OULUN SEUDUN
AMMATTIKORKEAKOULU

Jari Pukki

# YRITYKSEN OFFSITE-VARMUUSKOPIOINTI

OULUN SEUDUN
AMMATTIKORKEAKOULU

# YRITYKSEN OFFSITE-VARMUUSKOPIOINTI

Jari Pukki
Opinnäytetyö
Kevät 2012
Tietojenkäsittelyn koulutusohjelma
Oulun seudun ammattikorkeakoulu

**TIIVISTELMÄ**

Oulun seudun ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma

---

Tekijä: Jari Pukki
Opinnäytetyön nimi: Yrityksen offsite-varmuuskopiointi
Työn ohjaajat: Jukka Kaisto, Pekka Halmkrona
Työn valmistumislukukausi ja -vuosi: Kevät 2012 Sivumäärä: 34

---

Tämän opinnäytetyön tarkoituksena oli toteuttaa ratkaisu Oulussa toimivan ICT-alan yrityksen datan offsite-varmuuskopioimiselle. Yrityksen vaatimuksena ratkaisulle oli Linux-tiedostopalvelimelle varmuuskopioidun datan salaus ja siirto turvalliseen sijaintiin. Lisäksi ratkaisu tuli toteuttaa riittävän kustannustehokkaasti, joten ensisijaisesti päädyttiin tutkimaan avoimen lähdekoodin ohjelmistojen tuomia mahdollisuuksia.

Teoriapohja toteutukselle on hankittu pääasiassa varmuuskopiointiratkaisuja käsittelevistä internetlähteistä. Lähteiden ja kohdeyrityksen avulla toiminnallisen opinnäytetyön tuloksena syntyi ratkaisu off-site varmuuskopioinnille. Offsite–varmuuskopioinnin tarkoituksena on suojata yrityksen tallennettua tietoa katastrofitilanteen sattuessa. Varmuuskopioidut tiedot säilytetään perinteisen varmuuskopion lisäksi myös fyysisesti eri paikassa kuin alkuperäiset tiedostot. Esimerkiksi tulipalon, vesivahingon tai vakavan laitevian sattuessa varmuuskopioidut tiedot säilyvät turvattuna eri osoitteessa. Tiedot ovat varmassa tallessa ja voidaan palauttaa takaisin yrityksen käyttöön katastrofitilanteen jälkeen.

Datan salaus on toteutettu rsyncrypto-ohjelmalla. Palvelinhotellipalveluja tuottavan yrityksen tiloista on vuokrattu paikka tiedostopalvelimelle, johon data synkronoidaan rsync-ohjelmaa käyttäen. Opinnäytetyön tuloksena on toimiva ja helppokäyttöinen lisävarmennusjärjestelmä, joka pystyttiin toteuttamaan yrityksen olemassaolevalla laitteistolla. Toteutus on tallennuskapasiteetiltaan vielä laajennettavissa huomattavasti, joten ratkaisun toimii sellaisenaan yrityksen käytössä vielä pitkään.

---

Asiasanat: varmuuskopiointi, Linux, rsync, rsyncrypto

**ABSTRACT**

Oulu University of Applied Sciences
Degree Programme in Business Information Systems

---

Author: Jari Pukki
Title of thesis: Offsite backup of a case company
Supervisors: Jukka Kaisto, Pekka Halmkrona
Term and year when the thesis was submitted: Spring 2012 Number of pages: 34

---

The aim of this thesis was to prepare a backup solution for an ICT company located in Oulu, Finland. The requirements for the solution were to encrypt and transfer the company's data to a secure location. In addition, the solution needed to be implemented cost-effectively. Therefore, the investigated solutions were primarily open source software.

Theoretical framework consisted of internet sources related to backup solutions. The purpose of offsite backup is to protect company's stored information in the event of a disaster situation. This thesis is functional and the suggested solution was based on the sources, and the case company's assistance. According to the suggestion, the company's backup is now physically stored in two locations. One backup is stored in the company's facilities and the other backup in other location. For example, in case of a fire, water damage or a serious hardware failure, the backup data will be safe in a different address. All the data are in a safe place, and can be returned to the company's use after the situation of a disaster.

The data encryption is implemented with rsyncrypto program. A place for the file server is leased from a company providing server hosting services where the encrypted data are synchronized using rsync program. The solution is an effective and easy to use offsite-backup system. The storage capacity is highly expandable. Therefore, the solution is feasible for the company for a long time.

---

Keywords: backup, Linux, rsync, rsyncrypto

**SISÄLLYS**

# 1 JOHDANTO

Opinnäytetyön toimeksiantaja on oululainen ICT-alan yritys. Toiminnallisen opinnäytetyön tavoitteena on tuottaa yritykselle toimiva ja vakaa varmuuskopiointiratkaisu kustannustehokkaasti. Aiheen idea syntyi kohdeyrityksessä suoritetun opintoihini kuuluvan ammattiharjoittelun aikana.

Uusitun palvelinjärjestelmän myötä edellinen offsite-varmuuskopiointiratkaisu ei osoittautunut riittäväksi yrityksen tarpeisiin, joten lähdettiin kehittämään joustavaa ja helposti laajennettavaa, sekä helppokäyttöistä varmuuskopiointiratkaisua. Toteutus tuli lisäksi toteuttaa kustannustehokkaasti ja käyttää hyväksi yrityksen jo olemassaolevaa laitteistoa.

Työn vaatimuksena on varmentaa yrityksen Windows-palvelinten tiedot fyysisesti kahteen eri paikkaan. Varmennettavat palvelimet ovat Windows 2008 R2 -alustalla toimivia fyysisiä ja virtuaalipalvelimia. Työssä keskitytään jo varmennetun datan salaukseen Linux-palvelimella ja salatun datan siirtämiseen yrityksen ulkopuolella sijaitsevaan palvelinhotelliin.

Ratkaisu toteutetaan avoimen lähdekoodin sovelluksilla, koska vastaavan järjestelmän tuottaminen kaupallisella ratkaisulla aiheuttaisi lisäkustannuksia. Ratkaisuun käytetään kahta Linux-palvelinta, joista toinen sijaitsee yrityksen toimitilojen ulkopuolella ja toinen yrityksen sisäverkossa. Linux-palvelinten levyjärjestelmät on toteutettu vikasietoisella RAID-tekniikalla. Windows-palvelinten data kopioidaan Linux-tiedostopalvelimelle, jossa se salataan ja lähetetään edelleen yrityksen tilojen ulkopuolella sijaitsevalle palvelimelle.

Offsite-varmistus on lisäksi salattava riittävän tehokkaasti. Katastrofin, kuten tulipalon tai vesivahingon sattuessa, jossa fyysiset palvelimet rikkoontuvat käyttökelvottomaksi, kaikki yrityksen tietoverkkoon tallennettu tieto on turvassa toisessa paikassa ja voidaan helposti palauttaa uuteen järjestelmään.

# 2 RAID-LEVYJÄRJESTELMÄT

Lyhenne RAID (Reduntant Array of Independent Disks) tarkoittaa vikasietoista levyjärjestelmää. Tekniikalla haetaan vikasietoisuutta ja/tai nopeutta kasvattamalla kiintolevyjen määrää ja yhdistämällä ne yhdeksi loogiseksi levyksi. Käytössä voi olla useita kiintolevyjä, jotka käyttöjärjestelmä käsittelee yhtenä levynä. RAID tallettaa ja jakaa dataa tehokkaasti kaikkien levyjen kesken sen sijaan, että käyttöjärjestelmä käsittelee jokaista yksittäistä levyä omana levyjärjestelmänään. RAID-tasoja on useita ja seuraavassa on esitelty muutama yleisimmin käytetyistä tekniikoista. (Buffington 2010, s. 45). Taulukossa 1 esitetään muutaman yleisimmin käytetyn RAID-tason ominaisuuksia. RAID-järjestelmiä on useita ja olemassa olevia järjestelmiä voidaan yhdistellä.

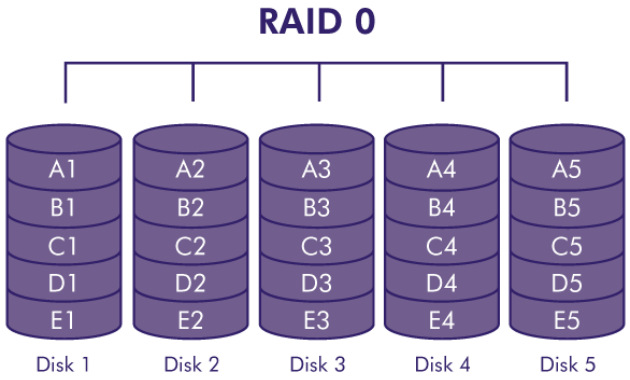
Nykyaikaisissa palvelimissa rikkoutunut levy on vaihdettavissa laitetta sammuttamatta (hot-swap) ja RAID-järjestelmä voidaan rekonstruktoida. Osa levyistä voidaan varata lennossa käyttöön otettavaksi (hot-spare). Levyn rikkoontuessa varalevy aktivoituu automaattisesti käyttöön. (Kozierok 2001, hakupäivä 22.12.2011.)

| **Kategoria** | **RAID-taso** | **Kapasiteetti** |
|---|---|---|
| Lomitus | 0 | 100 % |
| Peilaus | 1 | 50 % |
| Pariteettiratkaisut | 5 | 80 % |
| | 5 + spare | 60 % |

Taulukko 1. RAID-tallennuskapasiteetti

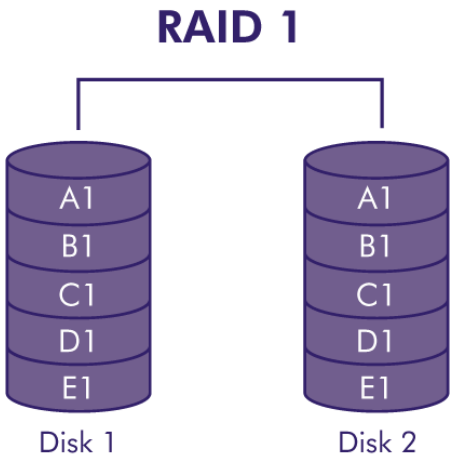
**RAID0** eli lomitus (striping) on tekniikka, jolla data jaetaan tasaisesti kaikille järjestelmän levyille. Monta levyä yhdistetään niin, että yhteenlaskettu tallennuskapasiteetti näkyy yhtenä loogisena tilana. Useamman levyn käyttäminen yhtä aikaa parantaa suorituskykyä. Vikasietoisuutta tämä ratkaisu ei tarjoa, sillä jos yksikin levy järjestelmästä rikkoontuu, menetetään kaikki levypakan data. Kuviossa 1 on esitetty datan jakautuminen tasaisesti usean levyn kesken. Kuvion palkit kuvaavat yksittäisiä levyjä, joiden sisällä datan lomitus on jaettu osiin A1-E5. Nollatasoa

käytetään kohteissa, missä vaaditaan suurta suorituskykyä, kuten kuvankäsittelyssä, äänenkäsittelyssä ja videosovelluksissa. (Buffington 2010, s. 45.)



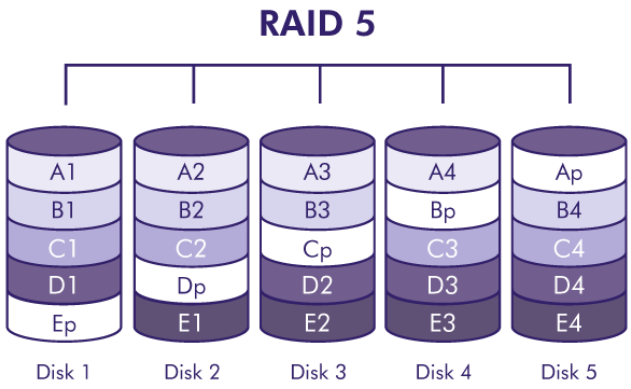

KUVIO 1. RAID 0 (Lacie 2011, hakupäivä 24.11.2011)

**RAID1** eli peilaus (mirroring) on tekniikka, jossa data kahdennetaan levyjen kesken.Toisen levyn hajotessa yhtään dataa ei menetetä. Peilaus tarjoaa parhaan varmistuksen, mutta kustannukset tuplaantuvat samalla. Toinen huono puoli on loogisen aseman koon rajoittuminen suurimman kiintolevyn mukaan. Peilausta käytetään usein palvelinkoneiden käyttöjärjestelmäosioilla (Buffington 2010, s. 46.) Kuvio 2 esittää peilauksen periaatteen.




KUVIO 2. RAID 1 (Lacie 2011, hakupäivä 24.11.2011.)

**RAID5** levypakassa täytyy olla vähintään kolme kiintolevyä. Taso 5 määrittelee datan levittämisen levyille ja datalohkot suojataan XOR-funktiolla lasketun tarkisteen avulla. Tallennuskapasiteettia kuluu tarkisteisiin yhden levyn verran. Jos yksi fyysisistä levyistä vikaantuu, kaikki data on silti edelleen käytettävissä, koska se luetaan pariteetti jaksoista. Vikaantuneen levyn tiedot järjestellään ja rakennetaan uudelleen varalevylle. Kun vikaantunut levy korvataan uudella, korvatusta levystä tulee uusi varalevy. RAID5 tarjoaa hyvän vikasietoisuuden ja suorituskyvyn. RAID5-tasoa käytetään tiedostopalvelinten levyjärjestelmissä. (Buffington 2010, s. 46-47.) Kuviossa 3 on kuvattu datan jakautuminen RAID5–tasoa käyttävällä levyjärjestelmällä.




KUVIO 3. RAID 5 (Lacie 2011, hakupäivä 24.11.2011.)

# 3 VARMUUSKOPIOINTI

Varmuuskopioinnilla tarkoitetaan sitä, että tiedostosta otetaan kopio talteen. Suurin syy varmuuskopioinnin tekemiselle on se, että tuotettua tietoa ei haluta menettää. Syitä tietojen menettämiselle on useita. Onnettomuudet, kuten tulipalo, vesivahinko, laitevika, ohjelmien aiheuttamat virheet sekä ihmisen oma toiminta ovat tyypillisiä syitä tietojen häviämiselle. Virukset ja erityyppiset haittaohjelmat voivat aiheuttaa tiedostojen tuhoutumista. Jos alkuperäinen tiedosto tuhoutuu tai korruptoituu, tiedosto palautetaan takaisin varmuuskopiolta. Yrityksen toiminnan jatkuvuuden kannalta varmuuskopiointi on kriittinen toimenpide. Jos yrityksen keskeiset tiedot menetetään, tietojen uudelleentuottaminen saattaa muodostua uhkaksi koko liiketoiminnalle. Tietovaurion aiheuttamat kustannukset ovat yrityksille aina välittömiä. Menetetty työaika ja aikataulujen venyminen tulevat kalliiksi. (Microsoft Corporation 2010a, hakupäivä 6.10.2011.)

Seuraukset voivat lisäksi olla paljon suurempiakin, kuin pelkästään tietojen häviäminen. Konkreettisin ja pahin seuraus on yrityksen mahdollisesti menettämät asiakkaat. Asiakastietokannan tuhoutuminen ja sitä kautta asiakastietojen menetys voi johtaa asiakaskontaktien katoamiseen lopullisesti. On myös mahdollista, että asiakas on riippuvainen yrityksen tarjoamista tiedoista, ja näiden tietojen menettäminen voi olla kohtalokasta. Tietojen menettämisellä voi olla negatiivinen vaikutus koko yrityksen imagoon ja yrityksen työntekijöiden työmoraaliin. (Preston 2007, s. 8-10.)

Varmuuskopioinnin tarkoituksena on myös tietyn turvallisuustason takaaminen. Tiedostoista on edelleen olemassa varmennettu versio, vaikka alkuperäinen tiedosto ei ole enää saatavissa. Varmuuskopiointi suoritetaan säännöllisin väliajoin ja varmuuskopiointitapoja on monia. Alkuperäisten tietojen varmentaminen on tapahduttava riittävän usein. Tehokkaimmillaan tiedot varmennetaan aina, kun tieto muuttuu. Käytännössä jatkuva varmentaminen ei ole aina mahdollista, joten yrityksen on syytä suunnitella varmuuskopiointiin strategia. Varmuuskopioita tulee tallentaa kahteen arkistoon. Yleinen käytäntö on varmistaa muuttuneet tiedostot yrityksen sisäverkkoon joka päivä ja kaikki muuttunut data kerran viikossa yrityksen ulkopuolelle varmuusarkistoon (offsite). Arkistoissa säilytetään useampia kopiosukupolvia, jolloin palauttamisen onnistumiselle saadaan riittävä varmuus. Yleinen tapa on säilyttää varmuuskopioista ainakin kolme erillistä sukupolvea (SecMeter 2011, hakupäivä 19.12.2011.)

Offsite–varmuuskopioinnilla tarkoitetaan liiketoiminnalle kriittisten tietojen tallentamista yrityksen ulkopuolelle. Offsite–varmuuskopiointi suojaa yritystä katastrofitilanteen sattuessa. Varmuuskopioidut tiedot täytyy myös säilyttää fyysisesti eri paikassa kuin alkuperäiset tiedostot. Esimerkiksi varkaustapauksen, tulipalon, vesivahingon tai vakavan laitevian sattuessa varmuuskopioidut tiedot säilyvät turvattuna eri osoitteessa. Katastrofitilanteessa kriittiset ja tärkeät tiedot ovat varmassa tallessa ja voidaan palauttaa takaisin yrityksen käyttöön (Frisch 2003, s. 713-716.)

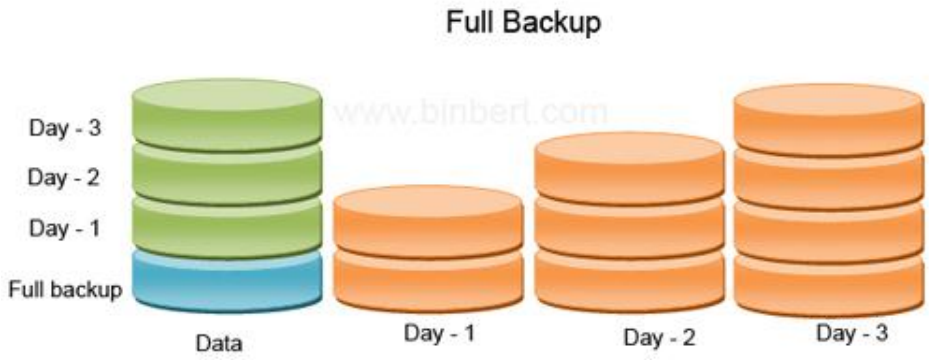
Varmuuskopioinnissa on tietoturvamielessä myös ongelmia. Varmuuskopioimalla tieto samaan paikkaan on samalla luotu keskitetty tietovarasto, josta löytyy kriittistä tietoa yrityksen toiminnasta. Varmuuskopioiden säilyttäminen saattaa myös aiheutua ongelmaksi. Kopio tulee olla nopeasti saatavilla silloin, kun sitä tarvitaan. Usein varmuuskopioita säilytetäänkin samassa tilassa palvelinten ja levyjärjestelmän kanssa. Kopioiden on oltava nopeasti saatavilla laiterikon sattuessa ja usein tämä tarkoittaakin sitä, että varmuuskopiot ovat samassa tilassa kuin koneet ja levypalvelin. Palvelimet, joilla varmuuskopioita säilytetään, tulee suojata erityisen hyvin. Varmuuskopioitu tieto pitää tietysti tarvittaessa pystyä palauttamaan. Kopioidut tiedostot palautetaan takaisin alkuperäiseen paikkaan tai mahdollisesti uuteen järjestelmään. Palauttaminen suoritetaan yleensä samalla ohjelmistolla, jolla varmuuskopio on otettu. Samaa ohjelmaa käyttämällä vältetään virheellisiä palautuksia ja palauttaminen tapahtuu usein myös nopeammin. (Greenberg 2011, hakupäivä 22.12.2011.)

Varmennukset perustuvat usein kolmeen erilaiseen varmuuskopiointitapaan. Tavat on jaettavissa kolmeen luokkaan; täysi varmistus, inkrementaalinen varmistus sekä differentaalinen varmistus. Varmistustapojen ero toisiinsa nähden on se, mitä milloinkin varmennetaan. Varmistusstrategiaa luotaessa kannattaa yrityksen huomioida omat tarpeensa tarkasti. Tietojen palauttamista varmuuskopiolta voidaan helpottaa tekemällä inkrementaalinen tai differentiaalinen varmennus täyden varmistuksen lisäksi. Yleisimpiä varmistusstrategioita onvat sellaiset, joissa täysi varmistus otetaan viikoittain ja inkrementaalinen varmistus päivittäin. (Durham 2002, s. 333.)

### 3.1 Täysi varmistus

Täydessä varmistuksessa kopioidaan kaikki valitut tiedostot ja tiedostot merkitään kopioiduksi. Ensimmäistä varmuuskopiosarjaa luotaessa tehdään usein täysi varmistus. Etuina täydessä varmistuksessa on varmistustekniikan yksinkertaisuus. Kaikki tiedostot ja kansiot on varmennettu

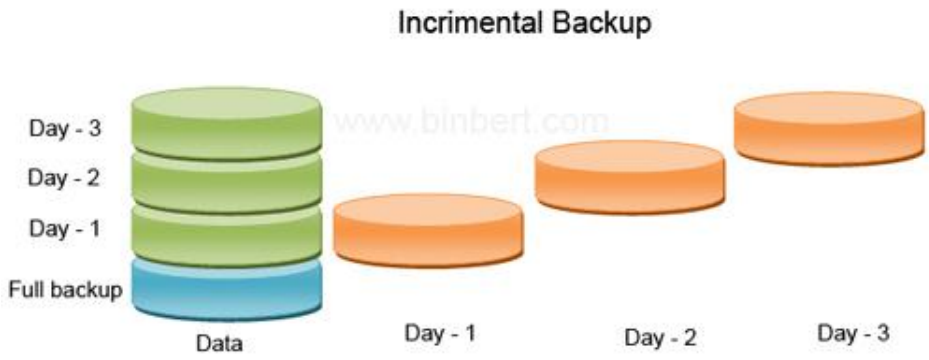
kerralla. Toimenpide vie kuitenkin paljon tallennuskapasiteettia ja aikaa. Täysi varmistus tuleekin suorittaa silloin, kun järjestelmän kuorma on minimissään. Tyypillisesti yritysten järjestelmien käyttö on suurimmillaan päivällä, joten varmistukset on syytä keskittää työajan ulkopuolelle. Täysi varmistus on pohjana myös muille varmistustavoille, koska tiedostosta tarvitaan aina täydellinen kopio, johon muutokset sisällytetään. Tiedostojen palauttamista varten tarvitaan vain viimeisin varmuuskopiosukupolvi (Preston 2007, s. 135.) Kuviossa 4 on esitetty datamäärän kasvu, kun täysi varmistus otetaan järjestelmästä päivittäin.




KUVIO 4. Täysi varmistus (Sebastian 2010, hakupäivä 25.12.2011)

### 3.2 Inkrementaalinen varmistus

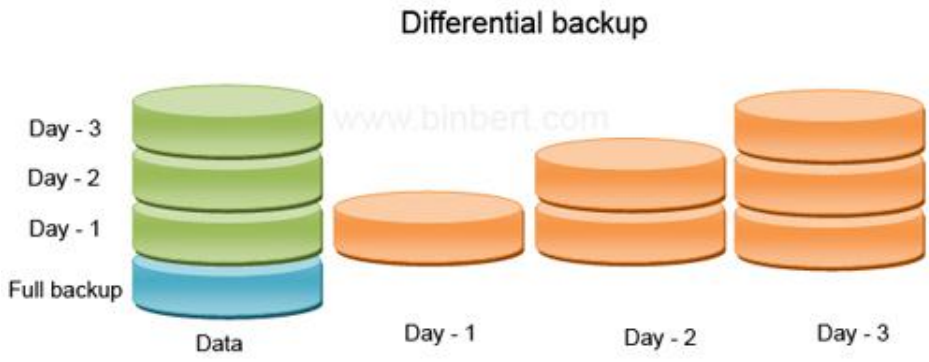
Inkrementaalinen eli lisäävä varmistus on menetelmä, jossa kopioidaan ainoastaan edellisen varmuuskopioinnin jälkeen muuttuneet tiedostot ja merkitään ne kopioiduksi. Täyttä ja inkrementaalista varmennusta käytetään usein rinnakkain, jolloin ensin luodaan täysi varmistus ja sen jälkeen jatketaan varmistusta inkrementaalisesti. Tietojen palauttamiseen tarvitaan viimeisin täysi varmistus sekä kaikki inkrementaaliset varmuuskopiosukupolvet. Teoreettisesti optimaalisessa tilanteessa, missä tallennusmedia ei hajoaisi ja palautuksen tarve olisi hyvin epätodennäköistä, inkrementaalisella varmennuksella voitaisiin minimoida tallennuskapasiteetin tarve. Tällöin edellisen varmistuskerran jälkeen tapahtuneiden muutosten tallentaminen olisi ainoastaan tarpeellista. (Preston 2007, s. 575.) Kuviossa 5 on esitetty inkrementaalisen varmistuksen datamäärän kasvu päivittäin toteutetussa varmennuksessa.




KUVIO 5. Inkrementaalinen varmistus (Sebastian, 2010, hakupäivä 25.12.2011)

### 3.3 Differentiaalinen varmistus

Differentiaalisessa varmistuksessa kopioidaan tiedostot, jotka on luotu tai joita on muutettu viimeisimmän täysvarmistuksen jälkeen. Tiedostoja ei differentiaalisessa varmistuksessa merkitä kopioiduksi. Prosessi vaatii inkrementaalista varmennusta enemmän tallennuskapasiteettia, mutta silti täyttä varmennusta vähemmän. Kun käytetään differentiaalista ja täyttä varmistusta rinnakkain, varmuuskopioiden palautus on nopeampaa kuin pelkästään täydessä varmistuksessa. Palautustilanteessa palautetaan viimeisin täysi varmistus ja viimeisin differentiaalinen varmistus (Preston 2007, s. 576.) Kuviossa 6 esitetään varmistusratkaisun viemä tallennustila, kun täyden varmistuksen jälkeen differentiaalinen varmistus tehdään päivittäin.




KUVIO 6. Differentiaalinen varmistus (Sebastian 2010, hakupäivä 25.12.2011)

# 4 TIEDON SALAUS

Yhtä kauan kuin ihmiset ovat kirjoittaneet ylös informaatiota, on ollut tarve pitää jokin tieto salassa. Tieto on salattu joko piilottamalla materiaali tai muuttamalla sen merkitystä. Tällaisia menetelmiä kutsutaan salaukseksi tai salakirjoitukseksi. (Mohan Krishnamurthy, 2008, s. 250)

Salaaminen tarkoittaa prosessia, jossa informaatiota sekoitetaan niin, että kuka tahansa ei sitä pysty lukemaan. Algoritmi on ohjelma, joilla sekoitetaan ja uudelleen järjestellään alkuperäinen viesti, jota kutsutaan selkokieliseksi tekstiksi. Salausavain on pala dataa, jolla salataan selkokielinen teksti koodikieliseksi. (Krishnamurthy 2008, s. 250.)

Salausmenetelmillä halutaan varmistaa tiedon luottamuksellisuus, eheys ja kiistämättömyys. Tavoitteena on salata tieto niin hyvin, että salauksen murtaminen kohtuullisessa ajassa ja kohtuullisin resurssein ei ole mahdollista. Se kuinka pitkä on kohtuullinen aika ja mikä on kohtuullinen resurssi, riippuu salattavan tiedon tärkeydestä. (Järvinen 2003, s. 77-105.)

Vahvoja salausmenetelmiä ovat sellaiset, joiden murtaminen nykypäivän hyökkääjän saatavissa olevilla laskentaresursseilla ei ole mahdollista. Hyvin toteutetuissa menetelmissä salauksen purku on mahdollista vain käymällä läpi kaikki mahdolliset avaimet ja kokeilemalla niitä kaikkia salauksen purkamiseen. Nykypäivän salaukset pohjautuvat joko bittien sekoittamiseen tietokoneella (symmetrinen salaus) tai matemaattiseen laskentaan (asymmetrinen salaus). (Mohan Krishnamurthy 2008, s. 250.)

## 4.1 Symmetrinen salaus

Symmetrinen salaus on vanhin ja tunnetuin salaustekniikka. Salausavainta, joka voi olla numero, sana tai vain jono satunnaisia kirjaimia, sovelletaan viestin tekstin sisällön muuttamiseksi tietyllä tavalla. Menetelmä voi olla niinkin yksinkertainen kuin siirtämällä jokaisen kirjaimen paikkaa aakkosissa. Niin kauan kuin sekä lähettäjä että vastaanottaja tietävät salaisen avaimen, he voivat salata ja purkaa kaikki viestit, jotka käyttävät tätä avainta. (Järvinen 2003, s.77.)

Symmetriset salaimet ovat jaettavissa kahteen ryhmään: lohko- ja jonosalaimet. Lohkosalauksessa viesti salataan lohko kerrallaan. Selväkielistä tekstiä käsitellään lohko

kerrallaan ja lohkot salataan aina samalla avaimella. Lohkon koot ovat tyypillisesti 64 tai 128 bittiä, koska siten ne on helppo koodata tehokkaiksi tietokoneohjelmiksi. Jonosalaus käsittelee jokaisen viestin bitin tai merkin kerrallaan. Salausavain vaihtuu jokaisen yksittäisen salausoperaation jälkeen. Varsinainen salaus tuotetaan yhdistämällä sen hetkinen avain ja selväkielinen teksti XOR-operaatiolla salattuun muotoon. (Kartalopoulos 2009, s.55-56.)

**DES**

Data Encryption Standard (DES) on laajalti käytetty metodi datan salaukseen käyttämällä yksityistä avainta, joka oli vaikea murtaa. Mahdollisia käytettäviä avaimia on yli 72 kvadriljoonaa (72,000,000,000,000,000). Jokaista salattavaa viestiä varten avain valitaan sattumalta. Kuten muutkin yksityisen avaimen salausmenetelmät, lähettäjä ja vastaanottaja täytyy tietää ja käyttää samaa yksityistä avainta. (Biasci 24.11.2011.)

## 4.2 Asymmetrinen salaus

Salaustekniikat ovat tuhansien vuosien ajan perustuneet lähettäjän ja vastaanottajan yhteiseen salaisuuteen. Nykyään on mahdollista käyttää tekniikkaa, jossa salausavain voi olla julkinen, mutta salauksen purkuun käytetään salattua avainta. Erilaisten avainten vuoksi tätä tekniikkaa kutsutaan asymmetriseksi salaukseksi. (Järvinen 2003, s. 77-105.)

Salaisten avainten ongelma on niiden vaihtaminen Internetin välityksellä ja samalla estää niitä joutumasta vääriin käsiin. Jokainen, joka tuntee salaisen avaimen, voi purkaa viestin. Yksi vastaus tähän ongelmaan on asymmetrinen salaus, johon liittyy kaksi avainta eli avainpari. Julkinen avain on vapaasti saatavilla kaikille, jotka haluavat lähettää sinulle viestin. Toinen, yksityinen avain pidetään salassa niin, että vain sinä tiedät sen. Viesti joka on salattu julkisella avaimella, voidaan purkaa ainoastaan samaa algoritmia käyttämällä, mutta käyttämällä oikeaa yksityistä avainta. Viesti joka on salattu yksityisellä avaimella, voidaan siis purkaa vain käyttämällä samaa avainta, mitä käytettiin salaukseen. Tämä tarkoittaa sitä, että käyttäjän ei tarvitse olla huolissaan julkisen avaimen jakamisesta Internetissä. Ongelmana asymmetrisessä salauksessa on kuitenkin se, että se on hitaampi kuin symmetrinen salaus. Se vaatii paljon enemmän laskentatehoa salaukseen ja salauksen purkuun. (Microsoft 2011b, hakupäivä 1.10.2011.)

### AES

Advanced Encryption Standard (AES) on Yhdysvaltain hallituksen kehittämä symmetrinen salausalgoritmi. AES on tammikuussa 1997 Yhdysvaltain kansallinen standardointi ja teknologia–instituutin (NIST) järjestämän projektin tulos, jolla haluttiin löytää järeämpi korvaaja Data Encryptin Standard (DES) algoritmille. Algoritmin vaatimuksena oli olla ilmainen maailmanlaajuisesti ja tarjota riittävän tasoista tietoturvaa tiedon salaukseen seuraavaksi 20–30 vuodeksi. Sen tuli olla helppokäyttöinen laitteisto- ja ohjelmistotasolla ja tarjota hyvää puolustusta lukuisia hyökkäysteknikoita vastaan. (Pelzl & Paar, hakupäivä 24.11.2011.)

### Twofish

Twofish on symmetrinen salausalgoritmi, joka on julkaistu vuonna 1998. Se oli yksi viidestä Advanced Encryption Standard (AES) kilpailun finaalisteista, jossa valittiin seuraajaa DES-salaukselle. Twofish salaimessa lohkon koko on 128 bittiä ja avaimen koko vaihtelee välillä 128-256 bittiä. Twofish ei ole patentoitu ja se on lisenssivapaa kaikille käyttäjille. (Schneier, hakupäivä 25.11.2011.)

### RSA

RSA on julkisen avaimen salausalgoritmi, joka perustuu yksityiseen ja julkiseen avaimeen sekä siihen, ettei yksityistä avainta voida nykytekniikalla käytännössä johtaa julkisesta avaimesta. Salattuja viestejä voidaan luoda julkisen avaimen avulla, mutta viestin lukemiseen tarvitaan lisäksi yksityinen avain. RSA on kehitetty vuonna 1977. Nimi RSA tulee sen kehittäjien, Ron Rivest, Adi Shamir ja Leonard Adleman, mukaan. (Hazan & Rundatz, hakupäivä 25.11.2011.)

# 5 LINUXIN SALAUSOHJELMAT

Yksi parhaista tavoista suojata tietokoneelle tallennettua arkaluontoista materiaalia on käyttää salausohjelmistoa. Salausohjelmistolla salataan tiedot niin, ettei niitä voida purkaa ilman erillistä avainta. Salausohjelmisto on tärkeä osa tietoturvaa, ja Linuxiin on olemassa useampia ohjelmistoja tähän käyttötarkoitukseen.

**Rsyncrypto**

Rsyncrypto on työkalu, jota voidaan käyttää salaukseen niin, että salattu data on mahdollista synkronoida toiseen palvelimeen käyttäen rsynciä. Rsyncrypton on suunnitellut Lingu Open Source Consulting. Rsyncrypto käyttää julkisen avaimen salausta, jossa jokainen tiedosto saa oman sattumanvaraisesti generoidun symmetrisen avaimen, jota kutsutaan istuntoavaimeksi (session key). Istuntoavain on salattu käyttäen julkista avainta. Lisäksi käytetyt algoritmit ovat niin perustyyppisiä kuin mahdollista. Rsyncrypto käyttää tiedostojen kryptaamiseen symmetristä lohkosalausta (AES). Jokainen tiedosto kryptataan erikseen uniikilla avaimella. Avain on tallennettu kahteen eri paikkaan. Toinen avaimista on varsinainen avaintiedosto, ja toinen on sisällytetty kryptattuun tiedostoon itseensä. Toinen kopio on kryptattu käyttämällä RSA julkista avainta, joka voidaan jakaa kaikille kryptatuille tiedostoille. Lisäksi peruskaava salaukseen perustuu suurelta osin CBC (Cipher-block chaining) -mallin salaukseen. CBC on lohkokoodausmenetelmä, jossa lähdetekstilohkolle aina ennen salausta suoritetaan XOR-operaatio edellisen salatun lohkon kanssa. CBC:llä koodatussa viestissä on hyvin vaikea havaita mitään säännönmukaisuuksia. (Rsyncrypto manual, hakupäivä 12.10.2011.)

**TrueCrypt**

TrueCrypt on yksi suosituimmista levynsalaustyökaluista. Sillä voidaan salata ja purkaa tiedostoja reaaliajassa. Ohjelmalla voidaan luoda virtuaalisia salattuja levyjä yksittäiseen salattuun osioon tai koko tallennuslaitteeseen. Ohjelma luo virtuaalisen salatun levyn tiedostoksi ja tiedosto liitetään järjestelmään kuin fyysinen levy. TrueCryptillä voidaan myös salata kokonaisia osioita tai tallennuslaitteita kuten USB-muisteja tai kiintolevyjä. Rinnankytkennän ja toimintojen limittämisen ansiosta dataa voidaan lukea ja kirjoittaa yhtä nopeasti kuin salaamattomissa levyissä. TrueCryptin käyttämät salausalgoritmit ovat AES-256, Serpent ja Twofish. (Auza 2010, hakupäivä 25.11.2011.)

### GNU Privacy Guard

GNU Privacy Guard (GPG) –ohjelman käyttö perustuu asymmetristen avainparien käyttöön. GPG on ilmaisohjelma, jota käytetään komentorivipohjaisesta käyttöliittymästä, mutta sille on olemassa myös graafisia käyttöliittymiä. GnuPG tukee useita eri salausalgoritmeja. (Auza, 2010, hakupäivä 25.11.2011.)

### Mcrypt

Mcrypt on suositun UNIX crypt packagen ja crypt–komennon korvaaja. Crypt oli salaustyökalu, jonka käyttämä algoritmi oli hyvin lähellä toisen maailmansodan kuuluisimman salauskoneen Enigman algoritmia. Mcrypt tarjoaa saman toiminnallisuuden kuin edeltäjänsä, mutta käyttää useita moderneja algoritmejä, kuten AES. (Auza, 2010, hakupäivä 25.11.2011.)

# 6 TYÖKALUT VARMUUSKOPIOINTIIN LINUXISSA

Kahden koneen väliseen tiedostojen varmuuskopiointiin ja synkronointiin on olemassa omia työkaluja, kuten rsync ja unison. Perinteisiä varmuuskopiointityökaluja ovat muun muassa cpio, tar ja dd, joiden käyttöön on olemassa hyviä ohjeita ja esimerkkiskriptejä. Tiedot on perinteisesti varmennettu nauhalle, mutta nykyisillä suurikapasiteettisilla kiintolevyillä tärkeät tiedot tallennetaan usein kiintolevylle. Kiintolevyltä tiedot on helppo siirtää tarvittaessa toiseen paikkaan. Erityisiä varmuuskopiointiohjelmia on runsaasti eri tarpeisiin. (Linux Wiki 2011a, hakupäivä 21.11.2011.)

**Tar**

Tar (tape archiver) on alun perin kehitetty tiedostojen arkistointiin nauhalle. Tar on komentoriviltä käytettävä ohjelma ja se on tarkoitettu tar-pakettien käsittelyyn. Tar-paketit sisältävät tiedoston tai useita tiedostoja ja hakemistoja koottuna yhden tiedoston sisälle. Koottuja paketteja kutsutaan tar–palloiksi. Linux-maailmassa tar-paketti (tarball) on samankaltainen rooli kuin Windowsin zip-tiedostoilla. Tar-pakettimuoto ei tosin itsessään sisällä pakkausta. Pakkaus yhdistetään joko bzip2, tai gzip-pakkaustekniikkaan, ja tiedostopäätteeksi tulee tar.bz2 tai tar.gz käytetyn pakkaustekniikan mukaan. Tar-paketti on Linux-maailman ylivoimaisesti käytetyin tiedonpakkaustapa, ja esimerkiksi jaeltavien ohjelmien lähdekoodit ovat käytännössä lähes aina tässä muodossa. Tar-ohjelma pystyy käsittelemään myös pakattuja tar -arkistoja. (Smith 2009, s. 254.)

**Cpio**

Cpio on myös alun perin kehitetty nauha-arkistointia varten. Ohjelman nimi on lyhenne sanoista ”Copy In Out”. Cpio:lla on oma arkistoformaattinsa, ja tiedostopäätteenä on tyypillisesti cpio. Ohjelmalla on mahdollista myös lukea muitakin tiedostoformaatteja, kuten esimerkiksi tar-arkistoja. Cpio on samankaltainen ohjelma kuin tar, mutta cpion suosio on huomattavasti vähäisempää tar:n verrattuna. Tämä voi johtua cpion käyttötavasta, joka poikkeaa muista arkistointiohjelmista. Ohjelman yhteydessä käytetään termejä In ja Out. Monessa muussa arkistointiohjelmassa termit ovat Create ja Extract. In tarkoittaa tiedostojen lukemista arkistosta tiedostojärjestelmään päin, ja Out tarkoittaa tiedostojen kopioimista järjestelmästä arkistoon. (Linux Wiki 2011b, hakupäivä 21.11.2011.)

### Dd

Dd on työkalu tiedostojen matalan tason kopiointiin ja muuntamiseen toiseen muotoon. Dd kopioi tiedoston sisällön bitintarkasti, ja tästä syystä työkalulla voidaan tehdä identtinen kopio levyn osiosta toiselle osiolle. Dd pystyy tekemään varmuuskopion myös kokonaisesta kovalevystä, vaikka kovalevyn tiedostojärjestelmä olisi tuntematon. Dd ei ole varsinainen varmuuskopiointiohjelma, mutta sitä voidaan käyttää varmuuskopiointiin. (Linux Wiki 2011c, hakupäivä 21.11.2011.)

### Rsync

Rsync on nopea ja erittäin monipuolinen tiedostojen kopiointityökalu, jolla ylläpidetään hakemistorakenteen ajantasaista kopiota. Rsync soveltuu hyvin varmuuskopion säilyttämiseen joko samalla koneella tai toisessa sijainnissa. Rsync on hyvä ratkaisu, kun halutaan varmentaa dataa verkon yli toiselle palvelimelle. Ohjelmaa käytetään myös kahden koneen väliseen synkronointiin. Rsync oletuksena kopioi vain muuttuneen datan, minkä vuoksi synkronointi tapahtuu nopeasti. Koska rsync kopioi vain edellisen synkronointikerran jälkeen muuttuneet tiedostot ja suurten tiedostojen muuttuneet osat, minimoidaan tiedonsiirtoon käytetty aika ja kaista. (Samba.org, How Rsync works, hakupäivä 10.11.2011.)

Andrew Tridgellin kehittämä delta-encoding -algoritmi vähentää datan siirtoa verkossa siirtämällä vain erot lähdetiedoston ja olemassa olevan tiedoston välillä. Rsync löytää tarpeelliset siirrettävät tiedostot käyttämällä "quick check" -algoritmia, joka etsii tiedostoja, jotka ovat muuttuneet kooltaan tai tiedoston muokkausaika on muuttunut. (Samba.org, How Rsync works, hakupäivä 10.11.2011.)

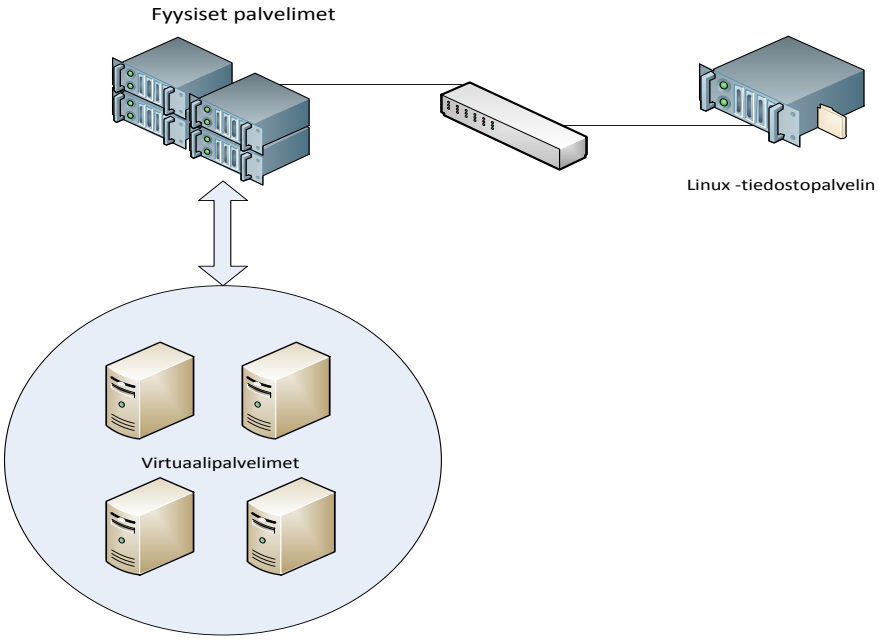
Algoritmi toimii niin, että päivitettävä tiedosto pilkotaan vakiopituisiin lohkoihin ja jokaiselle lohkolle lasketaan MD5-tarkastussumma sekä juokseva tarkastussumma. Tarkisteet lähetetään tiedoston lähettäjälle ja lähettävä tietokone laskee tarkisteiden avulla, mikä osa tiedostosta on muuttunut. Muuttunut data ja informaation siitä, miten se on muuttunut, liitetään vastaanottajan tiedostoon. Lopputuloksena on identtinen kopio lähettäjän versiosta. (Samba.org, How Rsync works, hakupäivä 10.11.2011.)

**Unison**

Unison on tiedostojen synkronointiin kehitetty työkalu. Se mahdollistaa kahden samanlaisen, mutta eri paikassa sijaitsevan tiedosto- ja hakemistokokoelman olemassa olon. Tiedostoja voidaan muokata erikseen kummassakin paikassa ja sen jälkeen suorittaa synkronointi, jolloin kummassakin paikassa on ajantasainen ja sama data. Unison toimii kahden tietokoneen väliseen synkronointiin, ja sitä voidaan käyttää myös SSH–etäyhteydellä. Tiedonsiirto on optimoitu käyttämällä pakkausprotokollaa, joka muistuttaa rsyncin toimintaa. (Gattol, hakupäivä 22.11.2011.)

# 7 KOHDEYRITYKSEN TIETOVERKKO

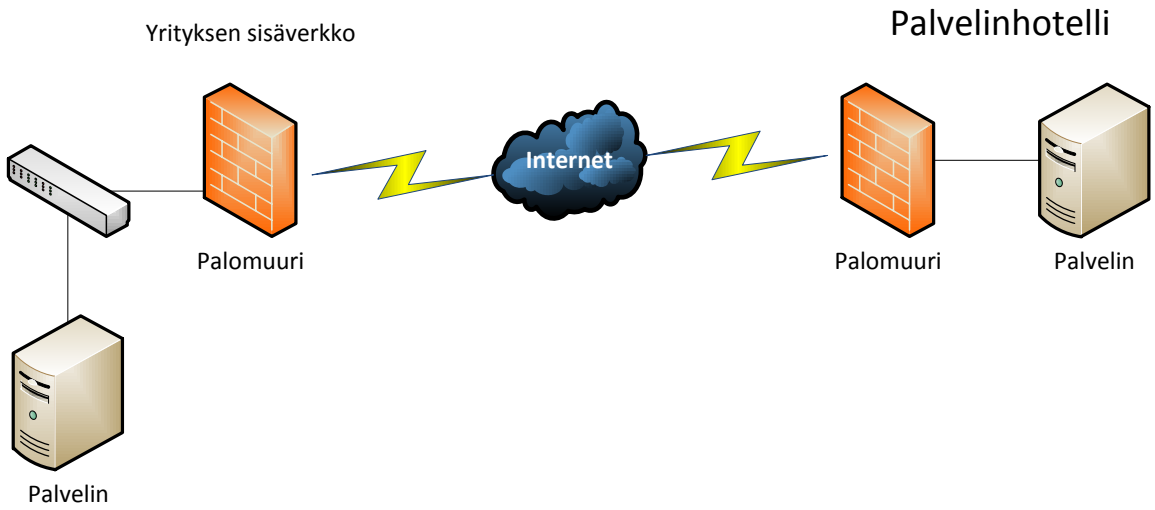
Varmennettavia fyysisiä palvelimia toimeksiantajan tietoverkossa on käytännössä kaksi, jotka sijaitsevat samassa konesalissa yrityksen toimitiloissa. Kuviosta 8 ilmenee sisäverkon palvelinten sijainti. Palvelimiin on asennettu Windows Server 2008 R2 -käyttöjärjestelmä ja Hyper-V-virtuaalisointiympäristö. Hyper-V–alustalla ajetaan virtuaalipalvelimia, joilla varsinaiset verkkopalvelut tuotetaan. Varmuuskopiot virtuaalipalvelimista otetaan erillisellä ohjelmalla, joka on asennettu kumpaankin fyysiseen palvelimeen. Yritykselle on hankittu Windows-palvelinten varmistusta varten Backupassist–ohjelma. Ohjelmalla otetaan varmuuskopio jokaisesta virtuaalipalvelimesta, joka siirretään Linux–tiedostopalvelimelle. Verkon käyttäjien tiedostot tallennetaan verkkolevyille. Näistä tiedostoista otetaan erikseen varmuuskopio, joten yksittäisen tiedoston palautus on mahdollista.




KUVIO 7. Sisäverkon palvelimet

Data on varmennettu ensin salaamattomana tiedostopalvelimelle yrityksen toimitiloissa. Sama data kryptataan ja lähetetään yrityksen tilojen ulkopuolella sijaitsevalle tiedostopalvelimelle, jolle on vuokrattu paikka palvelinhotellista. Kryptaus on toteutettu rsyncrypto–ohjelmalla, joka käyttää AES-salausta. Datan synkronointi palvelinten välillä on toteutettu rsync-ohjelmalla. Salaus ja synkronointi palvelinten välillä suoritetaan perjantaisin kello 16 jälkeen. Kuviossa 9 on esitetty

palvelinjärjestelmän rakenne yrityksen sisäverkossa sekä palvelinhotellissa. Yritys on kytketty Internetiin 10/10 Mb/s -yhteydellä.




KUVIO 8. Tiedostopalvelimet

# 8 LINUX – PALVELIMET

Kohdejärjestelmän varmennukseen on varattu kaksi Dell–räkkipalvelinta. Palvelimiin asennetaan Linux–käyttöjärjestelmät ja yrityksen ulkopuolisen verkon palvelimen tietoturvaominaisuuksia kiristetään. Käyttöjärjestelmäjakelua valittaessa kiinnitettiin huomiota suuresti Dell–palvelinten ja käyttöjärjestelmän yhteensopivuuteen.

## 8.1 Käyttöjärjestelmät

Linux–jakelupaketteja on olemassa satoja erilaisia, joista osa on suunnattu käyttötarpeiden mukaa erityyppiseen käyttöön. Jakeluversiota valittaessa, kiinnitettiin huomiota erityisesti käyttöjärjestelmän vakauteen. Palvelinten etähallintaa ja monitorointia varten käyttöjärjestelmän tulee tukea Dellin Open Manage Server (OMSA) –ohjelmistoa. Red Hat Linux –pohjaisissa jakeluissa on tuki suoraan OMSA:lle linux.dell.com –repositorion kautta.

Tiedostopalvelinten käyttöjärjestelmäksi valittiin Red Hat Entrerprise Linuxiin pohjautuva CentOS (Community Enterprise Operating System). CentOS tarjoaa yritystason käyttöjärjestelmän maksuttomasti. CentOS käyttää RPM-paketinhallintaa ja virallisten CentOS–pakettien lisäksi on myös mahdollista käyttää Red Hatin tekemiä paketteja. CentOS on ilmainen versio Red Hat Enterprise Linux –jakelusta. Tietoturvapäivitykset ovat myös CentOS–jakelussa maksuttomia. Koska CentOS ja Red Hat ovat lähestulkoon täydellisiä klooneja toisistaan, ne ovat sataprosenttisesti binääriyhteensopivia, jonka vuoksi Red Hat Linuxille tuotetut ohjelmistot toimivat CentOS:ssa ilman muutoksia. CentOS on lisäksi käytössä hyvin vakaa ja dokumentoitu järjestelmä. (Bashton, hakupäivä 18.10.2011.)

## 8.2 Levyjärjestelmä

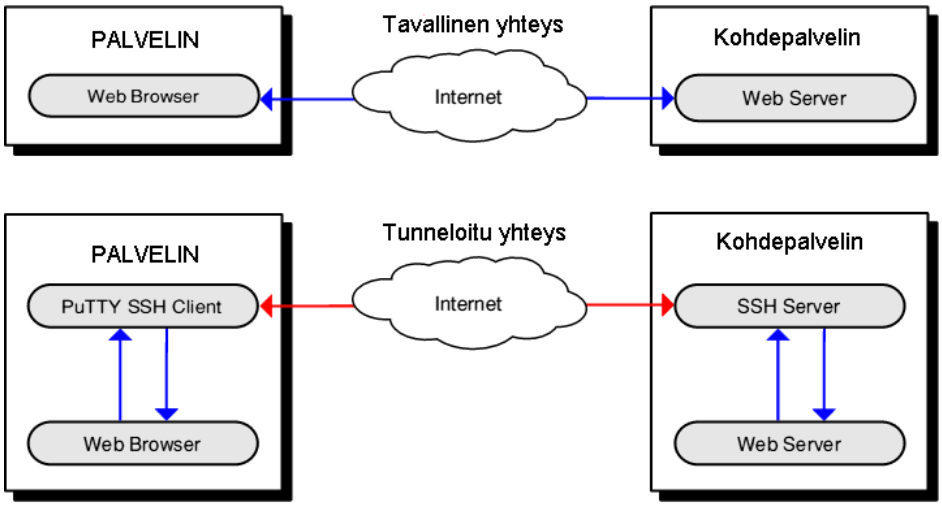
Tiedostopalvelinten levyjärjestelmä on toteutettu RAID5-menetelmällä. Hot Swapin ansiosta rikkoontunut levy voidaan vaihtaa sammuttamatta konetta ja uudelleen järjestää järjestelmään. Tekniikkaa käytetään erityisesti palvelinkoneissa, joissa viallinen kiintolevy tai virtalähde korvataan uusilla. Jotta laite voidaan luokitella Hot Swap –laitteeksi, se täytyy voida kytkeä laitteeseen sen ollessa käynnissä. Hot Swap –laitteen tulee myös olla lähes välittömästi

käyttökunnossa. Suurimmat edut Hot Swap –laitteista ovat niiden vaihtamisen helppous ja nopeus. Rikkoontunut komponentti voidaan vaihtaa uuteen ilman käyttökatkoksia. Hot Spare –menetelmällä saavutetaan myös parempi vikasietoisuus. Yksi levyistä on varattu lennossa käyttöön otettavaksi. Levyn rikkoontuessa varalle määritelty levy aktivoituu automaattisesti (Conjecture Corporation, hakupäivä 14.11.2011.) Tallennuskapasiteettia on hankittu yrityksen tarpeisiin riittävästi ja palvelimissa on laajennustilaa jäljellä tulevaisuutta varten.

### 8.3 Palvelimen toiminnan monitorointi

Koska toinen palvelimista on sijoitettu yrityksen toimitilojen ulkopuolelle, täytyy palvelimen toimintaa pystyä tarkkailemaan ja hallitsemaan etäyhteydellä. Palvelimeen asennettiin OMSA-ohjelmisto. OMSA on www-pohjainen hallintaohjelmisto, jolla voidaan tarkkailla laitteiston toimintaa etäyhteydellä selainikkunan tai komentorivin kautta.

Koska palvelimelle on estetty pääsy muilta kuin SSH ja rsync -protokollilta, luodaan monitorointiohjelmistoa varten putkitettu SSH–yhteys ja käytetään ohjelmaa selaimella SSH-tunnelin läpi. SSH-tunneli on suojattu etäyhteys palvelinkoneeseen, jolla välitetään http-portin liikenne tunnelin kautta. Ilmaisessa Putty telnet ja SSH –asiakasohjelmassa on mahdollista muodostaa SSH-tunneli. Kuviossa 10 on esitetty SSH-tunnelin käytön perusidea.




KUVIO 9. (Precende technologies, hakupäivä 22.12.2011, muunneltu)

## 8.4 Tietoturvan parantaminen

Salaamattoman datan palvelin sijaitsee yrityksen palomuurilla suojatussa sisäverkossa. Lisäksi sisäverkko on toteutettu NAT-osoitteenmuunnostekniikalla yksityisen luokan osoitteilla, joten yksittäiset koneet eivät näy Internetissä suoraan.

Yrityksen ulkopuolella olevaa palvelinta hallitaan SSH-etäyhteydellä komentoriviltä. Palvelinhotellissa on laitteistopohjainen palomuuri, jonne on palvelun tarjoaja määritellyt sallitut IP-osoitteet, joista on pääsy palvelimelle. Lisäksi käyttöjärjestelmä on varustettu omalla iptables–ohjelmistopalomuurilla, johon on tehty pääsylistaan poikkeus SSH–protokollalle tietystä IP-osoitteesta.

SSH-palvelimelle kirjautuminen on estetty pääkäyttäjänä (root). Esto on toteutettu SSH:n konfigurointitiedostoon pääkäyttäjän kirjautumista käsittelevää arvoa muuttamalla. SSH oletuksena kuuntelee porttia 22. Palvelimessa SSH-portti on konfiguroitu käyttämään muuta porttia. Portin vaihto parantaa hieman tietoturvaa ja näin voidaan torjua joitakin hyökkäyksiä. SSH–oletusportti vaihdetaan muuttamalla Port –riviä tiedostossa SSH:n konfigurointitiedostossa.

Security Enhanced Linux (SELinux) on Yhdysvaltain NSA:n Linuxin ytimeen kehittämä laajennus. SELinux on turvallisuusjärjestelmä, jolla rajoitetaan ohjelmien toimintaoikeuksia niitä varten kirjoitetuilla säännöillä.

SELinux - asetuksia säädetään muokkaamalla asetustiedostoa /etc/sysconfig/selinux. Kyseessä olevassa tiedostossa SELINUX - kentä arvolla määritellään SELinuxin tila. Mahdollisia arvoja ovat SELINUX=disabled, SELINUX=permissive ja SELINUX=enforcing.

- Disabled -arvolla SELinux on kokonaan poistettu käytöstä
- Permissive -arvolla SELinux sallii kaiken toiminnan, mutta varoittaa sääntöjen vastaisesta toiminnasta
- Enforcing-arvolla SELinux on täytössä täysin ja estää säätöjen vastaisen toiminnan kokonaan.

(SeLinux Project, hakupäivä 24.11.2011.)

### 8.5 Datan salaus ja synkronointi

Tiedostot synkronoidaan palvelinten välillä Rsync-ohjelmalla. Rsync-ohjelma valittiin, koska varmennettavan datan joukossa on paljon suuria tiedostoja, jotka muuttuvat vain vähän kerralla. Rsync-ohjelma toimii hyvin hitaan ja epävakaan linkin yli, ja sillä on riittävän hyvät lokiominaisuudet.

Rsyncrypto-ohjelmaa käytetään datan salaukseen ja salauksen purkuun. Rsyncrypton salaamat tiedostot synkronoidaan rsync-ohjelman avulla toiselle palvelimelle. Rsyncrypto salaa tiedostot siten, että rsync-ohjelman tehokkuus tiedonsiirrossa kärsii mahdollisimman vähän. Rsyncrypto salaa tiedostot käyttäen AES-salausta. Jokainen tiedosto salataan omalla avaimella ja avain tallennetaan kahteen paikkaan. Toinen on varsinainen avaintiedosto, ja toinen avain on sisällytetty jokaiseen salattuun tiedostoon. Tiedostoon sisällytetty avain on salattu RSA julkisella avaimella, joka voidaan jakaa kaikille salatuille tiedostoille. Ohjelma noudattaa GNU komentorivin syntaksia, joten ohjelma on käytettävyydeltään muiden Linuxin komentoriviohjelmien kaltainen. Rsyncryptossa on paljon asetuksia ja ohjelman toimintaa voi haluttaessa säätää monipuolisesti. Oletuksena rsyncrypto pakkaa salattavia tiedostoja. Ohjelma käyttää gzip-pakkausta. Vaikka tiedostoja pakataan, ei rsync-ohjelman tehokkuus juurikaan kärsi.

Rsync-ohjelman käyttöön löytyy Internetistä paljon hyviä valmiita skriptejä, joista voidaan helposti muokata omaan käyttötarkoitukseen soveltuva skripti. Rsync on myös helppo konfiguroida ja ottaa käyttöön. Ensimmäinen versio varmuuskopioista tehtiin sisäverkossa ennen offsite-palvelimen fyysistä siirtoa yrityksen ulkopuolelle. Rsync kuuntelee oletuksena porttia 873, joten iptables–palomuuriin täytyy myös määrittää portti 873 sallituksi. Lisäksi SELinuxiin täytyy sallia rsyncin käyttö setsebool –komennolla. Rsync käyttää omaa tiedostoa käyttäjänimelle ja salasanalle. Tätä varten luodaan salasanatiedosto ja ensimmäiselle riville kirjoitetaan rsync-ohjelman käyttämä käyttäjätunnus ja salasana. Tiedostolle määritellään oikeudet niin, että vain järjestelmän pääkäyttäjä voi lukea tiedostoa. Rsync–daemon on konfiguroitu tiedostoja vastaanottavaan palvelimeen.

Synkronointi palvelinten välillä on hoidettu skriptaamalla rsync-ohjelman käyttämät komennot ja ne suoritetaan automaattisesti kerran viikossa. Rsync-skriptiin on tiedostojen siirron lisäksi ohjelmoitu ominaisuus, joka lähettää ylläpitäjälle sähköpostin, jossa on tieto toimeenpiteen onnistumisesta sekä siirretyn datan määrästä ja mahdollisista virheistä lokitiedostossa.

Lokitiedostot tallennetaan palvelimelle suoritusajankohdan perusteella myöhempää tarkastelua varten.

Kun varmuuskopioista halutaan säilyttää useampia sukupolvia komentoon lisätään parametri –link-dest. Tämä on hyvä tapa ottaa täysi varmistus tiedostoista ilman, että hukataan paljon tallennuskapasiteettia. Rsync linkittää muuttumattomat tiedostot edelliseen varmuuskopioon ja varaa tallennustilaa vain muuttuneille tiedostoille. Kohdejärjestelmässä säilytetään varmuuskopioista kolme sukupolvea. Versiosukupolvia varten on palvelimella olemassa oma skripti, joka suoritettaessa tekee täyden varmistuksen ja varmuuskopion polkuun lisätään kuluva päivämäärä ja aika. Aina varmuuskopiota luotaessa toimenpiteen ajankohta ohjataan omaan tiedostoonsa. Erillisellä skriptillä lasketaan poistettava päivämäärä. Poistettavan päivämäärän sisältävä tiedosto synkronoidaan offsite-palvelimelle, jossa ajetaan päivittäin toinen skripti joka poistaa vanhimman version varmuuskopioista sekä viimeisimmän päivämäärän sisältävän tiedoston, joten varmuuskopioiden vaatima tallennuskapasiteetti pysyy kohtuullisena.

Ajettavat skriptit suoritetaan automaattisesti. Automatisoinnista huolehtii Unix-pohjaisten käyttöjärjestelmien ajastuspalvelu cron. Nimi cron tulee kreikan kielen sanasta ”chronos”, joka tarkoittaa aikaa. Cronin käyttö helpottaa tiettyjen toimintojen ajastamista ja automatisointia. Cronilla on usein Linuxissa automatisoitu varmuuskopiointi. Ajoituksista huolehtivat prosessit crond ja anacron. Ajastimia muokataan crontab–ohjelmalla, jolla ohjataan crond-daemonia. Crond ajaa komennot taustalla ja tarkistaa minuutin välein suoritettavia komentoja. Tehtävä suoritetaan, kun aikamääritykset täsmäävät nykyhetkeen. (Private World Domination Inc, hakupäivä 25.11.2011.)

Järjestelmään on suunniteltu ja testattu virtuaaliympäristössä toiminnallisuutta, jossa siirretystä datasta otetaan näytteitä sattumanvaraisesti ja verrataan niitä alkuperäisiin tiedostoihin. Näin saadaan siirrettyjen tiedostojen eheyden varmistamiseen lisäturvaa. Lisätoiminnoille on kirjoitettu skripti, jossa siirretystä datasta sattumanvaraisesti valitaan muutamia tiedostoja. Valittujen tiedostojen md5-tiiviste ja hakemistopolku tallennetaan erilliseen tiedostoon ja tiedosto lähetetään takaisin yrityksen sisäverkon palvelimelle. Siirretyn tiedoston sisällön perusteella valitaan sisäverkon palvelimella vastaavat tiedostot. Tiedostojen md5-tiivistettä ja hakemistopolkua verrataan vastaanotetun tiedoston sisältämiin tietoihin ja mikäli tiedot täsmäävät, varmistutaan tiedon eheydestä ja oikeellisuudesta.

Tiedostojen palauttaminen offsite-palvelimelta voidaan suorittaa verkon yli käyttäen rsynciä. Mikäli verkkopalveluja ei ole saatavilla tai tietoliikenneyhteydet ovat heikkoja, voidaan offsite-palvelin hakea palvelinhotellista ja suorittaa tiedostojen palautus suoraan lähiverkossa tai ulkoista kiintolevyä käyttäen. Tiedostojen salauksen purku suoritetaan avaintiedostoa käyttämällä. Avaintiedosto on tallennettu turvalliseen paikkaan. Rsyncypton komentoon lisätään salauksen purkamisen suorittava parametri ja avaintiedoston polku.

# 9 POHDINTA

Opinnäytetyön tavoitteena oli toteuttaa yrityksen offsite-varmuuskopiointiin kustannustehokas ratkaisu. Ratkaisuun oli käytettävissä kaksi fyysistä palvelinta, joihin hankittiin kiintolevyjä tallennuskapasiteetin lisäämiseksi. Muilta osin fyysiset palvelimet soveltuivat varmuuskopiointikäyttöön sellaisenaan.

Windows-palvelinten varmistus sekä sopivan palvelinhotellipalvelun valitseminen olivat myös osana toiminnallisessa toteutuksessa, mutta nämä osat rajattiin raportista pois osaksi aiheen laajuuden vuoksi, mutta myös siksi, että halusin keskittyä raportissa pelkästään Linux-käyttöjärjestelmällä toteutettuihin toimintoihin.

Aikaisempaa kokemusta Linuxista palvelinkäytössä minulla oli jo. Opinnäytetyön aikana sain syventää tietojani ja taitojani Linuxin parissa sekä mielenkiintoni avoimen lähdekoodin sovelluksilla toteutettuihin järjestelmiin lisääntyi huomattavasti. Tutkiessani erilaisia vaihtoehtoja offsite-varmistukseen tuli vastaan useita enemmän tai vähemmän kaupallisia ratkaisuja. Kuitenkaan kohdeyrityksen käyttöön soveltuvia valmiita ratkaisuja ei sellaisenaan löytynyt muita kuin rsyncryptolla ja rsync-ohjelmalla toteutettuna, muokattiin näiden ohjelmien avulla toimiva ja tehokas offsite-varmennus.

Linuxilla ja ilmaisilla ohjelmilla toteutetusta offsite-varmistuksesta saatiin juuri halutun kaltainen. Yksinkertainen ja helposti muokattavissa oleva järjestelmä taipuu yrityksen tarpeisiin, vaikka datamäärä lisääntyisikin. Kokonaisuutena opinnäytetyöni toiminnallisen osan toteutus onnistui kattamaan sille asetetut vaatimukset. Projektin onnistumisesta kuuluu iso kiitos yrityksen IT-osastolle, jonka kanssa toiminnallinen osa yhteistyössä toteutettiin.

Raportin kirjoittamisen loppuvaiheessa asensin virtuaalisen Linux-palvelinympäristön ja konfiguroin palvelimet vastaamaan opinnäytetyössä toteutettua järjestelmää. Virtuaalisessa ympäristössä sain palautettua mieleeni asennuksen eri vaiheet ja pääsin vielä testaamaan ohjelmien toimintaa käytännössä. Tästä oli suuri apu raportin kirjoittamisessa, koska kaikkien vaiheiden muistaminen jälkikäteen oli haastavaa.

Jatkokehitykselle jää tilaa tiedostojen verifiontiin suunnitellulla toiminnallisuudella, joka varmistaa salatun ja siirretyn datan aitouden. Toimintaa varten kehitin ohjelman, jossa varmuuskopioidusta datasta otetaan satunnaisia näytteitä tiedostoista ja verrataan näiden tiedostojen MD5-tiivistettä

sekä tiedostopolkua alkuperäiseen tiedostoon ja tiedostopolkuun. Toiminnallisuus saatetaan loppuun yrityksen toimesta. Skriptejä on jo valmiiksi kirjoitettu ja toimintaa olen testannut virtuaalisessa ympäristössä. Opinnäytetyön toiminnallisen osan aikana opin paljon myös bash-skriptaamisesta, joilla automatisoitiin ajettavat komennot. Jatkossa työn ansiosta opituilla tiedoilla kykenen toimimaan sujuvammin Linux-ympäristössä ja taitoni karttuivat huomattavasti.

# LÄHTEET


Auza, J. (13. 1 2010). Free and Open Source Encryption Software for Linux. Haettu 25. 11 2011 osoitteesta TechSource: http://www.junauza.com/2010/01/free-and-open-source-encryption.html

Bashton. Linux Server Distribution Comparsion. Haettu 18.10.2011 osoitteesta Bashton.com: http://bashton.com/linux-distribution-comparison/

Biasci, L. Data Encryption Standard (DES). Haettu 18.10.2011 osoitteesta SearchSecurity: http://searchsecurity.techtarget.com/definition/Data-Encryption-Standard

Buffington, J. (2010). Data Protection for Virtual Data Centers. Indianapolis: Wiley Publishing Inc.

Conjecture Corporation. What is Hot Swappable? Haettu 14.11.2011 osoitteesta wiseGEEK: http://www.wisegeek.com/what-is-hot-swappable.htm

Frisch, A. (2003). Essential system administration. O'Reilly Media.

Gattol, M. File Syncronation With Unison. Haettu 22.11.2011 osoitteesta Markus Gattol: http://www.markus-gattol.name/ws/unison.html

Greenberg, B. (2011). Are your backup systems secure? Haettu 22.11.2011 osoitteesta Infosec recources: http://resources.infosecinstitute.com/are-your-backup-systems-secure/

Hazan, F.;& Rundatz, F. What is RSA Algorithm. Haettu 25.11.2011 osoitteesta SearchSecurity: http://searchsecurity.techtarget.com/definition/RSA

Hoff, C. (2008). Cryptic Backup: a framework for automated compression, encryption and backup of data.

Järvinen, P. (2003). Salausmenetelmät. Docendo Finland Oy.

Kartalopoulos, S. V. (2009). Security of information and communication networks. John Wiley & Sons, Inc.

Kozierok, C. M. (2001). Hot Spares. Haettu 22.12.2011 osoitteesta The PC Guide: http://www.pcguide.com/ref/hdd/perf/raid/conf/advSpares-c.html

Lacie 2011. RAID. Haettu 24.11.2011 osoitteesta Lacie User Manuals: http://manuals.lacie.com/en/manuals/4big-quadra/raid

Linux Wiki 2011a, cpio. Haettu 21.11.2011 osoitteesta Linux Wiki: http://linux.fi/wiki/Cpio

Linux Wiki 2011b, dd. Haettu 21.11.2011 osoitteesta Linux Wiki: http://linux.fi/wiki/Dd

Linux Wiki 2011c, varmuuskopiointi. Haettu 21.11.2011 osoitteesta Linux Wiki: http://linux.fi/wiki/Varmuuskopiointi

Microsoft Corporation 2011a. Symmetric and asymmetric encryption. Haettu 1.10.2011 osoitteesta http://support.microsoft.com/kb/246071

Microsoft Corporation 2011b. Varmuuskopiointi. Haettu 6.10.2011 osoitteesta http://www.microsoft.com/finland/business/casestudies/comgate07.mspx

Mohan Krishnamurthy, E. S. (2008). How to cheat at securing linux. Elsevier, Inc.

Pelzl, J.;& Paar, C. The Advanced Encryption Standard (AES), Chapter 4. Haettu 24.11.2011 osoitteesta Understanding Cryptography: http://wiki.crypto.rub.de/Buch/download/Understanding-Cryptography-Chapter4.pdf

Petersen, R. (2008). Linux: The complete reference, Sixth Edition. The McGraw-Hill Companies.

Precende technologies. Using Putty to create a SSH tunnel. Haettu 22.12.2011 osoitteesta Precende Technologies: http://oldsite.precedence.co.uk/nc/putty.html

Preston, C. (2007). Backup and Recovery. O'Reilly Media Inc.

Private World Domination Inc. Cron Help Guide. Haettu 25.11.2011 osoitteesta Linuxhelp: http://www.linuxhelp.net/guides/cron/

Rsyncrypto. Man rsyncrypto. Haettu 12.10.2011 osoitteesta Rsyncrypto: http://pwet.fr/man/linux/commandes/rsyncrypto

Samba.org. How Rsync Works, A Practical Overview. Haettu 10.11.2011 osoitteesta Rsync Samba: http://rsync.samba.org/how-rsync-works.html

Schneier, B. Twofish. Haettu 25.11.2011 osoitteesta Schneier on security: http://www.schneier.com/twofish.html

SecMeter. SecMeter. Haettu 19.12.2011 osoitteesta Varmuuskopiointi: http://www.secmeter.com/varmuuskopiointi.html

SeLinux Project. FAQ. Haettu 24.11.2011 osoitteesta SELinux Project: http://selinuxproject.org/page/FAQ

Smith, R. W. (2009). Linux + Study Guide, 4th Edition. CompTIA.

Unix.com, S. P. help changing rsync script. Haettu 25.11.2011 osoitteesta unix - shell programming scripting: http://www.unix.com/shell-programming-scripting/154770-help-changing-rsync-script.html