riippuvuudet järjestelmässä, alustassa ja sovelluksien suhteessa toisiinsa. Teimme samalla myös varovaisen aika-arvion yliheitolle, jonka ilmoitimme asiakkaalle.

Suurin aikaa vievin osa järjestelmän asentamisessa liittyi tiedostojen ja tietokantojen luomiseen, päivittämiseen ja ylläpitoon. Muita tärkeitä kohtia projektin suunnittelussa ja valmistelussa olivat yliheiton jälkeen valvonta ja backupit. Ennen yliheittoa teimme mahdolliselle epäonnistumiselle palautussuunnitelman. Nämä kohdat jo toimivat meille alustavana ohjeena, siitä kuinka haluttu aikataulu pysyy hallinnassa ja eri projektin vaiheet toimivat käytännössä alustan rakentamisessa.

Kriittisen järjestelmän ympäristön valmistelussa arvioimme mahdolliset fyysiset muutokset, tarvittavat lisäkustannukset esimerkiksi uusista laitteistoista ja ohjelmistoista tai niiden toimituksista ja asennuksista. Vastuuhenkilöinä olimme näistä koko ajan hyvin perillä. Vastaavassa suuremmassa projektissa vastuuhenkilöitä tulisi olla useampia.

Asiakkaan tulisi määrätä projektissa selkeät resurssit eri suunnitteluvaiheille ja mahdolliselle ulkoisotetuille tekijöille. Vastaavanlaisessa projektissa suunnitteludokumentissa tulee kuvata tarkasti rakennettava sovellusarkkitehtuuri, tietokanta ja mahdollinen sovelluksen käyttöliittymäsuunnitelma. Lisäksi järjestelmästä tulee tehdä yleinen infrastruktuuri kaavio, jossa hahmotetaan selkeästi järjestelmän fyysinen malli mahdollisten tulevien muutosten kannalta. Tämän jälkeen tulee hakea yliheittoon saatavien valmiuksien hyväksyminen projektilta, käyttäjiltä ja vastaavalta tietohallinnolta. Tarvitaan selkeä aika ja selvitys, siitä mitä muutetaan, miksi ja mihin aikaan.

Käyttöönottovaiheen päätavoitteina pidimme sitä, että käyttöönotto sujuu ongelmitta ja aikataulussa sekä sitä, että järjestelmä saadaan asennettua tuotantoon ja käyttäjät informoidaan onnistuneesta toteutuksesta. Tavoitteiksi pitää myös asettaa se, että yliheiton jälkeen luodaan ympäristö, jossa järjestelmä toimii jatkuvasti ilman suurempia ongelmia. Muutaman viikon käytön jälkeen käyttäjät, sekä projektitiimi arvioivat uutta järjestelmää löytääkseen siitä paranneltavia aiheita, joita käydään läpi kuukausittaisessa palaverissa. Esimerkiksi mahdollisista koodivirheistä, mitkä eivät siis liity ympäristön rakenteelliseen toimintaan tai käyttäjävirheisiin, tehdään virheraportti ohjelmistopuolen toimittajalle.

Kokoamme tässä seuraavissa kappaleissa esitetyn tuloksen ennakoivista ja täydentävistä toimista, jotka kannattaa ottaa huomioon vastaavanlaisen projektin toteutusta tehdessä. Täytyy kuitenkin muistaa, että nämä arviot ovat aina tapauskohtaisia ja suuntaa antavat ohjeet on tehty kirjallisista teoreettisista lähteistä ja oman projektimme toteuttamisen parhaista käytännöistä, eikä niitä välttämättä voida aina suoraan soveltaa muihin vastaaviin projekteihin.

Tämä yksittäinen projektin toteuttaminen sujui myös ilman mahdollisia meistä riippumattomia ympäristötekijöiden ongelmia, jotka myös aina tulee ottaa huomioon mitä tahansa tietoinfrastruktuuri järjestelmää toteuttaessa.

Arvioitaessa järjestelmän yleistä toimivuutta voimme yrittää kartoittaa kriittiset järjestelmän toiminnat ja tilanteet, mitkä voivat generoida mahdollisesti tilanteen, jota ei voida yliheiton tai järjestelmän käytön yhteydessä sallia. Itse sovellusta ja ympäristöä voidaan pitää suhteellisen turvallisena, mutta ongelmatilanteessa näihin epäkohtiin pitää löytyä jokin ratkaisu ennaltaehkäisyllä tai ongelman toteutuessa mahdollinen kiertotie sen ratkaisemiseen.

Järjestelmän kannalta käytännön tilanteissa ongelman tunnistaminen on merkittävä asia. Jos ongelmat voidaan tunnistaa ja niihin reagoidaan tilanteen vaatimalla tavalla, niin järjestelmän käyttöä voidaan jatkaa monissa tilanteissa virheet huomioon ottaen. Järjestelmän tärkeimmät ominaisuudet, joilla sovelluksen käyttö voidaan varmistaa ennalta ovat:

- luotettavuus
- saatavuus
- käyttöturvallisuus
- luottamuksellisuus
- ylläpidettävyys

6.2.1 Alustavat toimenpiteet yliheittoa tehdessä

Suoritimme järjestelmän riittävän testaamisen testiympäristössä, joka oli identtinen tuotantoympäristön kanssa. Tutkimme Mahdollisten aiempien vastaavien projektien vikahistorioita, sekä aikaisempien projekteja. Ennakoimme mahdollisia ongelmatilanteita ja teimme suunnitelmia niiden korjaamisesta tilanteen vaatimalla tavalla.

6.2.2 Kriittisen järjestelmän vaatimus muutokset

Varmistimme käytön turvallisuuden sekä toiminnan muutoskohteissa, joita ovat sovellukset, itse järjestelmä ja alusta. Automatisoimme timecon järjestelmän toimintaprosessit virtualisoimalla. Otimme huomioon mahdolliset muutokset käyttäjän ja organisaation näkökulmasta sekä käyttäjäkoulutuksen, jossa noudatimme yleisiä ja asiakkaan määräyksiä sekä standardeja.

6.2.3 Ennakoivat toimenpiteet ongelmia varten

yritimme tunnistaa ja ennakoida mahdolliset järjestelmäviat, sekä arvioida mahdolliset menetykset (laitteisto- ja taloudelliset kulut). Tässä käytimme apuna järjestelmän monitorointia ja lokien seurantaa. Teimme suunnitelman järjestelmän palauttamiselle (varajärjestelmä, laitteet ja asennuspaketit). Otimme huomioon ongelmatilanteiden hallitsemisen, minimoimisen ja reagoimisen aikaisemmin koetun ja tehdyn kokemuksen perusteella.

6.2.4 Ongelmien aiheuttajat

Mahdollisia ongelmien aiheuttajia järjestelmässä voisivat olla esimerkiksi sovelluksen koodi virheet, ympäristön ja alustan suunnitteluvirheet (yhteensopimattomuudet, kantayhteydet sekä verkko-ongelmat). Lisäksi otimme suunnittelussa huomioon mahdollisten kolmansien osapuolien virheet tai käyttäjien itse generoimat virheet.

6.2.5 Tietoturvallisuus

Tietoturvallisuuden osalta tärkeimmät kohdat projektissa olivat laiteturvallisuus, sekä ohjelmisto turvallisuus. Pidimme huolta, että sovelluksien ja palveluiden viimeisimmät päivitykset olivat saatavilla, tai jo valmiiksi asennettuna. Noudatimme pääsääntöä, joka koostuu tiedon eheydestä, luotettavuudesta ja saatavuudesta. Lisäksi varmistuimme käyttäjien ja ylläpitäjien tietotaidosta, vaikka se ei meille projektissa suoranaisesti kuulunutkaan.

6.2.6 Järjestelmän yliheitossa huomioitavat tehtävät

Yliheittoa tehdessä varmistimm, että käyttäjät ja ylläpitäjät saavat kaikki tarvittavat tiedotteet tulevasta katkosta, lisäksi pyysimme asiakkaalta luvan ja ajankohdan implementaatiolle. Vastaavanlaisessa projektissa tulisi tehdä yksityiskohtainen dokumentaatio toteutettavasta tilanteesta, lisäksi projektissa tulee olla selkeä vastuualueet nimetyille henkilöille. Myös järjestelmän palauttamisesta tulee tehdä selkeä suunnitelma mahdollisen katastrofaalisen ongelman sattuessa.

6.2.7 Yliheiton jälkeiset toimenpiteet

Onnistuneen yliheiton jälkeen tulee seurata järjestelmän toiminnallisuutta, jossa otetaan huomioon mahdolliset kehittämishankkeet, sekä tässä vaiheessa tulee myös raportoida ilmaantuvista ongelmista. Mahdolliset huomatut ongelmat tulee korjata ja niihin tulee reagoida niiden kriittisyyden mukaan.

6.2.8 Yhteenveto täydentävistä toimista

Projektin edetessä mietimme vastauksia kysymyksiin, miten voimme tehdä kriittisten järjestelmien käyttöönotosta ja itse niiden käytöstä mahdollisimman sujuvaa. Totesimme että tärkeimpiä tekijöitä tässäkin projektissa oli ongelmakohtien ennakoiminen, tunnistaminen ja löytäminen. Mietimme mahdollisissa ongelmakohdissa, mikä olisi oikea tapa ohittaa tai poistaa ongelma.

Virtuaalisten järjestelmien ja sovelluksien hyvä tuntemus, sekä aikaisempien hieman samantyylisten projektien kokemus, auttoi ennalta miettimään mahdollisia ongelmakohtia yliheitossa, sekä loi meille kuvan siitä, kuinka tämä projekti tulisi teoreettisella tasolla toteuttaa. Kokemuksemme virtualisoinnista ja sen hyvästä liitettävyydestä auttoi meitä selvittämään esimerkiksi kulunvalvontajärjestelmään liittyviä erityispiirteitä. Otimme vertailukohdaksi yhden aiemman samantyyppisen projektin, josta otimme mallia parhaista käytännön toteutuksista.

Projekti antoi meille yleisen kuvauksen siitä, miten kulunvalvonta ohjelmisto voidaan muuttaa toimimaan täysin virtuaaliseen ympäristöön. Vaikka kriittinen järjestelmä kuvataankin usein sellaisena, jonka välittömässä toimintaympäristössä ongelmat tai mahdolliset katkokset tietojärjestelmässä voivat aiheuttaa vaaraa ihmishengille, tulkitsimme Timecon ympäristön vähintään (tuotanto)kriittiseksi järjestelmäksi asiakasyrityksellemme ja sen yleiselle toiminnalle. Kävimme läpi projektin eri vaiheet asennusvaiheesta aina onnistuneeseen viimeisteltyyn lopputulokseen. Onnistuneen yliheiton jälkeen projektissa käytettiin vielä aikaa järjestelmän tutkimiseen, seurantaan ja kehitykseen, jossa etsimme ongelmia, virheitä ja parannuskohteita.

7 Virtuaalisen alustan rakennus –Llisenssit

Itse alustan rakentaminen alkoi lähes tyhjästä, koska tulevasta uudesta alustasta ja ympärisöstä uusittiin lähes kaikki fyysiset laitteet uusiin. Tilasimme lisenssit projektille samaan aikaan, kuin itse palvelinraudan ja koska jo tiesimme, että käytössämme tulee olemaan Virtualcenter sekä ESX klusteri, niin tarvitsimme käyttöä varten Enterprise tason lisenssin. Enterprise lisenssi kattaa Virtualcenterin, lisenssipalvelimen, Vmotion:in, VCB:n sekä Converter:in. Lisenssipalvelin tuli olemaan sama palvelin, kuin itse Virtualcenter.

7.1 Alusta asennukset

Asiakkaamme päätyi palvelinraudassaan HP:n tarjoamiin räkkipalvelimiin, sillä Blade palvelimien hinta oli heidän mielestään liian korkea. Tämän lisäksi huomasimme että kyseisissä Blade-palvelimissa ei ollut LAN sekä SAN liitäntöjä riittävästi ympäristön tarpeisiin. Yhteensä palvelimia ostettiin seitsemän kappaletta, neljä toiseen toimipisteeseen ja kolme toiseen toimipisteeseen. Ennen fyysisen raudan ostoa käytiin läpi Vmwaren tuki kyseisille palvelimille

ja levypakalle. Pienen haasteen muodosti toisessa toimipisteessä ollut HP:n Eva levyjärjestelmä, joka ei ollut tuettu Vmwaressa sen nykyisessä päivitystasossaan. Levyjärjestelmä jouduttiin päivittämään uudempaan versioon ennen toisen toimipisteen ottamista tuotantokäyttöön. Toisessa toimipisteessä levyjärjestelmänä on EMC Clarion, mikä puolestaan oli ennestään tuettu. Asiakas asetti palvelimet paikoilleen ja pääsimme asentamaan ESX käyttöjärjestelmää palvelimiin. (liite 1)

7.2 ESX asennukset

ESX käyttöjärjestelmä perustuu RHEL:iin, (Red hat enterprise linux), joten pystyimme hyödyntämään paljon kokemuksiamme linux käyttöjärjestelmästä. Lähtötason asennus tehtiin muuten oletusasetuksilla lukuun ottamatta pientä levyosioiden tarkempaa määrittelyä. Nostimme Swap muistin jo tässä vaiheessa 1GB:iin, koska aikaisemmasta asennuksesta oli tiedossa, ettei oletus muisti 268MB tule riittämään, koska palvelimiin asennetaan vielä HP:n oma rautavalvonta ohjelmisto, erillinen valvonta-agentti ja backup- agentti. (Liite 2)

Teimme myös hakemiston /opt omaksi 2GB osiokseen, koska tiedossamme oli, että sovellus tulee kirjoittamaan sinne runsaasti HA (high availability) lokia. Pääosa lokeista kirjoitetaan kuitenkin normaalisti /var osion alle, joten annoimme sille hieman suuremman, 4GB kokoisen levyosion. Tämän jälkeen anoimme koneille ip-osoitteen, maskin, oletusyhdyskäytävän, nimipalvelimet sekä nimen. Asennuksessa ei ole erillistä pakettien valintaa vaan VMware on koostanut konsolille kaikki tarvittavat paketit. Palvelimien uudelleenkäynnistämisen jälkeen pääsimme itse konsoliin kiinni.

Edellisten konfigurointien jälkeen määritimme palvelimille ntp aikapalvelimen. Asiakkaalla oli verkossaan aikaa jakava palvelin, joten teimme siihen liitoksen. Ntp konfigurointi voidaan tehdä joko SSH yhteydellä tai VI Clientillä, päädyimme tekemään tämän SSH:n yli suoraan konsolilta. Käyttäjät voivat etsiä tarvittaessa Ntp konfigurointi tiedostoa /etc/ntp.conf sekä /etc/ntp/step-stickers hakemistojen alta. Kun saimme näiden tiedostojen editoimisen valmiiksi, tallensimme tiedostot ja käynnistimme ne ntp:n komennolla ’service ntpd start’. Komennolla ’ntpq –p ’ pystyimme seuraamaan milloin kiinnitys itse ntp palvelimeen tapahtuu.

Määrittelimme ntp:n uudelleenkäynnistyksessä mukaan komennolla ’chkconfig ntpd on’. Olimme huomanneet, että ajan määrittely on tärkeää ESX palvelimissa, koska koneet ovat rakennettu klusteriin. Luonnollisesti lokien tarkkailu on vaikeaa, jos palvelin ei ole ajassaan tai synkronoitu oikein. säädimme bios:in kellon kohdalleen komennolla ’hwclock --systohc –utp’. Huomasimme myös, että klusteriin liitettävien koneiden host tiedot pitää olla tarkasti määriteltynä, koska koneiden välillä kulkee paljon heartbeat toimintoja, joiden seuraaminen vika-analyysejä tehdessä on tärkeää (muun muassa Vmotion käyttää tätä). Muokkasimme siis

jokaisen ESX koneen /etc/hosts tiedostoa, mihin lisäsimme jokaisen ESX noden FQDN nimet ja lyhyet nimet.

Tämän jälkeen nostimme konsolin muistin VI Clientin kautta 268MB:stä 512MB:een, minkä jälkeen asensimme HP:n rautavalvonta ohjelmiston, sekä kolmannen osapuolen valvonta agentin. Viimeisin versio agentista osaa hyödyntää myös koneiden omaa rautavalvontaa. Kun olimme saaneet konsolin pystyyn asensimme siihen back-up agentin.

Yleisen Tietoturvan kannalta, sekä asiakkaan pyynnöistä määrittelimme ESX palvelimiin lokaalin palomuurin toimintaan. Palomuuri perustuu Linuxin Iptablesiin, mutta sitä komennetaan ESX:n omilla komennoilla. Komennolla ’esxcfg-firewall –q’ nähdään palomuurin konfiguraatio. Laitoimme palomuurin päälle ’esxcfg-firewall --blockIncoming’ sekä ’esxcfg-firewall --blockOutgoing’ komennoilla. Jouduimme vielä avaamaan ntp:lle, varmistusagentille ja valvonta-agentille portit palomuuriin, minkä jälkeen konfigurointi oli valmis. Viimeisenä toimenpiteenä disabloimme SSH:n konfiguraatiosta root kirjautumisen tietoturvasyistä.

7.3 Virtual center asennus

ESX alustojen asennuksen jälkeen asensimme itse Virtualcenterin. Virtualcenter asennettiin yhdelle ESX:n nodelle virtuaalikoneeksi, (Liite 17) annoimme sille 2vCPUn, 3GB muistia, sekä teimme levylle 20GB c-aseman ja 20GB d-aseman, jossa oli käyttöjärjestelmänä 32 bit. Windows 2003 Standard Server. Perusasennuksen jälkeen liitimme Virtualcenterin domainiin, (Liite 19) asensimme virustorjunnan, backup agentin ja valvonta ohjelmiston. Koneen ollessa virtuaalikone, ei sille tarvitse asentaa erikseen rautavalvontaa.

Tietokantojen asennuksessa ensimmäiseksi asensimme Virtualcenteriin Microsoftin SQL Serverin, johon loimme kaksi kantainstanssia. Olimme varanneet ensimmäisen kannan Virtualcenterille ja toisen Update Managerille. Kannat asennettiin d-asemalle, koska mielestämme ne on hyvä pitää erillään itse systeemilevystä.

Virtualcenterin asennus tehtiin hyvin oletusten mukaisesti, valitsimme lisenssipalvelimen asennettavaksi ja osoitimme sille tietokannat. Käytännössä tämän jälkeen asennus on jo valmis. Asennuksen jälkeen määrittelimme lisenssipalvelimen, minkä jälkeen kävimme Vmwaren sivuilta noutamassa lisenssi tiedostot, mitkä yhdistimme yhdeksi tiedostoksi ja osoitimme lisenssipalvelimelle.

7.4 Klusterin määrittely

Klusterin konfiguroinnissa otimme VI Clientilla yhteyden Virtualcenter palvelimeen. Teimme puurakenteeseen hakemiston, mikä nimettiin asiakkaan vaatiman tavan mukaan. Sen alle loimme Datacenterin, sekä määritimme klusterin HA:n sekä DRS:n ominaisuuksineen. Tämän jälkeen Lisäsimme ESX hostit klusteriin ja tarkistimme, ettei liittämisvaiheessa tule virheitä. Kun kaikki hostit oli lisätty klusteriin, tarkistimme että HA:n ja DRS:n ominaisuudet ovat päällä ja automatisoitu ja määritelty konservatiivisen ja aggressiivisen puoliväliin. (Litteet 4 ja 5)

Oletuksena ESX hosteilla oli 60 päivän kokeilulisenssi aktiivisena, mutta määritimme niille kuitenkin seuraavaksi lisenssi palvelimen, että kokeilun jälkeen lisenssin kanssa tule ongelmia. Teimme sen jälkeen resurssipoolit valmiiksi, mikä tarkoittaa sitä, että määrittelimme tuotantopalvelimille sekä testikoneille omat poolinsa. Tämän tarkoituksena oli myöntää tuotantopoolissa oleville virtuaalikoneille maksimiresurssit ja testikoneille minimi. Tämä tehtiin lähinnä sen takia, etteivät testikoneet kuluta tuotantokoneilta resursseja, vaan kaikille taataan yksilöllisesti resurssipoolissa määritetyt resurssimäärät tarpeidensa mukaan. (Liite 6, 7 ja 8)

7.5 Virtuaalikytkimet

ESX hostien verkkoasetuksien liittämiset verkkoon tehtiin niin, että konsoli ja Vmotion löytyvät saman virtuaalikytkimen takaa, mihin on liitettynä kaksi fyysistä verkkokorttia. Teimme vielä kaksi muuta virtuaalikytkintä virtuaalikoneille, mitkä liitettiin samalla tavalla, kuin konsolin ja Vmotion virtuaalikytkimet. Fyysisesti kaapelivedot varmistettiin vielä siten, että jokaisen liittimen verkkokorteista lähtevät kaapelit kytkettiin eri kytkimille ristiin, jolla pystyimme varmistamaan vielä lisää kaapelin vikasietoisuutta. Virtuaalikytkimiin loimme VLAN porttiryhmät, joiden alle sijoitimme itse virtuaalikoneet. (Liite 8)

7.6 SAN levyt

Kuitukortteja oli jokaisella ESX hostilla varattuna kaksi. Varmistimme nämä vetämällä kuitukaapelit ristiin eri kuitukytkimille, joihin teimme 300GB kokoisia LUN-levyjä viisi kappaletta. järjestelmässä Levyt tullaan skannaamaan aluksi yhdeltä ESX hostilta esille, jonka jälkeen ne nimetään, alustetaan ja niille valitaan blokkikoko. Tämän jälkeen muilla ESX hosteilla tehdään vain skannaus ja levyt tulevat näkyviin kaikilla hosteilla ja molemmista kuitukanavista.

7.7 Käyttöoikeudet

Lisäsimme käyttöoikeudet konsolin kautta virtuaalikoneille. Virtualcenterin hallintaan voidaan tässä käyttää Virtualcenter koneen lokaaleja tunnuksia tai domain tunnuksia. Teimme AD:lle oman ryhmän VMwaren hallintaa varten, mihin Virtualcenterin puolelta määritettiin administrator oikeudet myös Virtualcenterin hallintaan. Virtualcenterin lokaali administrator tunnus

toimii järjestelmässä samanlaisin käyttöoikeuksin. Asiakkaan pyynnöstä teimme 24/7 valvontaa varten oman paikallisen tunnuksen, mihin rajoitettiin kuitenkin ’Virtual Machine Power User’ oikeudet. Näillä tunnuksilla käyttäjällä on oikeudet esimerkiksi uudelleenkäynnistää virtuaalikone, mutta ei oikeuksia ESX hostien konfigurointiin tai hallintaan, virhe tilanteiden ennalta estämiseksi (liite 11)

7.8 Update Manager

Jatkoa varten asensimme valmiiksi Update Manager –bundlen Virtualcenteristä. Tämä on täysin automatisoitu päivitysmekanismi ESX koneille. Ennen tätä toimenpidettä oli syytä kuitenkin päivittää Virtualcenter uusimpaan versioon. Latasimme VMwaren sivuilta viimeisimmän asennuspaketin, minkä ajoimme nykyisen version päälle. Asennuspaketista aukeaa ohjeistettu asennus wizard, mitä seuraamalla pääsi tekemään päivityksen vaivattomasti oletus asetuksilla. Päivitystä ennen on kuitenkin hyvä ottaa tietokannasta varmuuskopio. Kun Virtualcenter on päivitetty, voidaan Update Managerilla ajaa valitut päivitykset ESX hosteille, sekä rakentaa baselinet jokaiselle ESX hostille, mihin sisältyy valitut päivitykset. Itse päivityksessä se tapahtuu niin, että update managerilla ajetaan ESX:lle päivitys skannaus, mikä ilmoittaa sen hetkiset saatavilla olevat päivitykset. Tämän jälkeen aletaan ajaa päivityksiä sisään. Update manager ajaa ESX hostin automaattisesti huoltotilaan (maintainence mode) ja hävittää olemassa olevat virtuaalikoneet toisille nodeille ilman katkoa järjestelmässä. Kun vaihe on ohi, alkaa itse päivitys. Järjestelmässämme oli viimeisimmät päivityspaketit koneilla valmiiksi asennettuna, joten meidän ei tarvinnut kyseistä päivitystä asennuksessa tehdä.

7.9 Versiopäivitykset

Tiesimme aikaisemmista projekteista jo sen, että ESX versiopäivitys voidaan tehdä katkotta tuotantojärjestelmiin. Tämä tehdään käyttämällä update manageria, mikä lataa suoraan verkosta päivitykset Virtualcenter koneelle. Kun päivitykset on ladattu järjestelmään, luodaan ESX-koneille päivittyneet baselinet kriittisille, sekä ei kriittisille päivityksille, tämän jälkeen ESX-koneille suoritetaan päivitystarkistus. Sen jälkeen ESX ajetaan päivitystilaan, jolloin virtuaalikoneet ESX:än sisällä automaatisesti migroidaan klusterin toisille ESX-hosteille, jolloin ESX ajautuu huoltotilaan, sekä asentaa päivitykset. Asennuksen jälkeen ESX käynnistyy mahdollisesti uudelleen, jonka jälkeen palvelin poistuu huoltotilastaan automaattisesti klusterin jäseneksi.

Ennen ESX päivityksiä suoritimme kuitenkin Virtualcenter sovelluksen päivityksen, joka tapahtui niin, että laitevalmistajan sivuilta ladattiin ko. virtualcenter ohjelma, mikä ajetaan olemassa olevan virtualcenterin päälle. Toimenpide ei aiheuta palveluihin muita katkoja, lukuun ottamatta hallintayhteyttä virtualcenteriin. Uusimmat Päivitykset sisältävät lisäominaisuuksia

ja tietoturva korjauksia olemassa oleviin versioihin. Virtualcenterin tietokanta päivitetään järjestelmässämme automaattisesti jokaisen päivityksen yhteydessä.

8 Konsolidointi ja muiden kriittisten järjestelmien konvertointi

Tarkoituksenamme oli virtualisoida mahdollisimman monta asiakkaan konetta ja palvelinta. Pääkohteina olivat Timecon kulunvalvonta palvelin, dhcp palvelin, domain controller palvelimia, tiedostopalvelin, sekä UPS valvonta palvelin. Takuu ja huoltosopimukset olivat ehtineet mennä kyseisistä palvelimista jo aikaisemmin umpeen, joten asiakas halusi virtualisoida kaikki nämä palvelimet. Palvelimet olivat kriittisessä roolissa asiakkaan infrastruktuurissa, joten halusimme virtualisoida ne nopealla aikataululla. Yhtenä asiakkaan tavoitteena oli myös saada kyseisille palveluille korkeampi käytettävyys aste.

8.1 Testikoneiden konvertointi ja testaaminen

Aloitimme konvertoinnin muutamalla testikoneella todetaksemme sen, että virtuaalikoneet toimivat moitteetta ympäristössä (Liite 13). Asensimme aluksi VMware Converterin yhdelle rautakoneelle, missä huomasimme että järjestelmän alustalla ei juuri ole väliä, kunhan verkkoyhteydet toimivat Converterin ja konvertoitavan koneen välillä (Liite 14 ja 15). Asennuksen jälkeen osoitimme Converterille vielä lisenssin (Liite 16 ja 17). Converterin Enterprise tason lisenssi mahdollistaa myös konvertoinnin kylmänä , joten teimme sen Converterin käynnistyslevyllä.

Asennuksen tultua valmiiksi aloitimme testin, mikä tehtiin kuumana, eli konvertoitava kone oli päällä. tätä ennen poistimme rautavalvonta ohjelmistot palvelimilta, jottei niistä tulisi olemaan haittaa, kun kone on konvertoitu virtuaaliseksi. Tämä vaatii Windows koneilla uudelleenkäynnistyksen. Lisäksi tarkistimme, että Server Service on päällä, koska Converter agentti vaatii sen toimiakseen järjestelmässä oikein ja määritetyillä tavoilla.

Konversio meni läpi ongelmitta ja virtuaaliklooni oli valmis käytettäväksi testi resurssipoolissa. Huomioimme aikaisemmista projekteista sen, että tässä tulee vaihtaa virtuaalikoneen levyohjain ajuri BusLogicista LSI Logiciin. Jos käyttöjärjestelmä on w2k3, niin tuettu ajuri on LSI Logic. Jostain syystä Converter asentaa aina BusLogicin oletuksilla, koska virtuaalikone toimii myös sillä, mutta LSI Logic on w2k3 palvelimissa vakaampi ja tuetumpi. Tämän jälkeen poistimme vitrtualioidusta koneesta turhat sarjaportit käytöstä tietoturvan lisäämiseksi.

Tämän jälkeen irrotimme fyysisenkoneen verkosta, mutta jätimme sen kuitenkin varmuuden vuoksi päälle, jos konvertointi ei jostain syystä mene läpi ja joutuisimme palaamaan lähtötilanteeseen. Vanhoja koneita on jopa vaarallista välillä sammuttaa, koska vanhat kiintolevyt,

mitkä ovat olleet pitkään päällä voivat helposti rikkoutua jäähtyessään. Kun saimme nostettua virtuaalikoneen ylös Vmware Tools asennus alkoi automaattisesti. (Liite 24)

Kun olimme saaneet Vmware Tools asennuksen on tehtyä, oli kone käynnistettävä uudelleen. Seuraavaksi tarkistimme palvelimen hallinnasta, että kaikki laite ja ohjelmistoajurit olivat asentuneet oikein. Sitten normaaliin tapaan annoimme koneelle ip-tiedot ja kytkimme verkkoadapterin kiinni palvelimeen, sekä tarkistimme samalla, että koneen verkkoadapteri on liitetty oikeaan VLAN porttiryhmään. (Liite 23)

Tämän jälkeen pystyimme testaamaan, miten Vmotion toimi virtuaalisena. Migroimme palvelimen toiselle klusterin nodelle, ja samalla ajoimme virtuaalikoneesta pingiä gateway:tä kohti, joka oli samassa verkossa ja vastasi icmp kutsuihin. Tässä vaiheessa totesimme sen, että normaali-olosuhteissa noin yksi icmp paketti putoaa kutsuissa pois, mutta käyttäjät eivät kykene sitä mitenkään normaalikäytössä huomaamaan.

Tämän jälkeen Ajoimme migraation jokaiselle ESX hostille, jotta varmistuimme siitä, että Vmotion toimii oikein. Teimme myös w2k testikoneella samaisen konversion, jossa on vain huomattava, että palvelin pitää uudelleenkäynnistää, jotta Converterin agentit alkavat toimimaan. Converter tekee uudelleenkäynnistyksen ennen kuin aloittaa kloonauksen. Seurasimme testikoneiden toimintaa pari tuntia, emmekä tänä aikana havainneet mitään ongelmia laitteissa. Tämä antoi meille jo varmuutta siitä, että voimme siirtyä tuotantoympäristöön.

8.2 Domain Controller- koneiden konvertointi

Asiakkaalla oli yhteensä 3 DC palvelinta ympäristössään, joista varmuuden vuoksi teimme aina yhden kerrallaan. Päätimme tehdä konversion kylmänä, ettei AD replikointi häiriinny konversiossa. Ennen kuin AD kone ajettiin alas, tarkistimme muiden AD koneiden toiminnan, jossa kävi ilmi että kaikki on hyvin palvelujen osalta. Otimme huomioon, että koneiden nimipalvelut osoittavat itseensä, koska muutoin haasteena tulee konversion jälkeen palvelimiin kirjautuminen.

Konversio meni onnistuneesti läpi ja samalla laajensimme C-asemaa 10GB -> 20GB. Teimme tämän sen takia, koska levytila oli vähäinen, eivätkä uusimmat päivitykset olisi muuten mahtuneet koneelle. Fyysinen rauta nostettiin ylös, mutta otimme siitä verkkoliittimen irti. Virtuaalinen DC näytti toimivan hyvin, replikointi lähti käyntiin, eikä järjestelmän tapahtumalokiin tullut mitään virheitä.

Seuraavaksi konvertoimme toisen DC palvelimen. Konvertointi meni hyvin, mutta DC koneiden osalta sääntönä on, etteivät ne saisi olla liitoksissa toisiinsa vikasietoisuuden takia. Tämän vuoksi teimme klusterin DRS konfiguraatioon säännön DC koneille, että ne pysyisivät kokoajan eri hosteilla. kun DRS automaattisesti jakaa kuormaa nodejen välillä, se käy sääntötaulukon läpi ja pitää huolen, etteivät DC koneet mene missään vaiheessa samalle nodelle. Tätä emme vielä asennusvaiheessa pystyneet kunnolla toteamaan, koska klusteri ympäristössä oli niin pieni kuormitus, eikä DRS siten automaattisesti siirtänyt virtuaalikoneita hosteilta toisiin. Kolmannen ja viimeisen DC koneen konvertointi onnistui ilman mitään virheitä.

8.3 UPS valvonta palvelimen konvertointi

seuraavaksi otimme työn alle UPS valvonta palvelimen. Emme kuitenkaan saaneet asennusta onnistumaan täydellisenä, koska valvontajärjestelmä itsessään käyttää fyysistä sarjaportti liitäntää UPS palvelimelle. Keskustelimme asiakkaan kanssa asiasta, mutta he päätyivät kuitenkin virtualisoimaan palvelimen. Eli käytännössä teimme konversion koneelle kuumana ja siihen normaalit toimenpiteet, jotta saimme virtuaalikoneen toimimaan. Ongelmaa sarjaportin kanssa mietimme seuraavaan palaveriin.

Ongelman ratkaisemiseksi päätimme liittää sarjaportin yhteen ESX hostiin, johon samalla teimme virtuaalikoneen asetuksiin määrityksen niin, että se käytti hostin eri sarjaporttia. DRS:ään teimme vielä poikkeussäännön, ettei se automaattisesti siirrä konetta toisille nodeille. Tällöin vaarana tosin oli, että jos kyseinen ESX hosti kaatuu, niin silloin HA siirtää virtuaalikoneen toiselle hostille, jolloin taas on vaarana se, että UPS valvonta ei toimi täydellisesti tai ehkä ollenkaan. Asiakkaamme tiedosti riskin ja halusi, että jatkamme suunnitelman mukaan. Tarkistimme että valvonta lähti toimimaan ja ESX hostin lokiin tuli merkintöjä sarjaportin aktiivisuudesta.

8.4 Tiedostopalvelimen konvertointi

Tiedostopalvelimen konvertointi päädyttiin tekemään kylmänä, koska koneella oli kartoitettuna levyjakoja, joiden sisältö muuttui usean kerran päivässä. Ajankohdaksi valittiin ilta, koska palvelimen suuren datamäärän takia konvertointi kestää useita tunteja ja silloin se vähiten haittaisi käyttäjien toimintaa. Virtuaalikoneelle osoitettiin 300GB kokoinen SAN levy koska kyseinen tiedosto tarvitsivat 260GB tilaa. Konvertointi kuitenkin epäonnistui ja sitä kokeiltiin uudestaan pariinkin otteeseen.

Lopulta huomasimme että SAN levy oli määritetty 1MB blokkikokoiseksi, jolloin suurin sallittu tiedostokoko on 256GB. Jouduimme poistamaan kyseisen levyn ESX:ltä kokonaan ja alusta-

maan se uudelleen 2MB blokkikoon mukaan, jolloin tiedoston koko voi olla 512GB. Tämän jälkeen konversio meni onnistuneesti läpi ja seuraavana aamuna palvelin oli käytettävissä.

8.5 DHCP palvelimen konvertointi

DHCP palvelimen osalta konversio tehtiin kuumana. Huomasimme kuitenkin, että palvelimen konvertointi kuitenkin olisi kannattanut tehdä ns. kylmänä koska dhcp- releaset eivät lähteneet aluksi toimimaan. Jouduimme odottamaan hetken, että releaset päivittyivät dhcp palveluun. Tämän jälkeen palvelin nousi pystyyn.

8.6 Microsoft lisenssipalvelimen konvertointi

Microsoft lisenssipalvelimen virtualisoimisessa tuli esiin kaksi haastetta, joista toisesta tiesimme jo ennen konvertoinnin aloittamista (toinen ongelma generoitui ensimmäisestä). Lisenssien hallinta ohjelmisto oli asiakkaalla kiinnitetty fyysisen palvelimen MAC- osoitteeseen ja kovalevyn tunnisteeseen (VolumeID). Tämä oli tehty puhtaasti tietoturvasyistä, ettei ohjelmistoa voida levittää kaikkialle.

Teimme konvertoinnin kuumana, joka meni läpi ilman ongelmia. Otimme alkuperäisestä koneesta MAC- osoitteen ja kovalevyntunnisteen talteen ja irrotimme palvelimen verkosta. Nostimme virtuaalikoneen ylös ja aukaisimme tietokoneen hallinnan, josta menimme device manageriin ja valitsimme ominaisuudet verkkokortista. Täältä käsin pääsimme manuaalisesti syöttämään virtuaalikoneelle alkuperäisen MAC- osoitteen, minkä käyttöjärjestelmä ja lisenssienhallinta automaattisesti havaitsee.

Virtuaalikoneen oikeaa MAC- osoitetta ei voitu muuttaa, koska ESX jakaa virtuaalikoneille vain tiettyä osoite-allasta, jota emme asennuksen jälkeen voineet jälkikäteen muokata. Tarkistimme komennolla ’ipconfig /all’ että MAC- osoite oli vaihtunut. Kovalevyn tunnisteen muokkaamiseen käytimme kolmannen osapuolen ohjelmistoa. Työkalulla vaihdoimme kovalevyn tunnisteen ja tarkistimme virtuaalikoneen nimen komentokehotteessa komennolla dir’. uudelleennimeämisen jälkeen pystyimme testaamaan lisenssienhallinta ohjelmistoa, mikä lähti heti toimimaan.

9 Varmennukset ja järjestelmän palauttaminen

Palvelimien konvertointivaiheen jälkeen tietokannasta otettiin nauhavarmistus mekanismilla kopio. Lisäksi itse palvelimesta tehdtiin tiedostopohjainen backup, mikä ajetaan nauhoille kerran päivässä (incremental) sekä kerran viikossa (full backup). Jatkossa olisi tarkoitus luopua virtuaalikoneiden nauhavarmistuksista kokonaan ottamalla käyttöön Symantec:n virtuaa-

likoneen backup. Siinä ajetaan virtuaalikoneesta backup suoraan fyysiselle SAN levylle, hyödyntäen Vmwaren Snapshot Manageria. Tällä tavoin saadaan full backup ja tiedostopohjainen backup virtuaalikoneesta.

Kulunvalvonta ympäristön osalta kahdensimme verkkoyhteyden toisella varareitillä, minkä asiakasyritys oli omalta puoleltaan toteuttanut. Mahdollisessa ongelmatilanteessa, missä koko järjestelmä on resetoitunut pitää ensiksi varmistaa tietojen ja asetuksien päivittymiset ympäristön laitteille. Ennen järjestelmän uudelleenkäynnistämistä laitteille voidaan ajaa testejä, ei operatiivisilla tiedoilla, jolloin mahdolliset kytkennät tai asetuksien muutokset, eivät aiheuta ympäristölle lisää ongelmia.

Teimme itsellemme ohjeen siitä, kuinka mahdollisesta kriittisestä tapahtumasta järjestelmä voidaan palauttaa ongelmaa edeltäneeseen tilaan. Tällaista tilannetta varten on nimetty projektiryhmästä ja käyttäjistä henkilöt, ketkä osaavat sen tehdä. Nimeämisissä on käytetty asiakkaan henkilöstöä. Tietokannan yhteyshenkilönä toimii kolmannen osapuolen edustaja. Asiakkaan toimintatavan ja organisaation mukaan ilmoitukset tehdään aluksi kolmannelle osapuolelle, paikalliseen Service Deskiin, ja tarvittaville esimiehille, mistä tieto edelleen välitetään asiakkaan puolelle.

Kävimme läpi mahdolliset syyt järjestelmän palauttamiselle ja niiden välittömät seuraukset kriittisessä tilanteessa. Näitä olivat muunmuassa:
- Verkko, välityspalvelin ja sähkösyöttö-ongelmat
- Timecon sovelluksen käyttö-ongelmat tai mahdolliset lisenssit.
- laite ongelmat: reitittimet, palvelimet ja kytkimet
- Järjestelmän tietokannan yhteys, levytila yms. ongelmat.

Recovery-tilanteet aiheuttavat aina katkoksen järjestelmässä. Tietokannat oli asiakkaalla suunniteltu siten, että katko onneksi kohdistui vain osaan kannasta tai taulutilasta, eikä koko kantaan. Kannassa oli 3 eri taulua, mistä sovellukset hakevat dataa. Jos mahdollista vain on vaikuttaa kannan suunnitteluun, niin se tulisi aina suunnitella edellä mainitulla tavalla. Itse kannan kopiointi kannattaa tehdä mahdollisimman hiljaiseen aikaan sovelluksen tai käyttäjien näkökulmasta.

Tilansuunnittelussa tulisi tarkastaa, että vapaata tilaa kannasta löytyy, tila ei saa kasvaa loppuun (häiriöiden määrä lisääntyy progressiivisesti, deadlocks yms.). Uusimmat versio päivitykset tulisi hajauttaa tasaisesti koko vuodelle, koska tämä vähentää organisointi tarvetta ja koneiden kuormitusta. Lisäksi kannattaa ajaa tietokannan indeksoinnit valmiiksi ennen kopiointia, koska huonosti järjestetyn tietokannan käsittely vie selkeästi enemmän aikaa, kuin jär-

jestetyn. Palautuksessa tulee määritellä selkeä toimintasuunnitelma yleisimmille virheille, esimerkiksi fyysisenlevyn hajoamisen yhteydessä. (DBT/YR, 2000)

Palautus suunnitelman teimme alustavasti testiympäristössä, mistä käy ilmi mitä järjestelmän palauttamisessa tehdään ja käydään läpi. Tämä tehdään sen vuoksi, että toiminta oikeassa kriisitilanteessa joustavaa ja ennen kaikkea siinä tiedetään mitä tehdään. Tämä testi tulee tehdä vähintään kerran vuodessa ylläpitävän tiimin kesken. Tavoitteena on kerätä kaikki mahdolliset ongelmat ja epäselvät tilanteet järjestelmän palauttamisen aikana. Mahdolliset muuttuneet elementit tekniikoissa, prosesseissa tai organisaatiossa tulee ottaa vuosittain huomioon. (Liite 28).

Varmistukset tehtiin vanhalla käytännöllä virtuaalikoneiden osalta. Virtuaalikoneisiin asennettiin kuitenkin varmistus agentti, mikä varmistaa koneen tiedostotasolla. Käytännössä varmistus tapahtuu samalla tavalla, kuin kyseessä olisi fyysinen palvelin. Jos virtuaalikone hajoaa, esimerkiksi käyttöjärjestelmä korruptoituu alustan kaaduttua, joudutaan asentamaan uusi virtuaalikone vastaavalla kokoonpanolla, kuin alkuperäinen ja palauttamaan nauhoilta sitten sen tiedostot. Vaikka tämä onkin vanhanaikainen menetelmä, se on kuitenkin helpompi, kuin rautatasolla tehtävä vastaava toimenpide. Jos resurssit antavat myöden, voidaan kriittisten palvelimien osalta tehdä varakoneet valmiiksi, joko perusasennuksella tai tekemällä koneesta valmis image.

Kriittisten järjestelmien osalta VMware tarjoaa virtuaaliympäristöön Snapshot Managerin, jota projektissa olisi voitu hyödyntää. Työkalulla voidaan ottaa virtuaalikoneesta sen hetkinen ”otos”. Käytännössä tätä voidaan käyttää esimerkiksi ohjelmistojen tai käyttöjärjestelmien päivityksen yhteydessä, mikä toimiin näin: Ennen palvelimen päivittämistä SnapShot Manager laitetaan päälle, joka ESX :n puolella alkaa kirjoittamaan muutoksia toiseen vmdk tiedostoon, joka automaattisesti nimetään ’delta-vmdk’ tiedostoksi. Snapshot:teja voidaan ottaa tarvittaessa useampiakin ja niitä hallitaan Virtualcenterin kautta. Snapshot Manageria ei kuitenkaan kannata jättää päälle kauaksi aikaa, koska se kuluttaa kokoajan LUN:in levytilaa. Eli jos päivityksen jälkeen virtuaalikone näyttää toimivan hyvin, otetaan viimeisin tilanne käyttöön ja poistetaan vanhat jäänteet. Jos taas jokin menee pieleen, voidaan hallinnan kautta palata alkuperäiseen ja poistaa uudempi versio virtuaalikoneesta.

Tulevaisuudessa varmistukset asiakkaalla tullaan muuttamaan toisenlaisiksi. Mahdollisuuksien mukaan voidaan käyttää myös VMwaren VCB varmistusta, tai kolmansien osapuolien tarjoamia tuotteita. Suosittelimme asiakkaalle Netapp sovellusta, jolla voidaan ottaa virtuaalikoneista täysi varmistus muutamassa sekunnissa. Varmistuksien kierto voidaan ajoittaa Netapp:n hallinnasta. Järjestelmän palautus tapahtuu myös nopeasti. Netapp:n varmistusjärjestelmä hyö-

dyntää Vmwaren Snapshot mekanismia. Emme kuitenkaan saaneet projektin aikana asiakkaalta päätöstä uudesta varmistustekniikasta.

Projektissa ESX:n varmistukset hoidettiin nauhavarmistuksilla, eli asensimme koneisiin ja palvelimiin varmistus agentit. Varmensimme kaikki tiedostot, lukuun ottamatta virtuaalikoneiden tiedostoja, koska kyseisistä tiedostoista ei saanut otettua varmistusta lainkaan. Tämä johtui siitä, että tiedostot olivat varattuja, eli kun virtuaalikone on päällä, se kirjoittaa ja varaa omia tiedostojaan.

Jos järjestelmän koneet ovat ESX klusterissa, on kriittisessä ongelmassa nopeampaa asentaa kokonaan uudestaan kyseinen ESX, kuin siihen kuuluvat koneet. Asennus vie määrittelyt mukaan lukien alle kaksi tuntia, ja kun kyseessä on klusteri, ei katkoja virtuaalikoneisiin tule lainkaan.

10 Milloin järjestelmä kannattaa virtualisoida?

Tällä hetkellä maailmalla on lukemattomia palvelimia ja sovelluksia, jotka käyttävät vielä vanhaa Windows 2003 server-käyttöjärjestelmää. Monet yritykset varmasti miettivät uusia päivitysprojekteja Windows 2008 server-käyttöjärjestelmän markkinoille tulon jälkeen. Tällaisessa tapauksessa tulee mielestämme miettiä muitakin, kuin perinteisiä ratkaisuja, koska usein uusilla ratkaisuilla voidaan lisätä tuottavuutta ja esimerkiksi tietoturvaa.

Kuten olemme aikaisemmin todenneet, virtualisointi tuo uudenlaista joustavuutta, hallintaa ja kustannussäästöjä käyttäjilleen. Nykyisissä useimmissa palvelinsaleissa koneiden käyttöaste on noin 10-15%:n tasolla. Meidän mielestämme Virtualisointi projektin aloittaminen tuleekin aloittaa ensimmäiseksi konesalista. Erilaiset projektit ja niiden käyttämät räätälöidyt vaatimukset laitteistoilta ja ohjelmistoilta luovat pohjan vastaavanlaisten hankkeiden tarpeellisuudelle ja käytölle, jotka varmasti muovautuvat käyttäjien, organisaation tai ympäristön mukaan.

Kuitenkin lähtökohtana voidaan pitää sitä, että ei kannata virtualisoida erittäin raskaita sovelluksia, jotka kuormittavat palvelimia paljon. Tällaisissa palvelimissa on yleensä monta suoritinta, paljon muistia, tiukat vaatimukset tukemilleen laitteille, sekä suuret vasteajat ajamissaan toiminnoissa. Lisäksi oppivat ja liian ”älykkäät” sekä eksoottiset sovellukset, jotka yrittävät mukauttaa itseään ympäristön vaatimiin resursseihin todennäköisesti eivät sovellu virtualisoitavaksi. Tästä esimerkkinä voisimme mainita exchange 2007 ohjelman, joka pyrkii itsessään säätämään toimintaansa laitteistoresurssien mukaan. (Hämäläinen, 2010)

Ensimmäisenä käytännönläheisenä tavoitteena pidämme käyttöasteen parantamista, joka voidaan saavuttaa virtualisoimalla vähiten kuormitettuja koneita ja palvelimia. Lähinnä parhaita kohteita voisi perustapauksessa olla tulostus- ja tiedostopalvelimet, testipalvelimet sekä pieniä käyttäjämääriä palvelevat applikaatio- ja tietokantapalvelimet. Näiden viimeiseksi mainittujen palvelimien käyttö-aste on noin 5-15%.

Jos laitteiston rautapuolella on ikää muutamia vuosia ja siellä ajetaan perinteisiä sovelluksia ja palvelimia eivät ne vie virtualisoituna tehoja nykyaikaisista monisuoritinpalvelimesta lähes lainkaan. Käytännössä jopa kymmenen vanhojen koneiden palvelut voidaan siirtää yhteen ajanmukaiseen palvelimeen. (Hämäläinen, 2010.)

Omassa projektissamme käytetty kulunvalvontajärjestelmä oli joustava sovellus, joka testaamisen jälkeen näytti toimivan virtuaalisessa ympäristössä hyvin. Tässä pidimme kuitenkin mielessä, että kaikilla virtualisointi valmistajilla on omat standardit tuetuille laitteille ja ympäristöille, joten asianmukainen tietämys ja toiminta on projektin aloituksessa välttämätöntä.

Tärkeimpänä seikkana itse pitäisimme lähinnä loppukäyttäjän tai tilaajan kokemuksia. Virtualisoinnissa on mielestämme kyse kuitenkin konkreettisesti paljon muustakin, kuin vain kustannussäästöistä, joita monet tahot pitävät tärkeimpänä kriteerinä. Varmasti myös tyytyväiset käyttäjät ovat myös tuotteliaampia kuin tyytymättömät. (Reimaa, Tietoviikko, 2010.)

Virtualisoinnilla on muitakin konkreettisimpia etuja. esimerkiksi testiympäristöissä ja ohjelmistokehityksessä on helppo rakentaa identtinen virtuaalipalvelin, joka kuitenkin on täysin itsenäinen ja erillään tuotantoympäristöstä. Tällä tavoin voidaan nähdä muutosten seuraukset häiritsemättä alkuperäistä ympäristöä. Kopioimalla virtuaalipalvelin asetuksineen ja määrityksineen saadaan pystytettyä uusi sovellusympäristö erittäin nopeasti.

projektissamme pyrimme rakentamaan asiakkaalle (käyttäjille ja ylläpitäjille) järkevän ja toimivan ympäristön sen yleisen käytettävyyden ja hallinnan kannalta. Jos tällainen ratkaisu toimii myös kriittisissä järjestelmissä ilman mitään ongelmia, kannattaa se meidän mielestämme virtualisoida. Kasvavassa ja muuttuvassa virtualisoidussa ympäristössä on vaivatonta siirtää resursseja, virtuaalipalvelimia ja sovelluksia tarpeen mukaan pienempään tai suurempaan laitteistojen tai ympäristön vaatimaan palvelukonseptiin.

Vastaavan projektin tekijän pitää miettiä, mitä käyttöjärjestelmiä ja sovelluksia sekä niiden lisenssejä ja versioita on jo ennestään ympäristön käytössä. Pitää olla tiedossa, voidaanko ne siirtää uuteen virtualisoituun ympäristöön sellaisinaan, vai pitääkö sovelluksia mahdollisesti alkaa asentamaan uudestaan. Kaikki ohjelmistot eivät edes ole virtualisoitavissa valmistajan tuen puuttuessa. Lisäksi tulee muistaa, että Itse tuotteiden ja virtuaaliratkaisujen toimitta-

minen on vasta hakemassa muotoaan, joten niissäkin kannattaa vielä virtualisoinnin aloittamisessa olla tarkkana. Käyttäjän kannalta emme näe sille eroa, missä hänen käyttämäänsä sovellusta tai käyttöjärjestelmäänsä ajetaan, kunhan vain kaikki käyttäjän tarvitsemat osat järjestelmiin toimivat.

Mielestämme omassa rakennetussa virtualisoiduissa ympäristöissämme saavutetaan parempi saatavuus ja ennen kaikkea käytettävyys-aste, koska palvelut toimivat vikasietoisemmassa ympäristössä. Vanhojen järjestelmien osalta Virtualiosointia (konvertointi fyysisestä virtuaaliseen) kannattaa pitää lähinnä hätävarana, koska usein uuden virtuaalisen järjestelmän tai sovelluksen tilaajat pitävät uutta virtuaaliympäristöä täysin uutena järjestelmänä, vaikka tosiasiassa vain alustan rautapuoli on muuttunut. Vanha järjestelmä on siis edelleen samanlainen kuin ennen.

Esimerkiksi Windows NT palvelimien virtualisointi poistaa vanhan raudan hajoamisen vaaran, mutta itse järjestelmä on edelleen täysin sama. järjestelmän ylläpitäjien pitää siis edelleen hoitaa virtualisoidun sovelluksen tai palvelun päivittäminen normaaliin vanhaan tapaan. Tällaisissa tapauksissa järjestelmän virtualiosointi kannattaa suorittaa uudelleen asennuksella virtuaaliseksi, mikäli vain on mahdollista.
Mahdollisia virtualisointiprojekteja mietittäessä organisaatioiden tulee kuitenkin tarkistaa aikataulut koska pääoman tuottoastetavoitteiden täyttyminen vie usein oletettua enemmän aikaa. Organisaatioiden on otettava realistisempi ote virtualisointiprojekteihin, sanoo Computacenterin johtaja Paul Casey. Monet eivät saavuttaneet tavoitetuottoaan, mutta syynä on se, että he käyttivät virtualisointipalveluntarjoajan ennusteita ja mittaustyökaluja. Ne vetävät tuloksia kotiin päin eivätkä anna realistista kuvaa suhteessa organisaatioon. (Casey, 2010.)

Työkaluja käytetään myös väärin. Casey jatkaa: ”Organisaatiot ovat usein liian optimistisia mitatessaan tuottoa. Käyttämällä samoja työkaluja me saimme aivan eri tuloksia kuin itse organisaatiot.”

Ongelma ei piile kuitenkaan vain työkaluissa. Caseyn mukaan virtualisointia ja siihen sijoitetun pääoman tuottoa pitäisi tarkkailla pidemmällä tähtäimellä kuin mitä nykyisin tehdään. Pääoman tuottoa ei ole helppo mitata ja se pohjautuu kustannussäästöihin. Todelliset hyödyt ovat joustavampia, esimerkiksi että yhtiö siirtyy tuhatpaikkaisesta rakennuksesta viisisataapaikkaiseen. Tämän tyyppistä laskelmaa ei ole helppo laatia. ”Virtualisoinnille on ehdottomasti tarvetta enkä näe kysynnän vähenevän. Hallinta ja teknologiat on vain mietittävä uudestaan. Suunnittelu on avainasemassa”, Casey (2010) sanoo.

Toteutuneet kustannussäästöt saadaan selville pidemmän ajan seurannan tuloksena ennen virtualisoitia ja sen suorittamisen jälkeen, eikä niihin ole yksiselitteistä kaavaa tai laskelmaa joilla kustannuksia voidaan nopeasti laskea. Tehostuneet resurssien käytön mukana tulevat säästöt on kuitenkin selkeä tavoite, joihin vaikuttavat mm. tilat, sähkönkäyttö ja laitteiden huolto.

Virtualisoinnin asetelmaa voi kuvata kustannustehokkuudessa tuottoastetavoitteella (ROI), jossa siis organisaatio ilmoittaa sitomilleen rahoille ansaitun tuoton. Sijoitetun pääoman tuottoprosenttia lasketaan monella eri tavalla laskijasta ja käytöstä riippuen. Pääoman tuottoa ei ole helppo mitata ja se pohjautuu kustannussäästöihin. (Kauppi, Tietoviikko, 2010.)

11 Yhteenveto ja johtopäätökset

Projektin edetessä mietimme vastauksia kysymyksiin, miten voimme tehdä kriittisten järjestelmien käyttöönotosta ja itse niiden käytöstä mahdollisimman sujuvaa. Totesimme että tärkeimpiä tekijöitä tässäkin projektissa oli ongelmakohtien ennakoiminen, tunnistaminen ja löytäminen.

Mietimme mahdollisissa ongelmakohdissa, mikä olisi oikea tapa ohittaa tai poistaa ongelma. Virtuaalisten järjestelmien ja sovelluksien hyvä tuntemus sekä aikaisempien hieman samantyylisten projektien onnistumisten kokemus auttoi miettimään ennalta mahdollisia ongelmakohtia yliheitossa, sekä loi meille kuvan siitä, kuinka tämä projekti tulisi teoreettisesti toteuttaa.

Kokemuksemme virtualisoinnin mahdollisuuksista ja sen hyvästä liitettävyydestä auttoi meitä selvittämään muun muassa kulunvalvontajärjestelmään liittyviä erityispiirteitä. Otimme vertailukohdaksi yhden aiemman samantyyppisen projektin, josta otimme mallia parhaista käytännön toteutuksista.

Tämä työ antoi meille yleisen ja mielenkiintoisen kuvauksen siitä, miten kulunvalvontaohjelmisto voidaan muuttaa toimimaan täysin virtuaaliseen ympäristöön. Vaikka kriittinen järjestelmä kuvataankin usein sellaisena, jonka välittömässä toimintaympäristössä ongelmat tai mahdolliset katkokset tietojärjestelmässä voivat aiheuttaa vaaraa ihmishengille, tulkitsimme Timecon-ympäristön vähintään (tuotanto)-kriittiseksi järjestelmäksi asiakasyrityksellemme ja sen yleiselle toiminnalle.

Työssä käsiteltiin ja toteutettiin Timecon-sovellus toimimaan uudessa ympäristössä virtuaalisena. Tässä työssä kuvasimme projektin eri vaiheet asennusvaiheesta aina onnistuneeseen viimeisteltyyn lopputulokseen. Onnistuneen muutosprosessin jälkeen projektissa käytettiin

vielä aikaa järjestelmän tutkimiseen, seurantaan ja kehitykseen, jossa etsimme ongelmia, virheitä ja parannuskohteita.

Lopputyön kirjoittaminen oli suhteellisen haasteellinen urakka. Ongelmia aiheutti projektissa asiakkaan ympäristön ymmärtäminen ja niihin halutun lopputuloksen soveltaminen. Kun lopputyön sisältö ja perimmäinen idea saatiin valmiiksi mietittyä, kirjoittaminen oli suhteellisen nopeaa. Omat päivätyömme kuitenkin lykkäsivät omia aikataulujamme paljon. Lisäksi koska työn tekijöitä oli kaksi, aikataulujen sovittaminen ei ollut niin helppoa, käytännössä aikaa löydettiin ainoastaan viikonloppuisin. Työn kirjoittamista kriittisistä järjestelmistä ja niiden muutoksista hankaloitti lähdemateriaalien huono saatavuus, joten jouduimme yhdistelemään ja keräämään tietoa monista eri lähteistä, mikä oli suhteellisen hidasta.

Omasta mielestämme Timecon-järjestelmän virtuaalisointitoteutusta voi projektina pitää onnistuneena, koska kaikki asiakkaan vaatimukset täytettiin. Itse virtuaalisena toimiva järjestelmä toimi lähes samanlaisena kuin fyysisenäkin. Erot tulevat esiin vasta suuremmassa mittakaavassa virtualisoinnin osalta.

LÄHTEET

Sähköiset lähteet:

Cisco Systems Inc. 1992-2007. Viitattu 14.03.2009
<http://www.cisco.com/web/FI/solutions/datacenter/architecture/tco_home.html>

Clark, J. 2007. Review of VMware virtual center. Viitattu 16.08.2009
<http://clark-technet.com/2007/03/review-of-vmware-virtual-center>

Davis, D. 2009. VMware virtual center benefits. Viitattu 10.02.2009
<http://www.petri.co.il/vmware-virtual-center.htm>

Davids, D. 2007. What is VMware ESX server and why you need it. Viitattu 15.07.2009
<http://www.trainsignaltraining.com/what-is-vmware-esx-server-and-why-you-need-it/2007-12-10/

Hämäläinen, P. Tetokone. 2010. Verkkovoimaa virtuaalisesti. Viitattu 28.02.2010
<http://www.tietokone.fi/lehti/fallback/verkkovoimaa_virtuaalisest_1058>

Jussila, M. 2008 tietoturvaintegraatio. Viitattu 17.01.2009
<http://www.fidelix.fi/pdf/Artikkeli_Turvaintegraatio_080224_Fidelix_Jussila_v1.1.pdf>

Kauppi, E. Tietoviikko. 2010. Viitattu 14.10.2009
<http://www.tietoviikko.fi/kaikki_uutiset/article367625.ece>

Kinnunen, O. 2009. Kannattaako virtualisoida ja miksi?. Viitattu 06.01.2010
<http://www.cisco.com/web/FI/expo2009/documents/Olli_Kinnunen.pdf>

Knuutila, J. 2008. X48-pohjaisten palvelinten virtualisointi
https://publications.theseus.fi/bitstream/handle/10024/1712/Knuutinen_Jarkko.pdf?sequence=1

Leveson, N G. Evaluation of software safety. Viitattu 11.11.2009
<http://delivery.acm.org/10.1145/110000/100327/p223leveson.pdf?key1=100327&key2=8527339521&coll=GUIDE&dl=GUIDE&CFID=65300207&CFTOKEN=44205923>

Lintala, A. 2006. Ohjelmistojen virtualisointi. Viitattu 10.4.2009
< https://oa.doria.fi/bitstream/handle/10024/5049/TMP.objres.380.pdf?sequence=1>

Microsoft. 2009. Viitattu 10.06.2009
<http://www.microsoft.com/systemcenter/softgrid/default.mspx>

Niscayah Group. 2009. Teknologiaratkaisut. Viitattu 17.04.2009
<http://www.niscayah.fi/Templates/Page____6410.aspx>

Oy Hedengren ab. 2006. Turvallisuustekniikka. Viitattu 10.03.2008
<http://www.hedengrensecurity.fi/paatuoteryhma_security?kategoria=10200>

Peippo, M. 2008. Kulunvalvonta osana yritysturvallisuutta. Viitattu 11.07.2009
<http://tekniikka.ncp.fi/turvallisuustekniikka/files/tiva/peippo/kulunvalvonta.pdf>

Päivinen, V. 2008. VMware infrastructure järjestelmän käyttöönotto.

https://publications.theseus.fi/bitstream/handle/10024/1224/Insinoorityo_Ville_Paivinen.pdf?sequence=1

Reimaa, R. Tietoviikko. 2010. Viitattu 09.09.2010
<http://www.tietoviikko.fi/msareena/msteemat/article372590.ece>

Saarinen, J. 2006. Palvelinten siirto virtuaalialustalle. Viitattu 11.3.2010
< https://oa.doria.fi/bitstream/handle/10024/30149/TMP.objres.397.pdf?sequence=1>

Tamtron solutions. 2009. Työajanseuranta ja kulunvalvonta. Viitattu 14.09.2008
<http://www.tamtronsolutions.fi/ohjelmistot.html>

Tanhumäki, H. 2006, Kriittisten tietojärjestelmien muutoksen hallinta. Viitattu 12.12.2008
<http://www.cs.uta.fi/research/thesis/masters/Tanhuamaki_Hannu.pdf>

Tietokantojen suunnittelun pienryhmä. 2000. DB2-tietokantasuunnittelu. Viitattu 12.10.2009
<ftp://ftp.software.ibm.com/software/emea/fi/db2/ug/tietokantasuunnittelu.pdf>

Valtionvarainministeriö. 2009. Viitattu 05.05.2008
<http://www.vm.fi/tietoturvasanasto/sisällys.htm>

Wikipedia. 2010. Viitattu 13.08.2009
http://en.wikipedia.org/wiki/Virtualization

VMware inc. 2009. Announcing VMware 4. Viitattu 12.04.2009
<http://www.vmware.com/>

Painetut lähteet:

Järvinen, P. & Järvinen, A. 2000. Tutkimustyön metodeista. Tampere: Opinpajankirja.

Leveson, N.G. 1990. Evaluation of software safety. Addisson-Wesley

Uusitalo, H. 2001. Tiede, tutkimus ja tutkielma, Johdatus tutkielman maailmaan. Helsinki: WSOY.

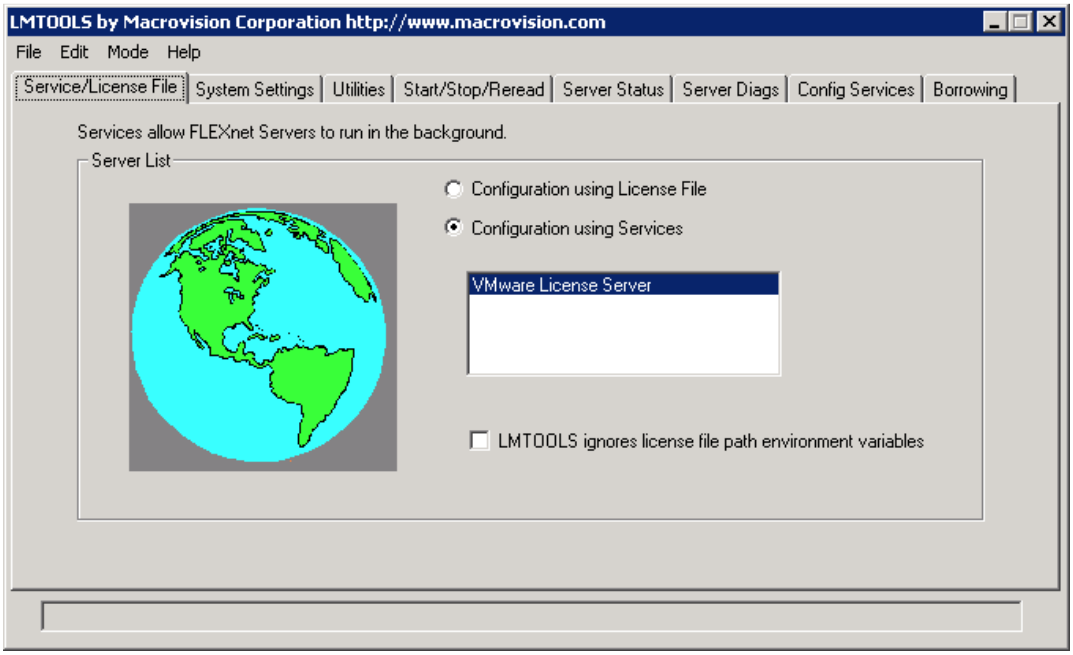
Ridley, J. 1983. Safety at work. London: Butterworths.

VMware infrastructure 3: install and configure student manual, ESX server 3.5 and VC 2.5

Wale, S. 2009. Linux Administration: A Beginner's Guide. Yhdysvallat: McGraw Hill

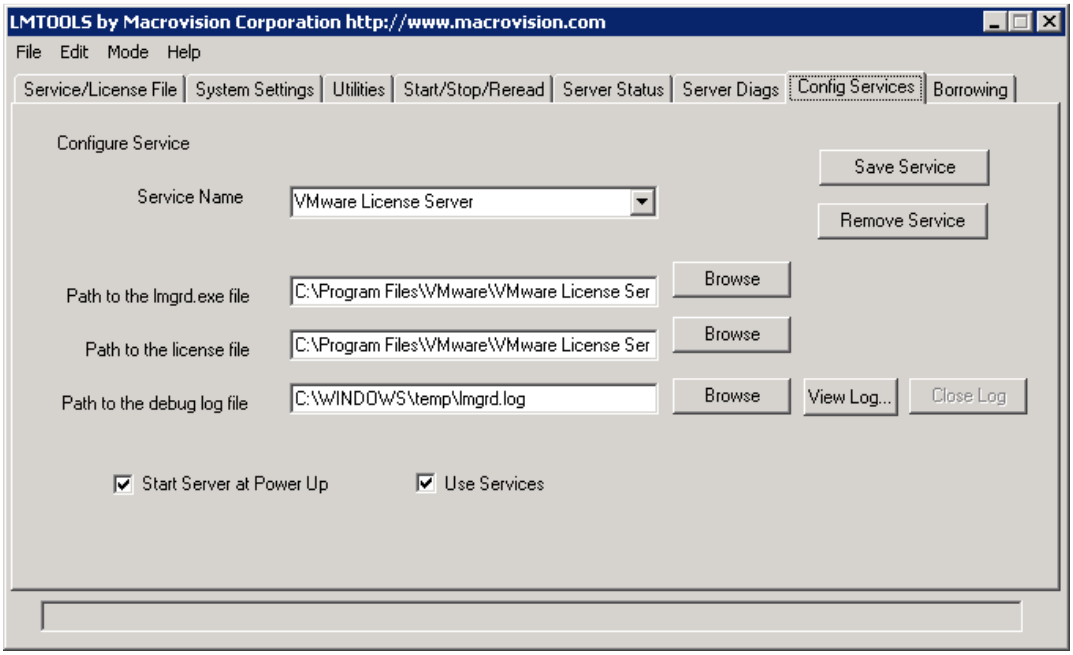
LIITTEET

Alustan lisenssointi



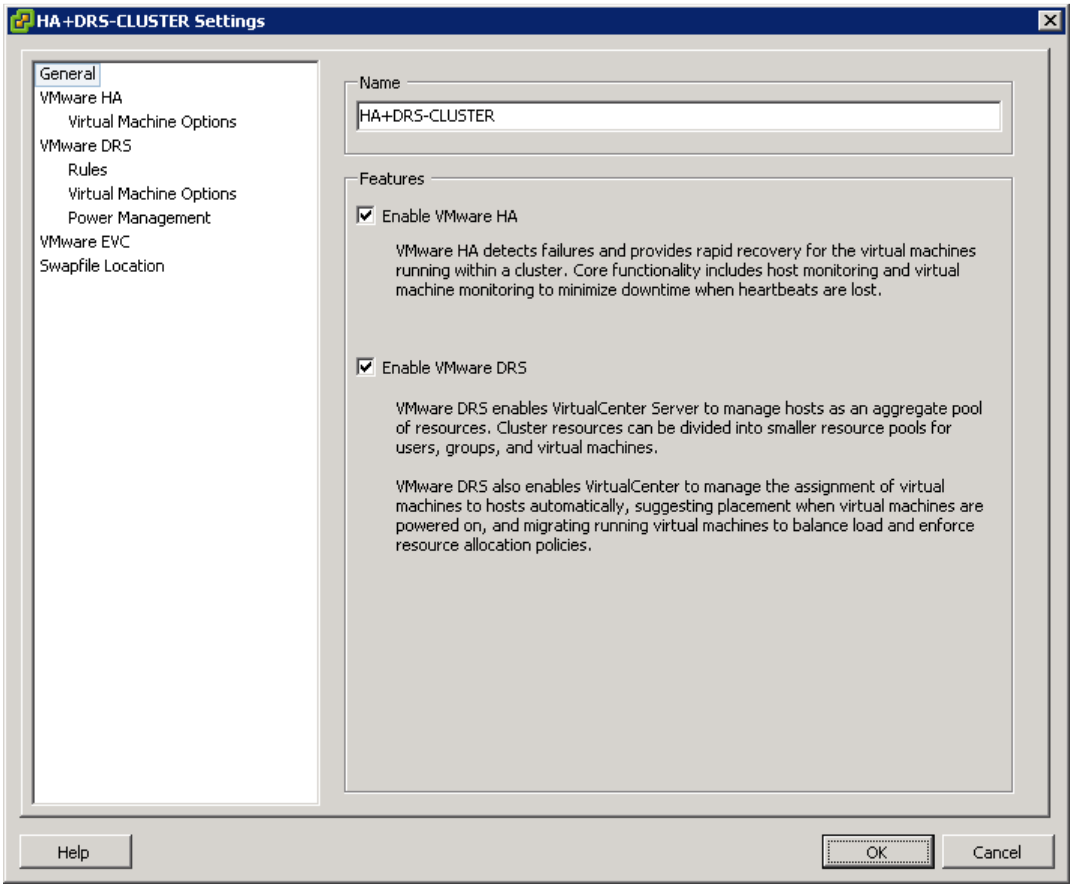
Lisenssi-palvelimen konfigurointi, avataan lisenssipalvelimen konfigurointiohjelma.

Lisenssin määrittely



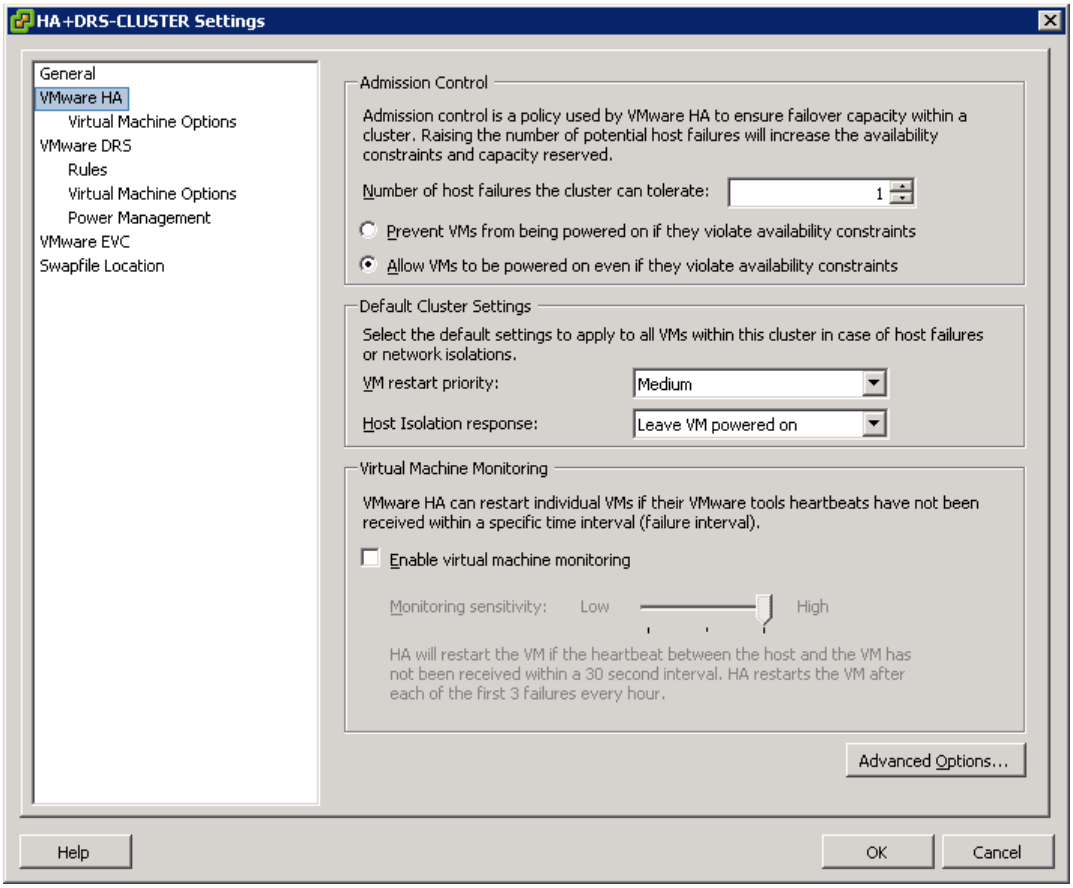
Määritellään lisenssipalvelimelle lisenssi tiedosto, joka on noudetetu VMWaren sivuilta. Lisenssitiedosto sisältää lisenssit virtualcenterille ja ESX koneille.

Lisenssin klusteri asetukset



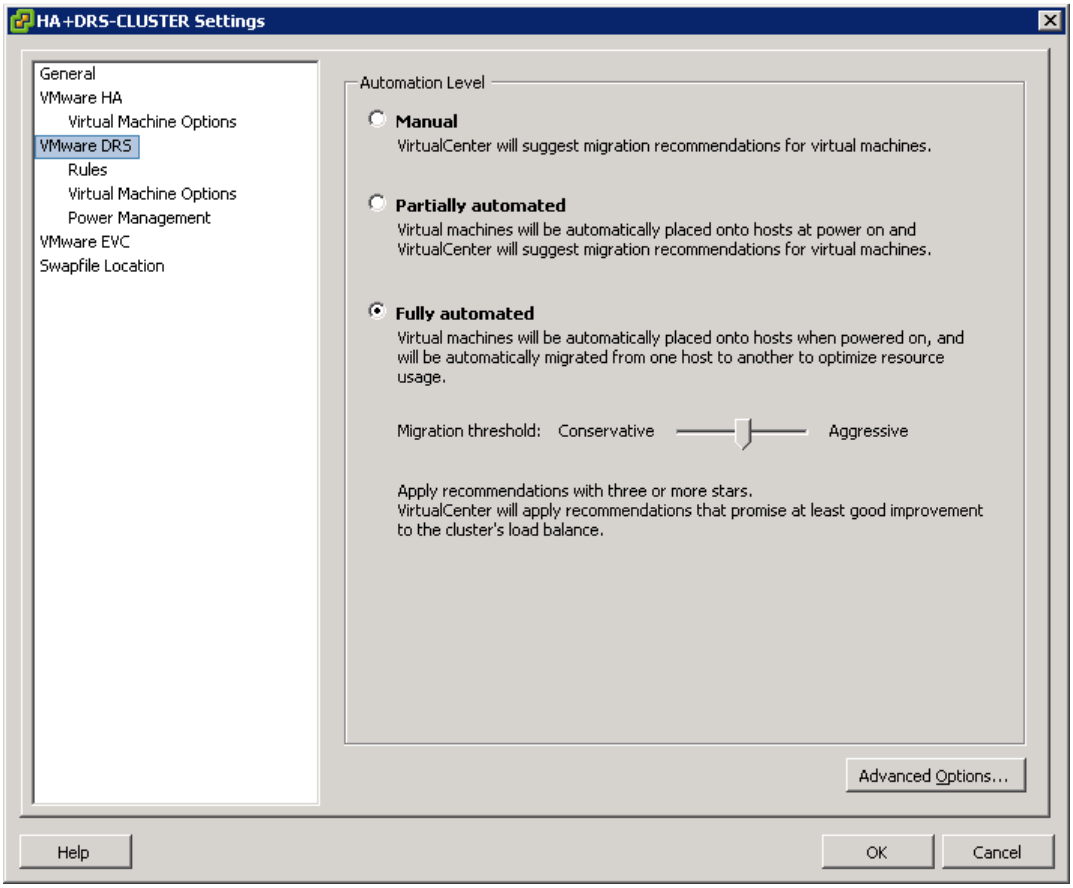
Klusterin asetuksissa määritetään HA ja DRS päälle.

HA asetukset



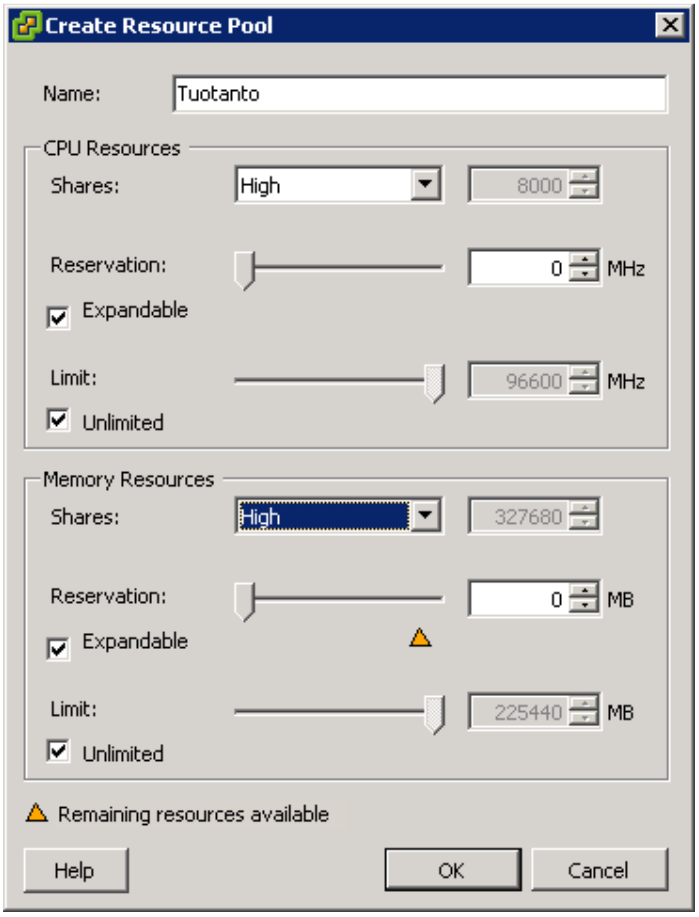
HA:n asetuksissa määritetään, kuinka montaa isäntäkonetta klusteri voi hallita ja käyttää virtuaalikoneiden virhetilanteissa.

DRS asetukset



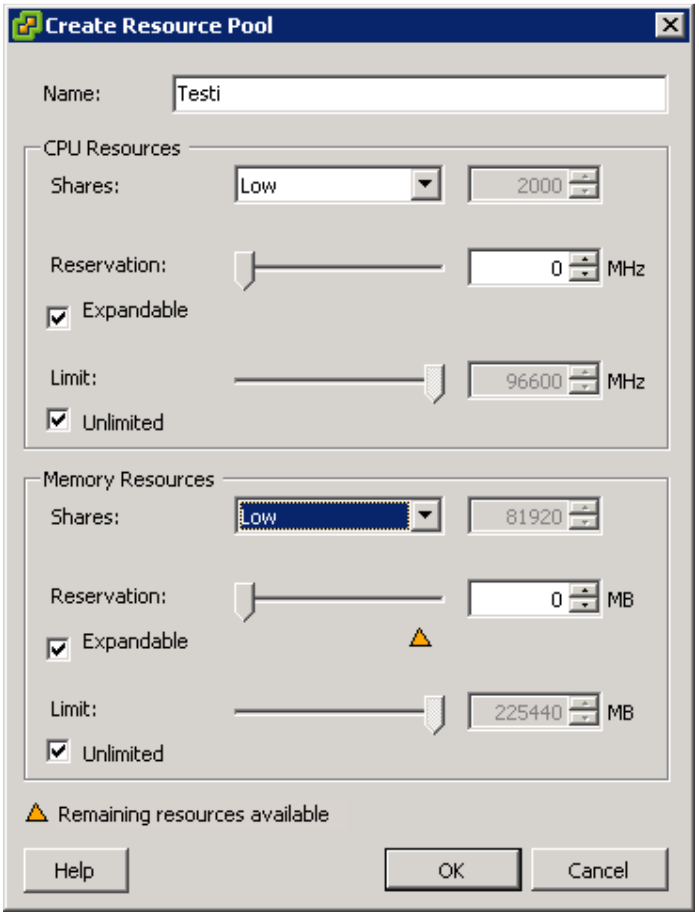
Määritellään DRS täysin automaattiseksi. Kuormanjaon herkkyyden säädöllä, määrätään, kuinka aggressiivisesti virtuaalikoneita siirretään hostilta hostille.

Resurssialtaan luonti



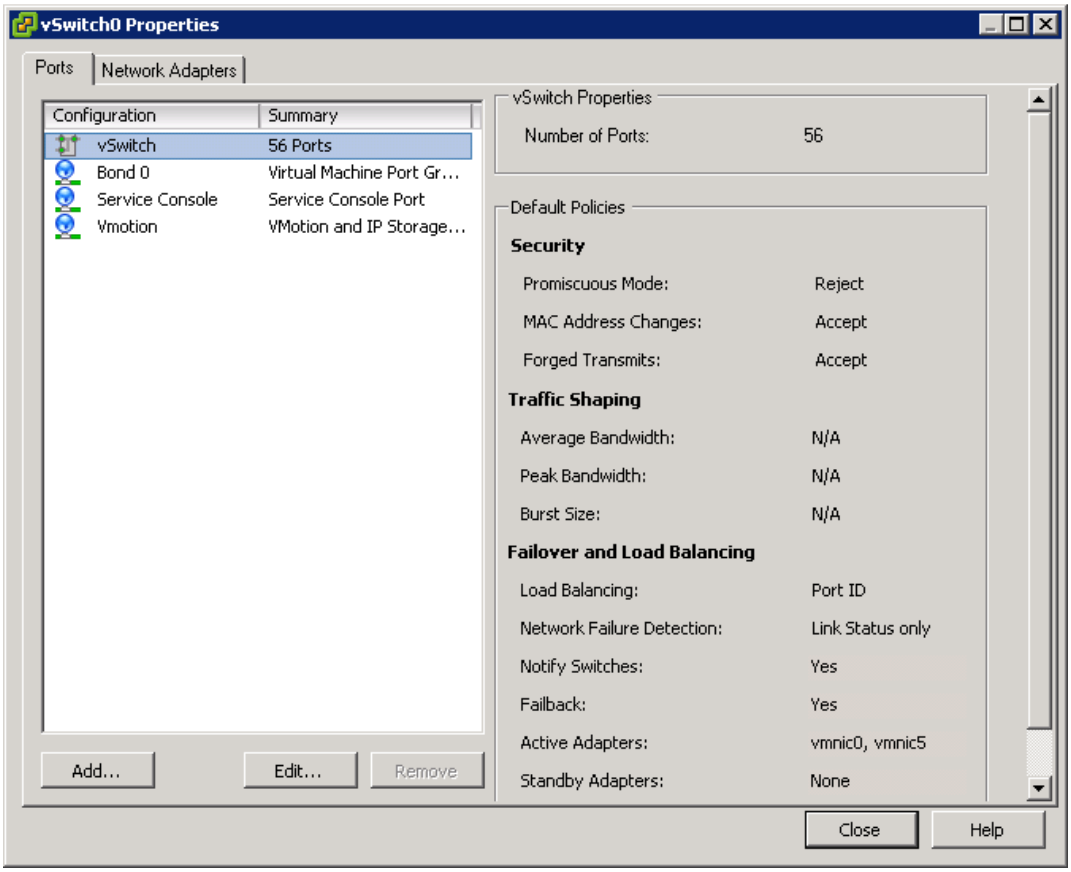
Luodaan tuotanto virtuaalikoneille oma resurssiallas, jossa määritetään virtuaalikoneille korkein mahdollinen prosessorin käyttöaste ja muistin saanti.

Resurssialtaan muistin jako



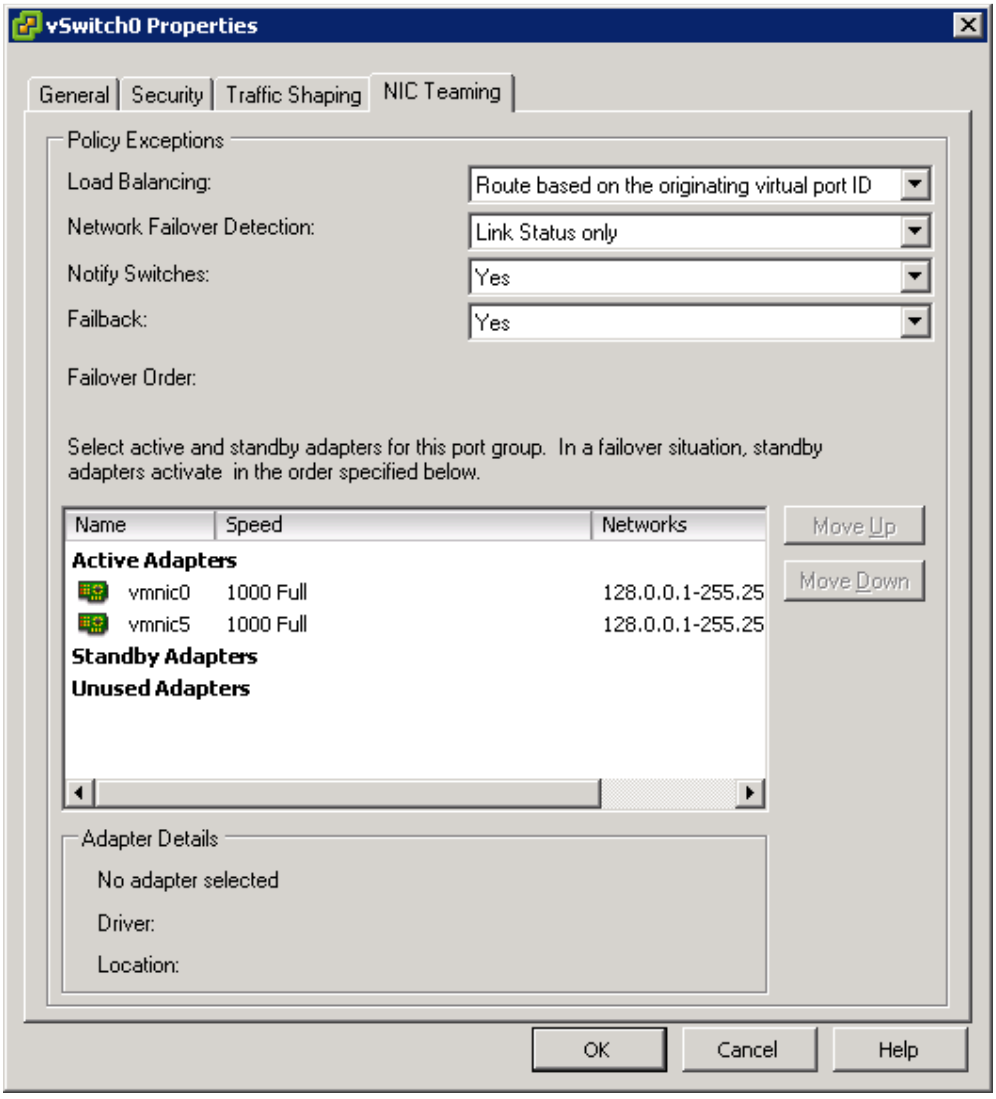
Testikoneille tehdään sama toimenpide kuin tuotantokoneille, jotta ne eivät syö tuotantokoneiden resursseja.

Vituaalikytkimet



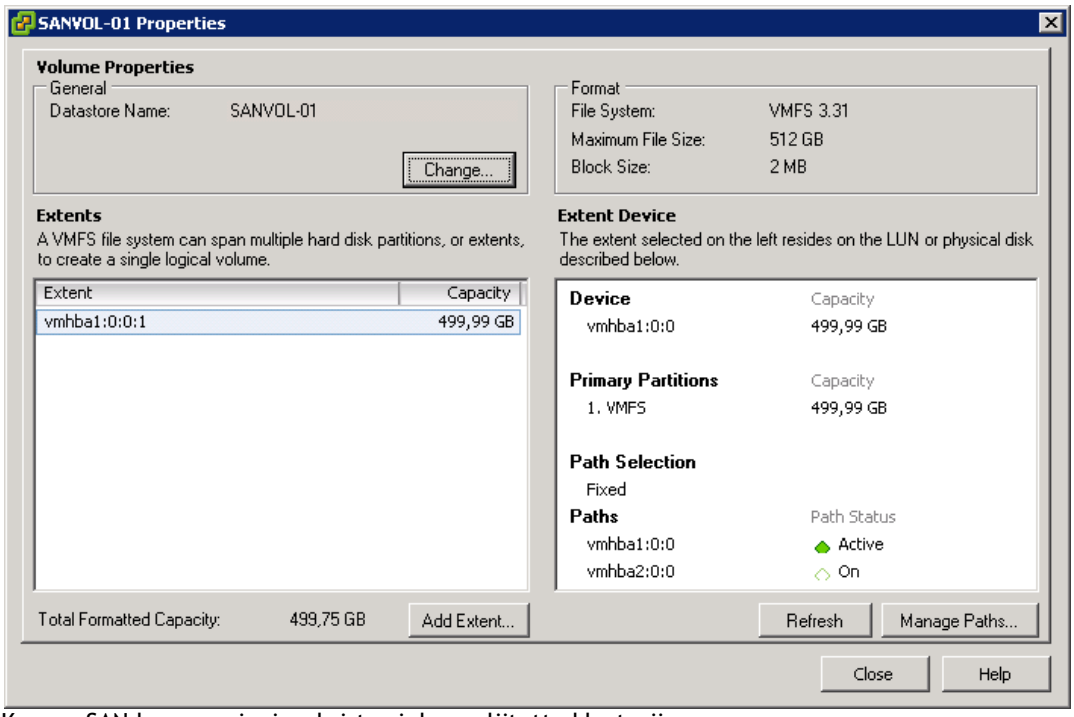
Virtuaalikytkimen asetukset jätetään oletuksille; kuten kuvassa näkyy

SAN-levyt



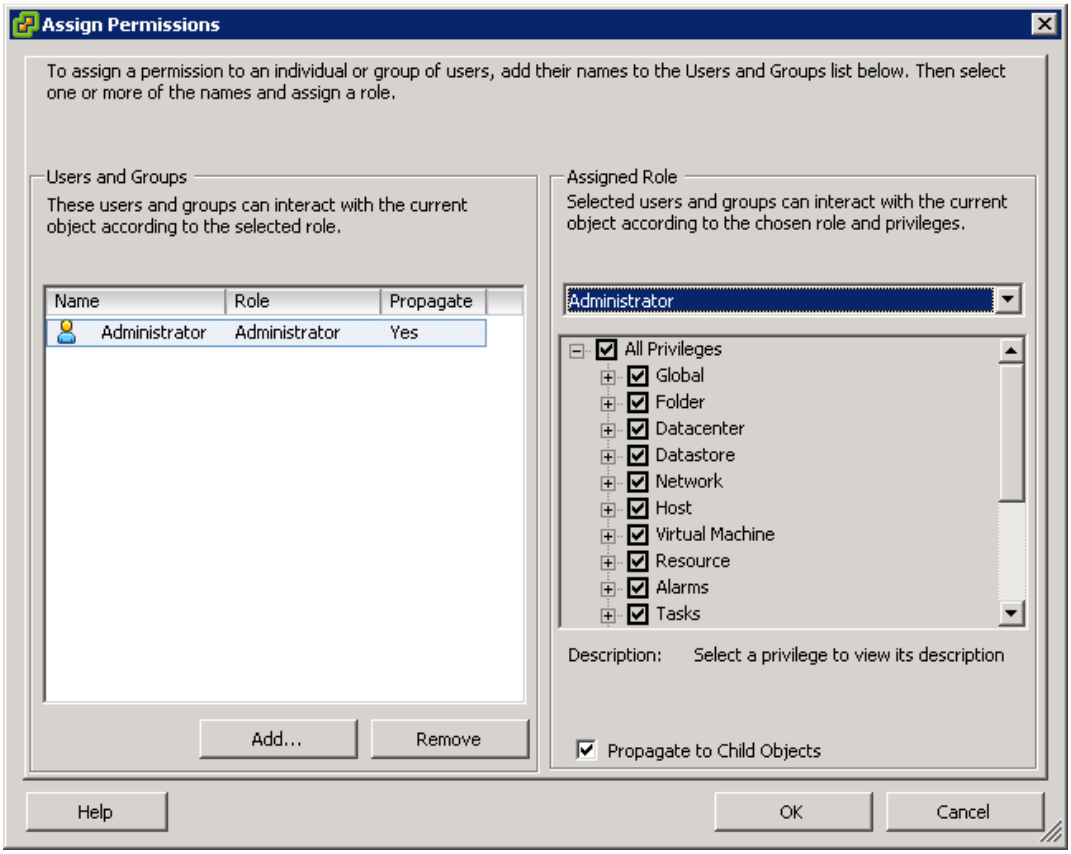
virtuaalikytkin kahdennetaan liittämällä siihen kaksi verkkokorttia

SAN levyn ominaisuudet



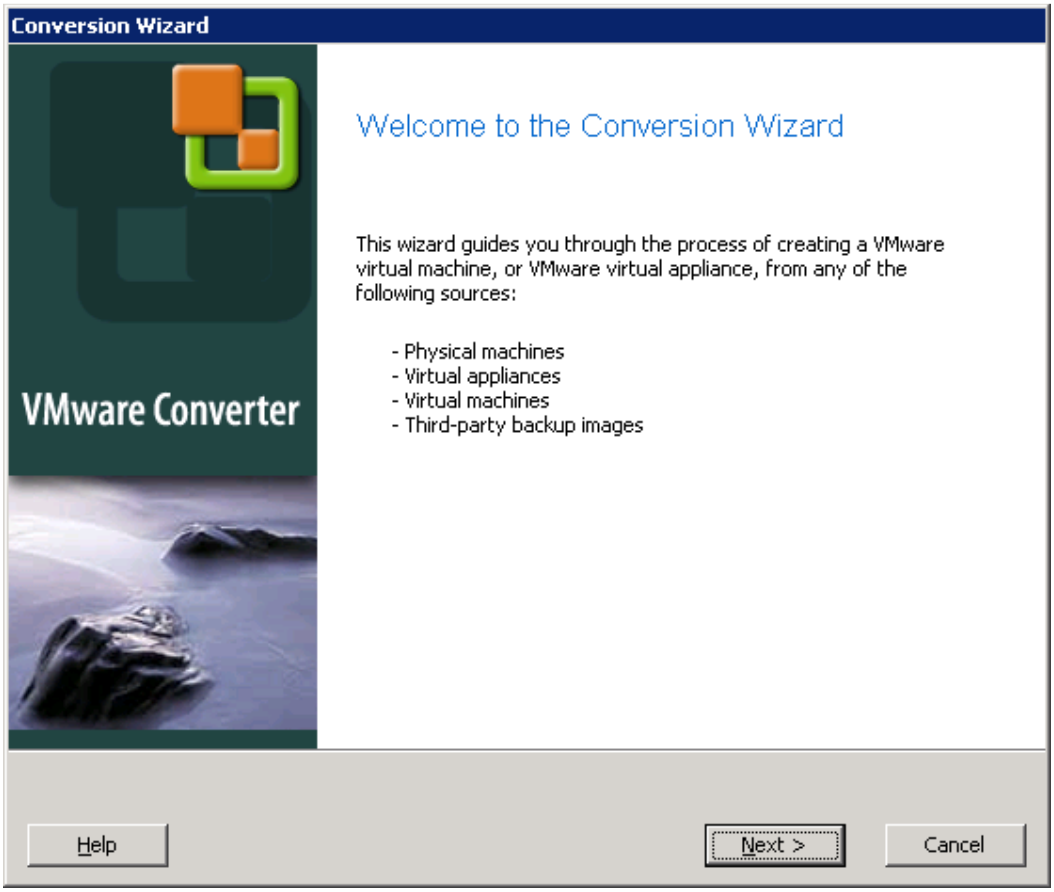
Kuvaus SAN levyn ominaisuuksista, joka on liitetty klusteriin

Käyttöoikeudet



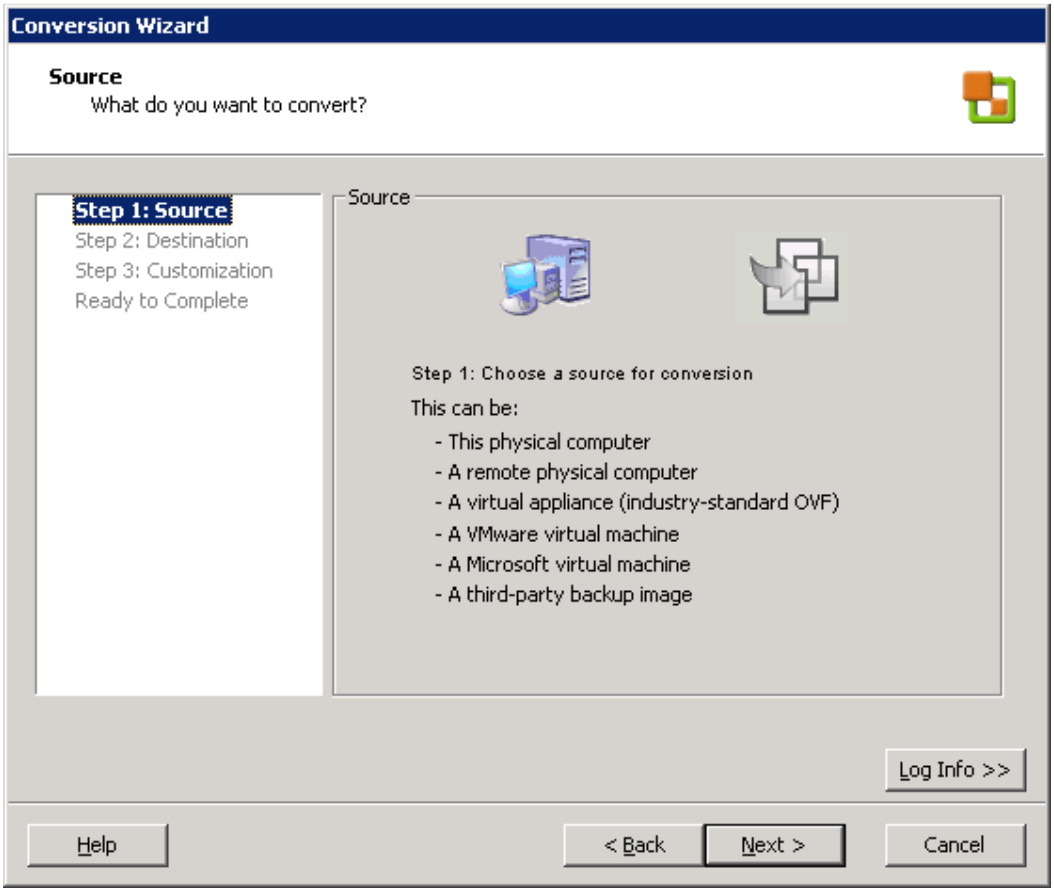
Näkymä käyttöoikeuksien luonnista konfigurointi ikkunassa

Konsolidointi ja muiden kriittisten järjestelmien konvertointi



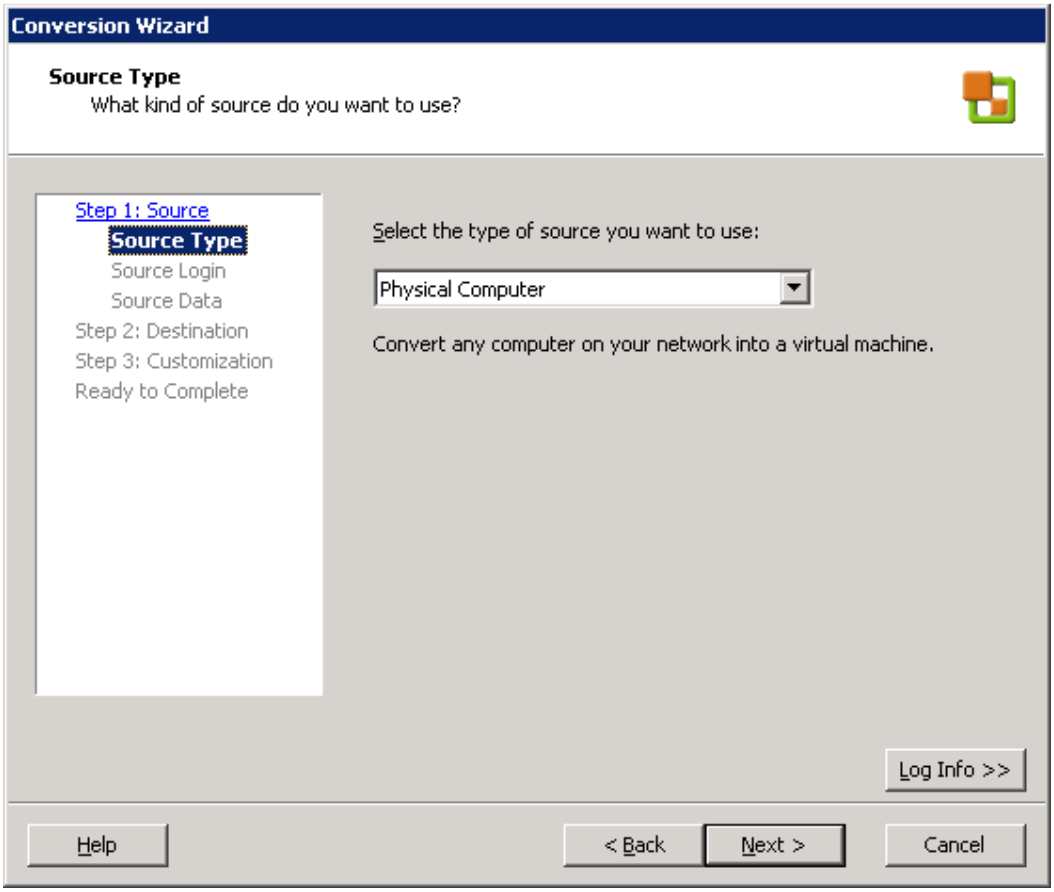
Käynnistetään VMWare converter ohjelma

Konversion käynnistys



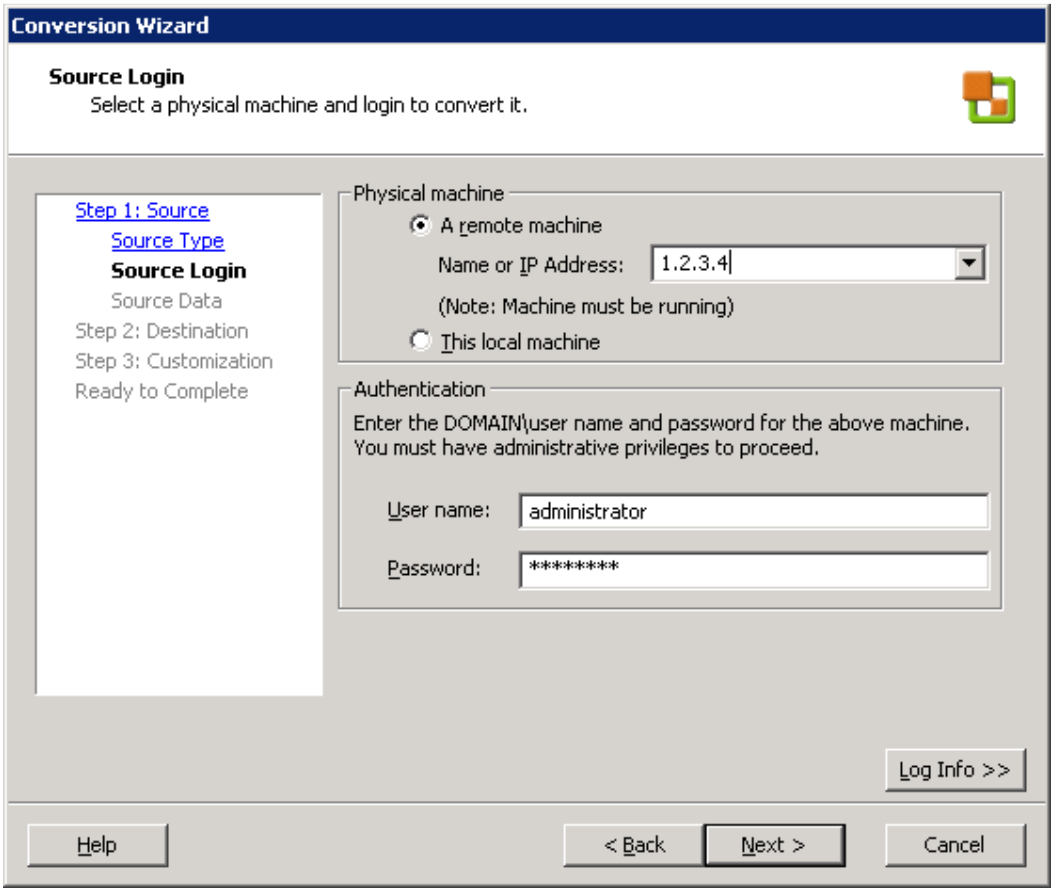
Aloitetaan konversio määrittämällä seuraavaksi lähdekone

Lähdekoneen määritys



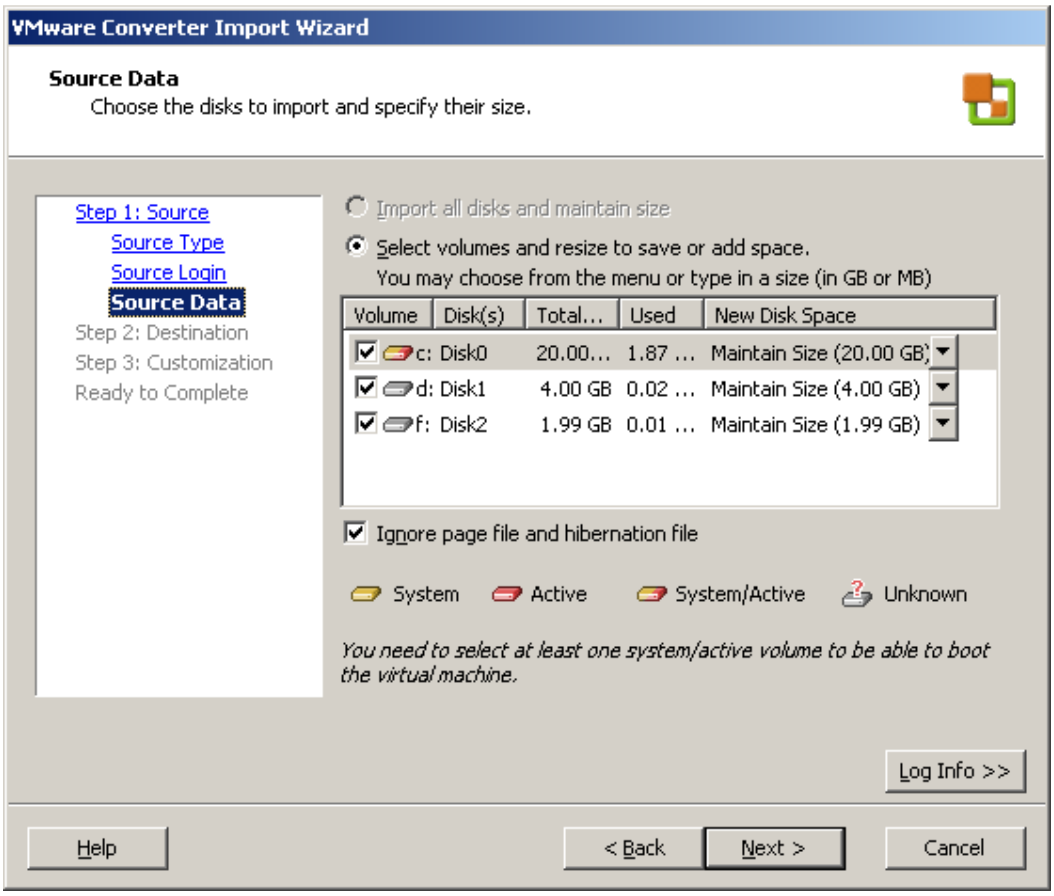
Määritetään, että lähdekone on fyysinen

Lähdekoneen konfiguirointi



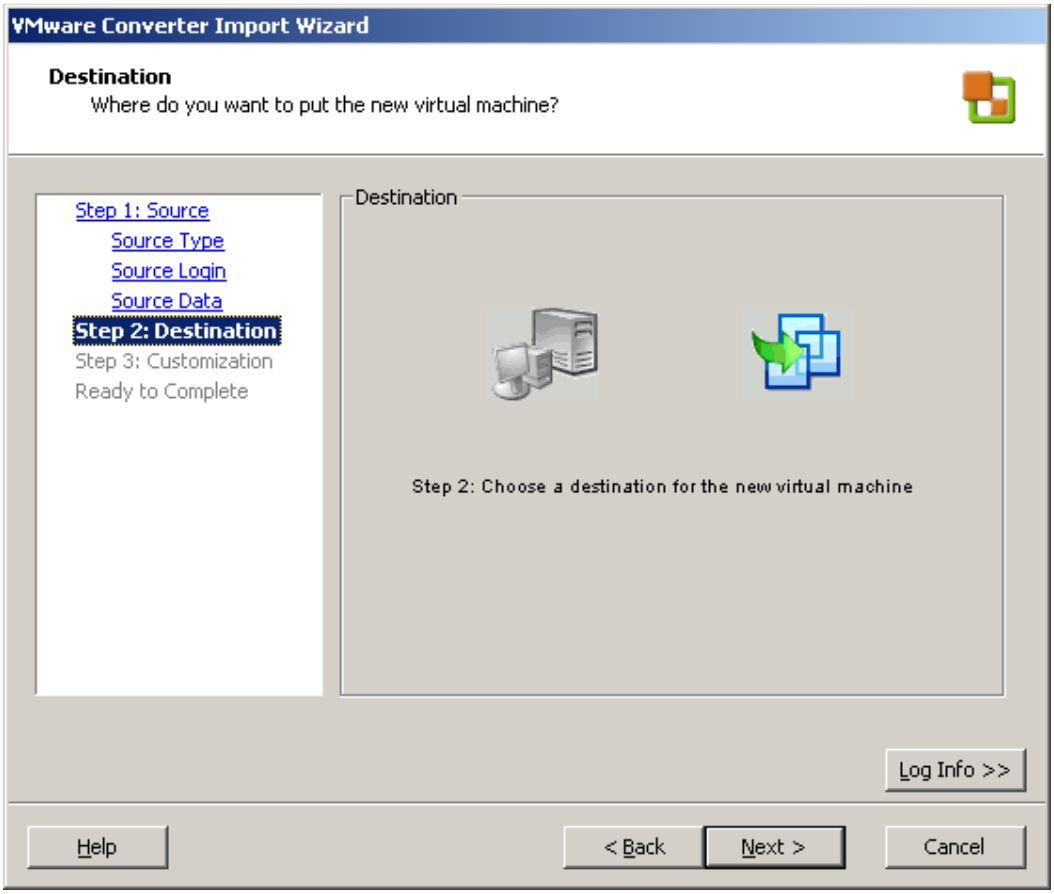
Annetaan lähdekoneen IP-osoite sekä annetaan koneelle kirjautumistunnukset

Fyysisen koneen levyjako



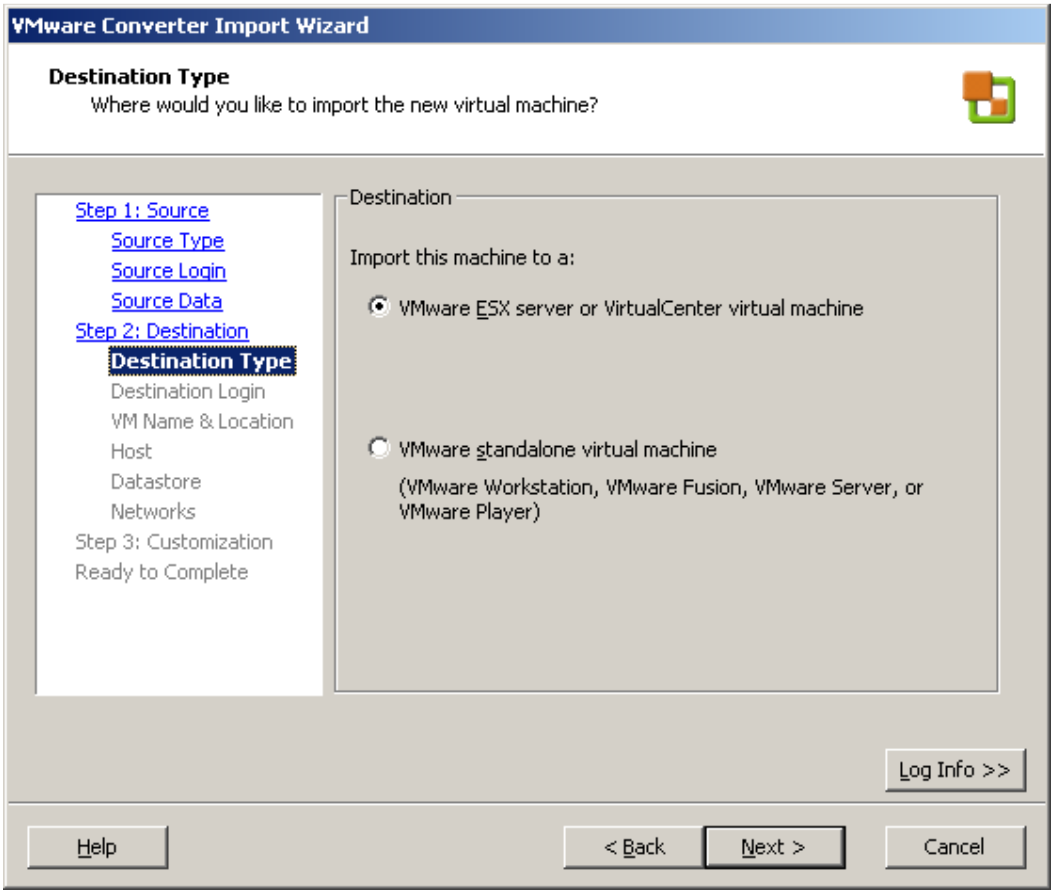
Valitaan fyysisen koneen levyt joiden kokoa voidaan halutessa muuttaa.

Virtuaalisen kohde koneen määrittely



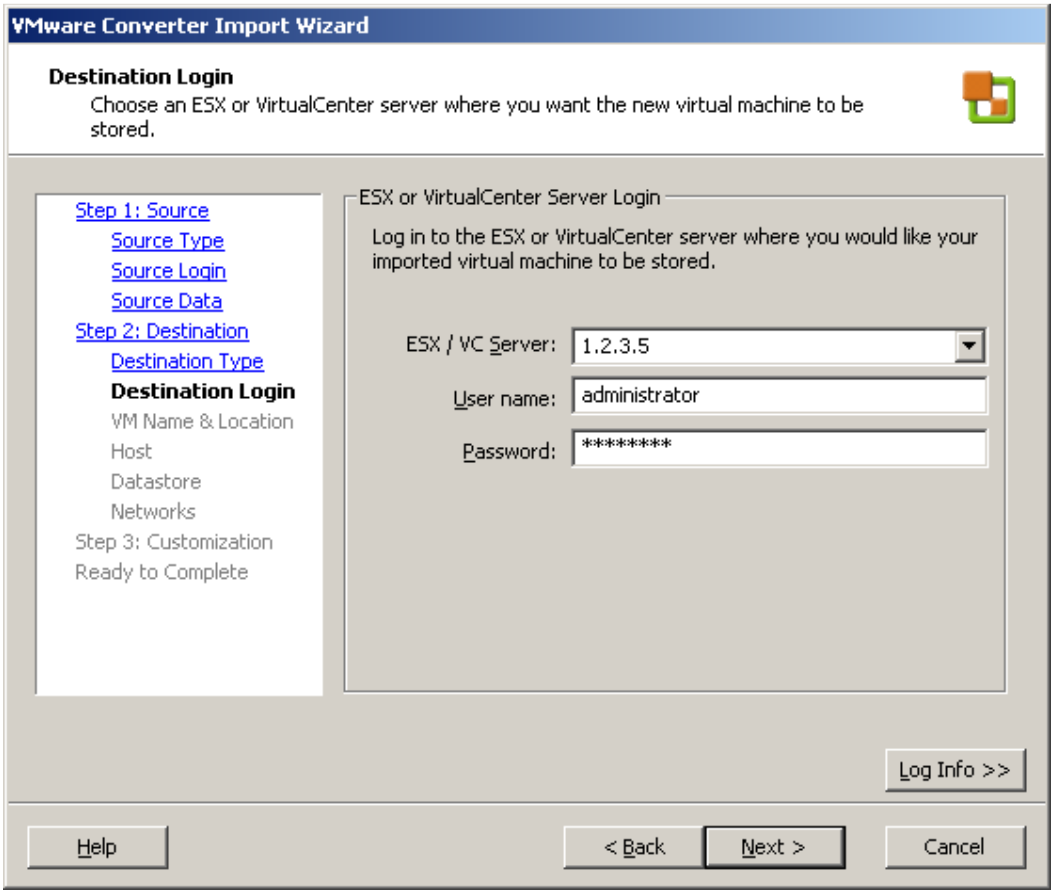
Määritetään kohde kone, mihin fyysinen kone virtualisoidaan.

Kohdekoneen määrittely (ESX tai virtualcenter)



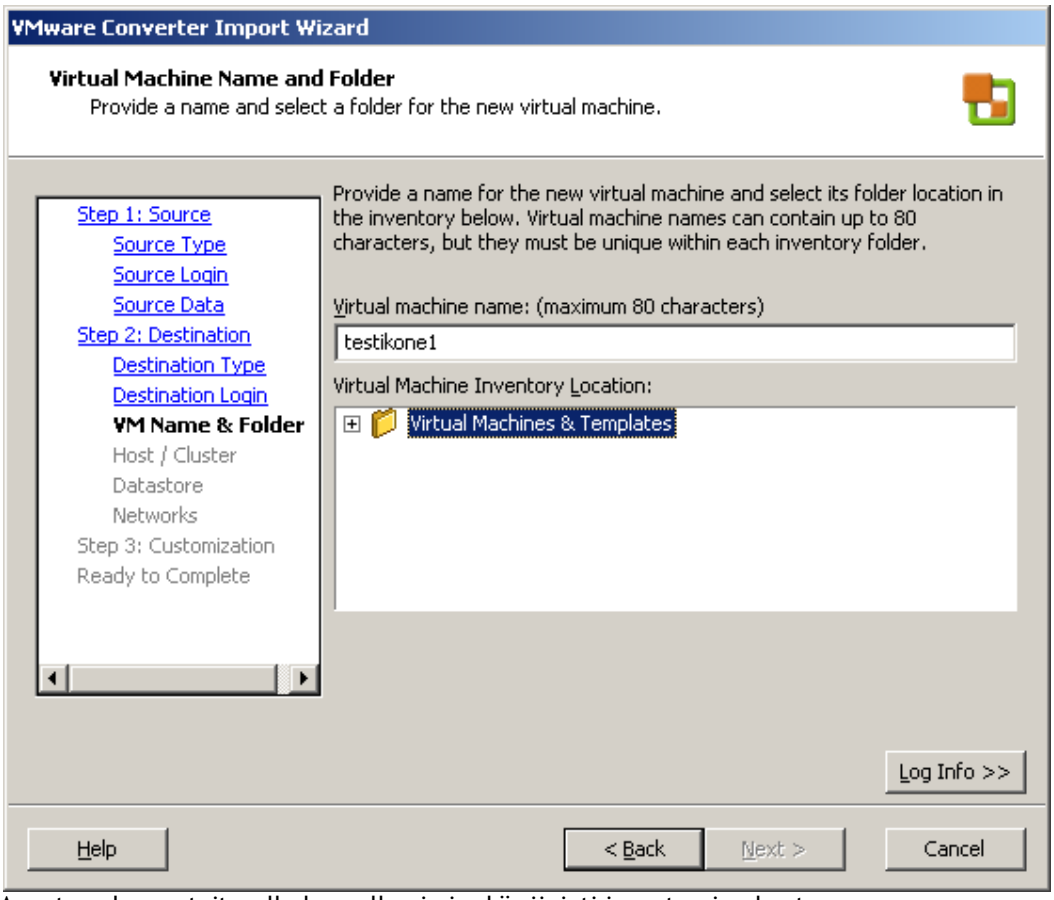
Määritetään kohdekoneeksi virtualcenter tai ESX.

ESX:n tai virtualcenterin ip-osoite sekä sisäänkirjautumistunnuksien luonti



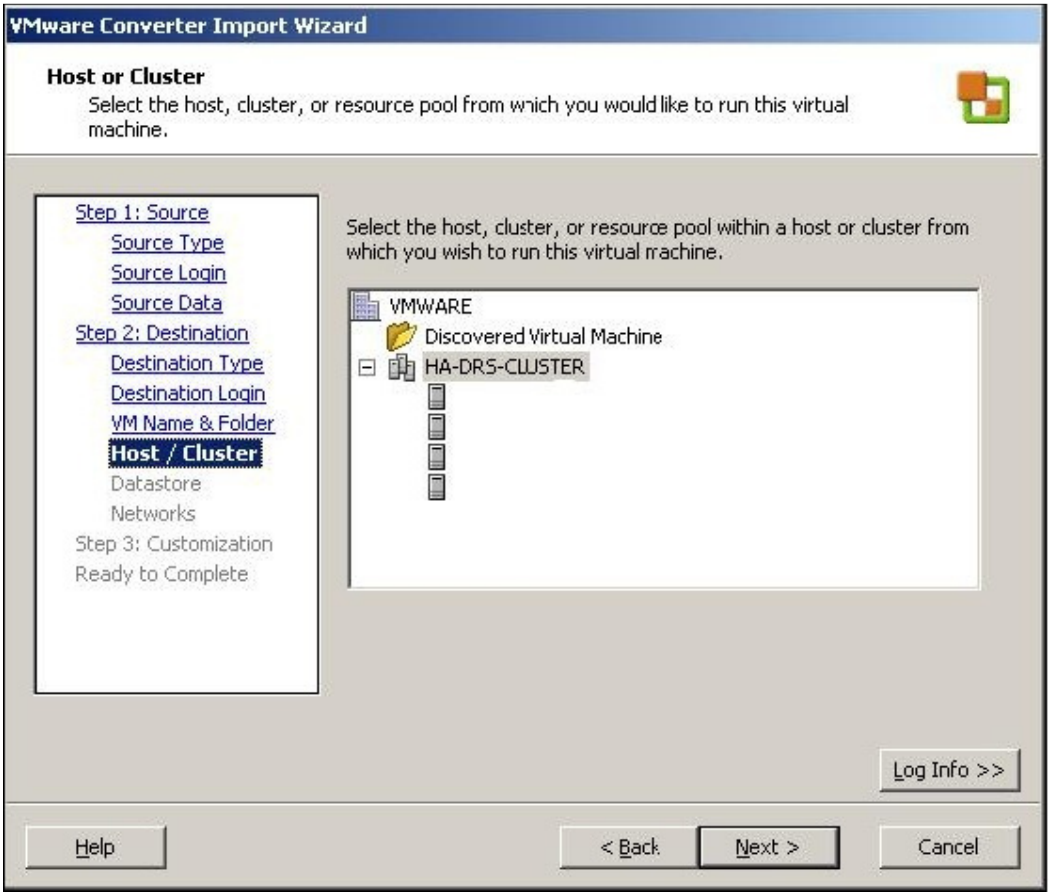
Syötetään ESX:n tai virtualcenterin ip-osoite sekä sisäänkirjautumistunnukset

Konvertoitavan koneen määrittely



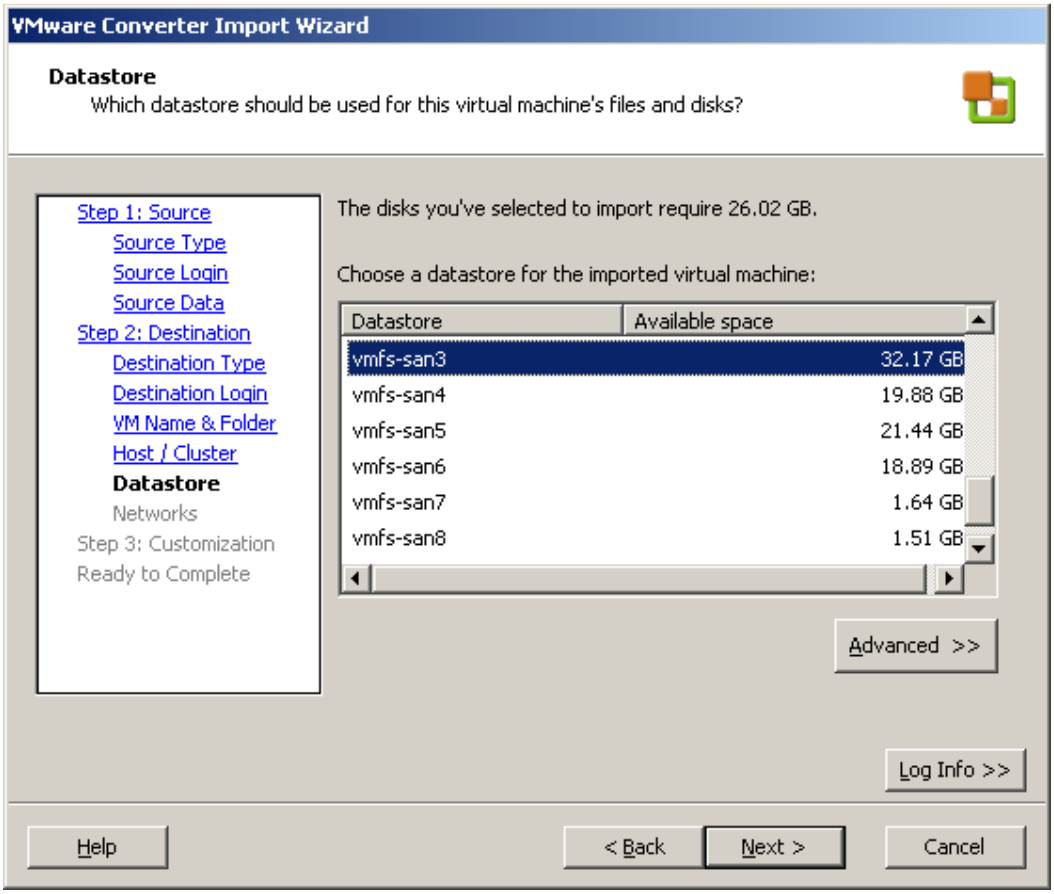
Annetaan konvertoitavalle koneelle nimi sekä sijainti inventaariorakenteessa

ESX koneen valitseminen inventaario rakenteessa



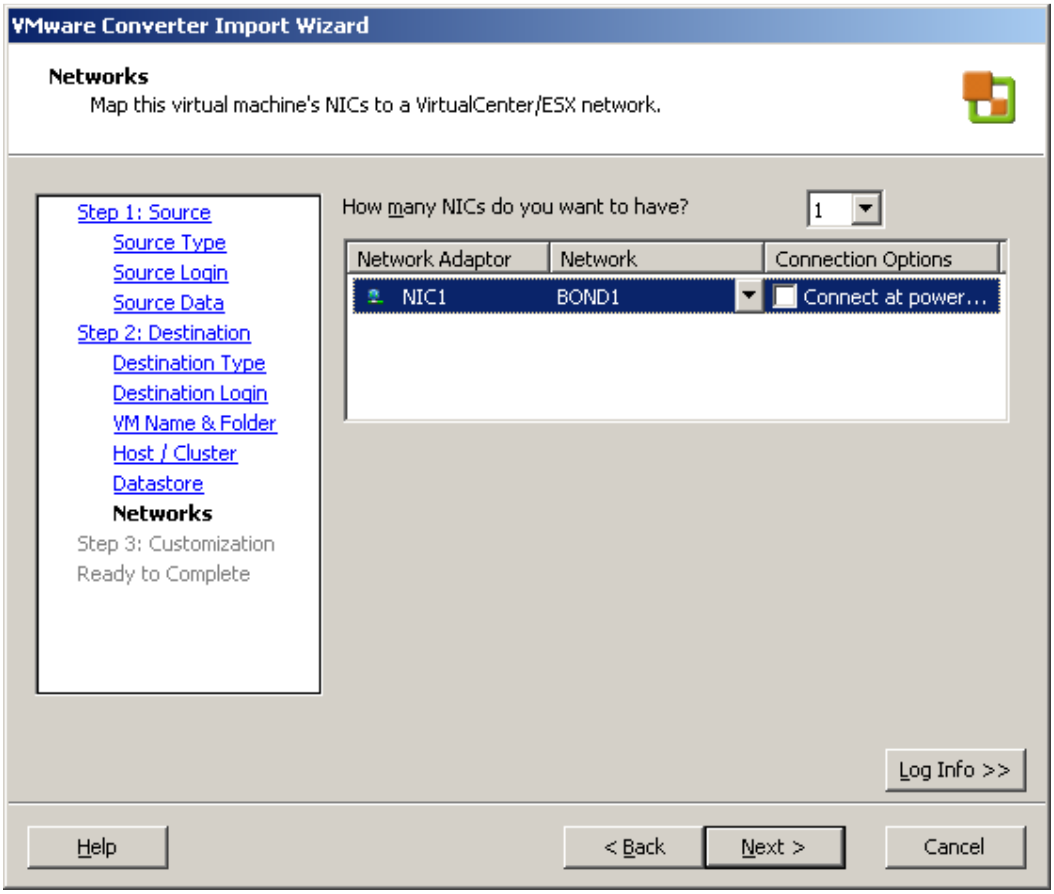
Valitaan ESX kone.

Kuitulevyn valitseminen



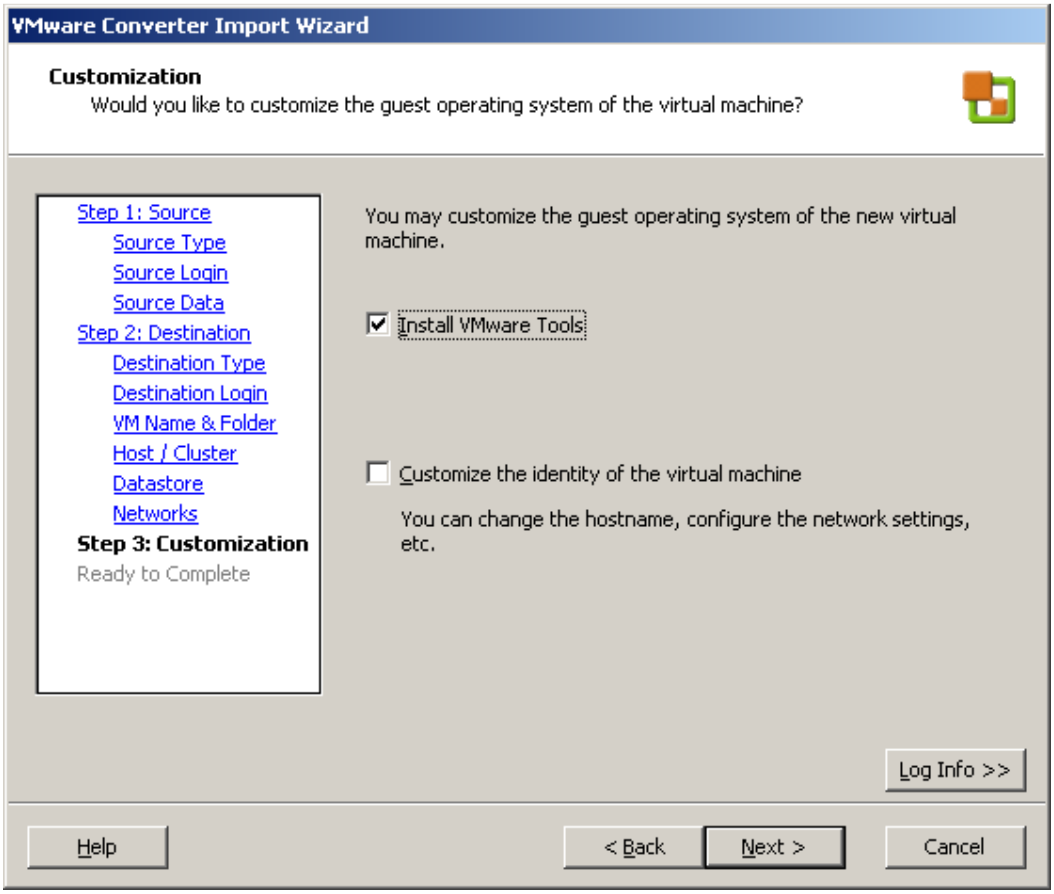
Valitaan kuitulevy, johon virtuaalikoneen tiedostot tallentuvat

Verkkoasetuksen määrittelyt (verkkokortit ja VLAN)



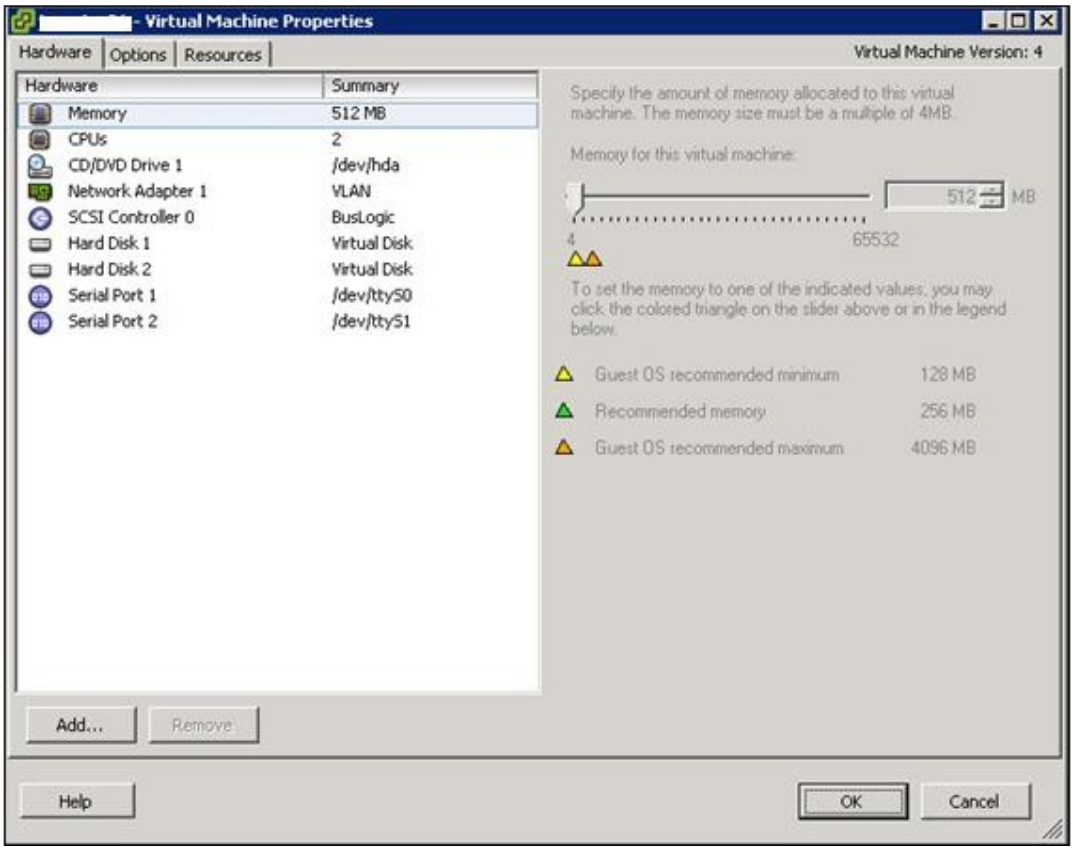
Määritellään kuinka monta verkkokorttia virtuaalikoneelle tulee, sekä valitaan virtuaalikoneelle verkko (VLAN)c

VMware tools paketin asennus



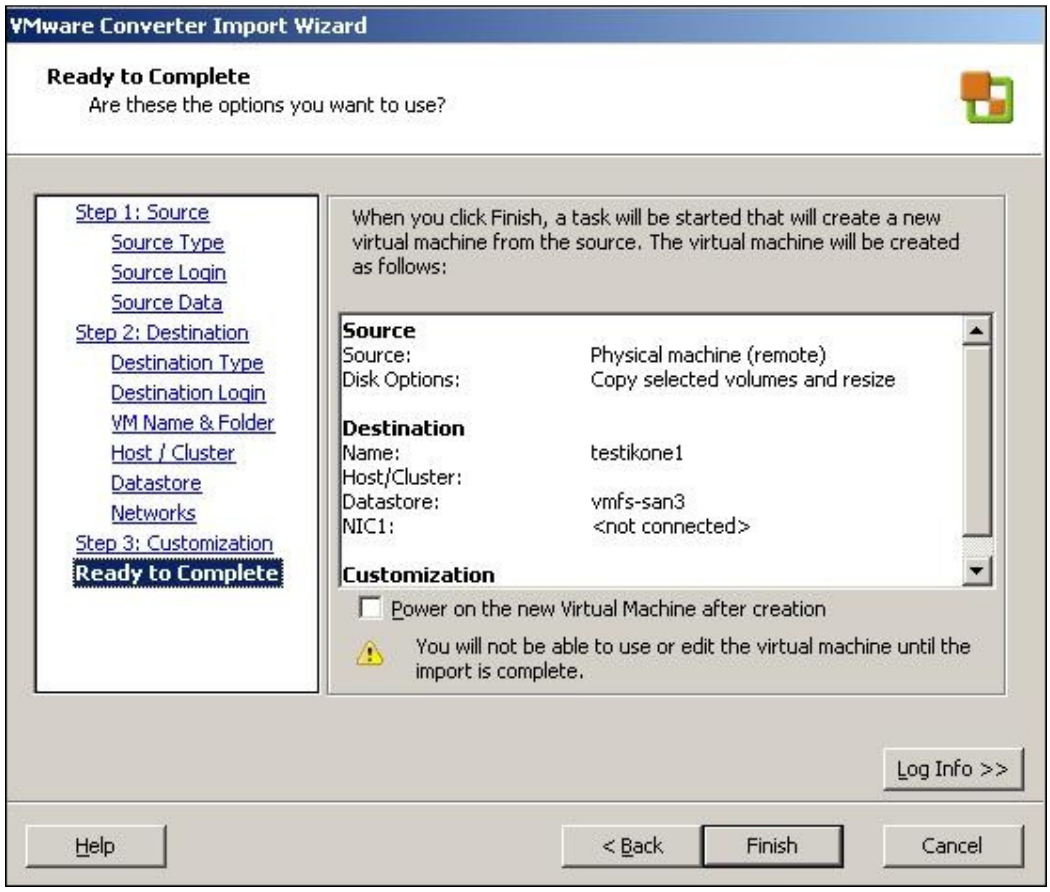
Asennetaan VMware tools -paketti palvelimen virtualisoinnin jälkeen

Virtuaalikoneen ominaisuudet



Virtuaalikoneen ominaisuudet

Yhteenveto suorittavasta konvertoinnista



Yhteenveto suoritettavasta toiminteesta.

YLEISESTI KÄYTETTYJÄ LINUX KOMENTOJA ESX KÄYTTÖJÄRJESTELMÄSSÄ

NTP konfiguirointi löytyy yleensä hakemistosta	/etc/ntp.conf sekä /etc/ntp/step-stickers
NTP:n käynnistäminen	’service ntpd start’
NTPn yhteyden muodostumisen seuranta	’ntpq –p ’
NTP palvelun käynnistyminen uudelleenkäynnistyksen yhteydessä	’chkconfig ntpd on’
BIOSin kellon ajastaminen	’hwclock –systohc utp’
palomuurin konfiguraatio kysely	’esxcfg-firewall –q’
palomuurin liikenteen suodatus päälle ESX:ssä	’esxcfg-firewall --blockIncoming’ & ’esxcfg-firewall -- blockOutgoing
ESX:n resurssien hallinta näkymä virtuaalikoneista	’esxtop’

PALAUTUSSUNNITELMA

TAPAHTUMAN KUVAUS	AIKA	KOMMENTIT
1.testin aloitus	9:25	
2. kriittinen ogelma havaittu	9:30	käyttäjät ja adminit informoit
3. rollback ohjeistus	9:35	ohjeitus läpikäyty, aloitetaan
4. testiympäristön kanta suljetaan	9:55	
5. kannan rollback aloitettu	10:05	kannan imagessa pitkä viive, tilanne kuitenkin etenee
6.kannan kopionti valmis	10:30	kannan kopioinnista lokit talteen
7. rollback valmis	10:45	Rollbackista eventit & lokit ylös
8. sisäänkirjaus TimeConiin	10:55	sisäänkirjaus onnistuu, järjestelmä Näyttää toimivan
9. testataan ovet ja reitittemet	11:05	reitittimet ja ovet toimivat
10. testaus OK	11:10	
11.käyttäjille ilmoitus onnistuneesta Recoverystä	11:20	käyttäjät infottu
12. VMwaren ja timeconin rajapinta Toiminnassa	11:30	rajapinnassa ei havaittu virheitä
13. testi loppuu	11:35	

HUOMIOITAVAA	KOMMENTIT	VASTUUT
-kaikki kontaktilistan käyttäjät ajanatasalla, tämä tarkistettu.	X	
-mitä ympäristöa testissä käytettiin -testissä/harjoituksessa	testiympäristö	
käytetetyt palvelimet	XXX	
- Miten varmistutaan että recoveryssä tapahtuneet muutokset ovat ajantasalla ja valideja		
- kuinka testattu että ed. Toimii moitteettomasti	testattu Timeconin ovia. Yhteys reitittimiin toimii	

- Saavutettiinko RTO	kyllä
- tämän hetkisen levyjärjestelmän backup suoritetaan viikottain täysi kopio, (full) päivittäin (inceremental), back up toiminne alkaa 03:30am	aikojen muutoksesta käyttäjien ja organisaation hyväksyntä
- selvitä levyjärjestelmän koko, riittävää tilaa tulee löytyä väh. 2-3GB.	putsaa levyä tarpeeksi, tilaa lisää levytilaa
- poista levyltä puolivuotta vanhat kanta Dumpit	poistettu-->lisää tilaa
-Päivitä testiympäristön kanta uusimpaan Vesioon	Päivitetty.