



LAHDEN AMMATTIKORKEAKOULU
Lahti University of Applied Sciences

PELIPALVELINKLUSTERI

Digital Hammer Oy

LAHDEN
AMMATTIKORKEAKOULU
Tekniikan ala
Tietotekniikka
Tietoliikennetekniikka
Opinnäytetyö
Kevät 2015
Emil Paavola

Lahden ammattikorkeakoulu
Tekniikan ala

PAAVOLA, EMIL:

Pelipalvelinklusteri
Digital Hammer Oy

Tietoliikennetekniikan opinnäytetyö, 45 sivua

Kevät 2015

TIIVISTELMÄ

Tämän opinnäytetyön tavoitteena oli selvittää, miten luodaan helposti kloonautuva ja kasvatettava virtuaalinen Linuxin Ubuntu-käyttöjärjestelmäpohjainen pelipalvelinklusteri mobiilipeliyrityksen käyttöön. Vuonna 2013 perustettu lahtelainen mobiilipeliyritys Digital Hammer Oy tarvitsi käyttöönsä kyseisen klusterin tulevan mobiilipelinsä monipelidatan tallennusta varten. Pelipalvelinklusterin virtuaalisuudesta on suurta hyötyä, koska kustannukset laitehankintojen osalta voidaan siten minimoida, sillä palvelin on olemassa virtuaalisena hankitun pilvipalvelun sisällä, eikä näin ollen tarvita fyysistä palvelintietokonetta.

Tietotekniikassa klusterilla tarkoitetaan usean tietokoneen rypästä, jonka laitteet voivat tarvittaessa tehdä yhteistyötä ja suorittaa prosesseja tehokkaammin keskenään. Jos yksi tietokone klusterissa vioittuu, voivat muut koneet ottaa vioittuneen laitteen tehtävät hoidettavikseen. Klustereita käytetään etenkin sovelluksissa ja laitteissa, jotka ovat jatkuvasti käytössä, kuten esimerkiksi pelipalvelimet.

Virtuaalipalvelinpalvelun hankkiminen perustuu niin sanottuun IaaS-malliin (Infrastructure as a service), jossa palveluntarjoalta vuokrataan virtuaalinen tietokone tai palvelin. Virtuaalisia palvelinpalveluita tarjoavat muun muassa Amazon ja Yahoo.

Palvelimen pohjana toimii palveluntarjoajan verkkosivuilla luotu palvelinmikroinstanssi, joka voi olla esimerkiksi tietokantapohjainen. Instanssia varten luodaan ”key pair”- eli avainparitiedosto, joka toimii avaimena palvelimen sisään.

Palvelimen asetuksia muokataan erillisellä komentotulkki-sovelluksella ja sen kautta palvelinta myös hallinnoidaan ja asennetaan sovelluksia. Palvelininstanssin avainparia voidaan jakaa yrityksen työntekijöiden kesken, jotta kaikki pääsevät tarvittaessa muokkaamaan palvelimen asetuksia.

Asiasanat: klusteri, palvelin, virtuaalisuus, avainpari, tietoliikennetekniikka, IaaS, tietokanta, mikroinstanssi

Lahti University of Applied Sciences
Faculty of Technology

PAAVOLA, EMIL:

Game Server Cluster
Digital Hammer Ltd

Bachelor's Thesis in Telecommunications Technology, 45 pages

Spring 2015

ABSTRACT

The purpose of this thesis was to find out how to create a virtual game server cluster on the Linux Ubuntu operating system, which could be cloned and grown easily for the use of a mobile game company. Digital Hammer Ltd from Lahti needed the cluster to use it in an upcoming mobile game to store the game's multiplayer data. The benefit of using a virtual cluster rather than a physical one is that the costs for buying a set of computers to create a physical cluster can be minimized since the cluster is running inside a cloud service.

In telecommunications technology a cluster means a group of computers which can co-operate and run processes effectively together when needed. If a device fails in the cluster, the other devices can take over the processes. The clusters are mainly used in applications and servers which are constantly running, like for instance a game server.

A virtual server service is based on the so-called IaaS model (Infrastructure as a Service) where a user can lease a virtual computer or a server from the service provider. Service providers that lease virtual services are for example Amazon and Yahoo.

As the base of the server there is a micro instance that can be created at the user account of the service provider's Internet page. The instance can be configured to work as a database. To access the running instance, a key pair must be created for it, which works as a key to the server.

The settings of the running server can be configured via a shell application (command line) which is also used to administrate and to install the programs to the server. The key pair created for the micro instance can be distributed to the employees of the company so they can all access and configure the server settings when needed.

Key words: cluster, server, virtuality, key pair, telecommunications technology, IaaS, database, micro instance

LYHENNELUETTELO

ASF	Apache Software Foundation, Apachen sovellussäätiö
AWS	Amazon Web Services, Amazonin Internet-palvelut
BASH	Bourne Again Shell
CSMA/CD	Carrier Sense Multiple Access/Collision Detection, verkon kuulostelu/törmäyksen tunnistus
EC2	Elastic Compute Cloud, joustava laskentapilvi
EMR	Elastic MapReduce, joustava kartansupistus
FTP	File Transfer Protocol, tiedostonsiirtoprotokolla
GIF	Graphic Interchange Format, graafinen vaihtotiedostomuoto
HTTP	Hypertext Transfer Protocol, hypertekstin siirtoprotokolla
IaaS	Infrastructure as a Service, infrastruktuuri palveluna
IAM	Identity and Access Management, henkilöllisyys- ja pääsyhallinta
IP	Internet Protocol, Internet-protokolla
LAN	Local Area Network, lähiverkko
MAC	Media Access Control, median kirjatumisen valvonta
MS-DOS	Microsoft Disk Operating System, Microsoftin levykäyttöjärjestelmä
OSI	Open Systems Interconnection Reference Model, avointen järjestelmien referenssiyhdistelmämalli
PaaS	Platform as a Service, kehitysalusta palveluna
PHP	PHP: Hypertext Preprocessor, PHP: hypertekstin esiprosessori
PNG	Portable Network Graphics, kannettavat verkkografiikat
RAM	Random Access Memory, satunnainen pääsymuisti
RDS	Relational Database Service, relaatiotietokantapalvelu
SaaS	Software as a Service, sovellus palveluna
SMTP	Simple Mail Transfer Protocol, yksinkertainen postinsiirtoprotokolla

SQL	Structured Query Language, rakenteellinen kyselykieli
SSH	Secure SHell, suojattu kuori
TCP	Transmission Control Protocol, siirtokontrollointiprotokolla
UDP	User Datagram Protocol, käyttäjädatagrammiprotokolla

SISÄLLYS

1	JOHDANTO	1
2	TIETOVERKON PERUSTEET	2
2.1	OSI-malli	2
2.2	TCP/IP-malli	4
2.3	Ethernet-verkko	5
3	PALVELIN JA KLUSTERI	7
3.1	Palvelimen ominaisuudet	7
3.2	Klusterin ominaisuudet	8
3.3	Palvelimelle asennettavien sovellusten perusteet	11
3.3.1	MySQL	11
3.3.2	PHP	12
3.3.3	Apache	12
3.3.4	phpMyAdmin	12
3.3.5	Python	13
3.4	Bash-käskyt	13
4	VIRTUALISOINTI JA PILVIPALVELUT	15
4.1	SaaS	15
4.2	PaaS	15
4.3	IaaS	16
5	PELIPALVELINKLUSTERI MOBIILPELIYRITYKSEN KÄYTTÖÖN	17
5.1	Pelipalvelinklusterin suunnittelu	17
5.2	Pilvipalveluntarjoajat	19
5.2.1	Amazon Web Services	19
5.2.2	Windows Azure	20
5.2.3	Google Cloud	20
5.2.4	Palveluntarjoajien vertailu ja valinta	20
5.3	Palvelun hankkiminen Amazonilta	22
5.4	EC2-instanssin luonti	22
5.5	Avainparin luonti	29
5.6	Palvelimelle kirjautuminen SSH:n avulla	30
5.7	Palvelimen perusasetukset ja sovellusten asentaminen	31
5.8	Klusterin luonti	33

6 YHTEENVETO

40

LÄHTEET

42

1 JOHDANTO

Mobiilipelit ovat saavuttaneet todellisen huippunsa 2010-luvulla. Mobiilipelit ovat suosittuja kaiken ikäisten mobiililaitteiden käyttäjien keskuudessa ja yksinkertaisten ideoidensa vuoksi mobiilipelit ovat myös todella koukuttavia ja joillekin alan yrityksille maksullinen lisäsisältö ja oheistuotteet tuottavat hyviä tuloksia. Esimerkkinä hyvästä tuloksesta mainittakoon huippusuosion saavuttaneen Angry Birds -pelisarjan luonut suomalainen Rovio, jonka liikevaihto oli vuoden 2013 lopussa noin 154 miljoonaa euroa (Taloussanomat 2014). Vieläkin mittavampaan tulokseen on päässyt niin ikään suomalainen mobiilipeliyrittäjä Supercell huikauttavalla 1,545 miljardin euron liikevaihdollaan vuonna 2014 (Talouselämä 2015).

Hyvän mobiilipelin salaisuus on yksinkertaisuudessa ja omalaatuisissa ideoissa. Pelin ei tarvitse olla liian haastava, eikä pelin tarvitse omata huippugrafiikoita vaan tulee miettiä ratkaisua, mitä nykypäivän pelaaja haluaa peliltä, jotta mielenkiinto peliä kohtaan säilyisi järkkymättömänä.

On havaittavissa, että parhaiten tuottava ratkaisu mobiilipeliyrittäjille ovat ilmaisapelit, jotka käyttäjä voi ladata ilmaiseksi sovelluskaupasta ja pelata ilmaiseksi niin paljon kuin haluaa, mutta tällöin yleensä pelissä tulee usein vastaan mainoksia, joilla peliä sponsoroidaan. Maksullisella lisäsisällöllä pelissä voidaan edetä normaalia nopeammin.

Mobiilipeliala on jatkuvassa kasvussa, ja onkin ennustettu, että esimerkiksi Kaakkois-Aasian maissa, joissa pelejä ostettiin 1,1 miljardilla dollarilla vuonna 2014, pelimyynä kaksinkertaistuu vuoteen 2017 mennessä. Syinä tähän ovat kasvavat älylaitteiden markkinat ja etenkin älypuhelimien helpompi saatavuus maailmalla. (IGN 2015.)

Tämän opinnäytetyön tavoitteena on selvittää, kuinka luodaan helposti kopioitavissa eli kloonattavissa oleva virtuaalinen pelipalvelinklusteri. Lahtelainen Digital Hammer Oy on vuonna 2013 perustettu mobiilipeliyrittäjä, joka tarvitsi kyseistä klusteria tulevaa moninpeliään varten. Klusteria käytettäisiin moninpelidatan, kuten pelaajatietojen ja ennätys- eli ”high score” -listojen tallentamiseen.

2 TIETOVERKON PERUSTEET

Tietoverkolla tarkoitetaan usean tietokoneen muodostamaa verkkoa, joka muodostetaan, jotta tietokoneet voisivat kommunikoida keskenään. Maailman suurin tietoverkko on nimeltään Internet, joka kehittyi alun perin sotilaskäyttöön tarkoitettusta ARPANET-verkosta.

Tietoteknisiä asioita tutkittaessa on aina aluksi tärkeää selvittää asioiden alkuperä. Tietoliikennetekniikan maailma koostuu useista erilaisista standardeista ja malleista.

2.1 OSI-malli

OSI-malli eli Open Systems Interconnection Reference Model on vuonna 1983 käyttöönhyväksytty standardi, jonka avulla tiedonsiirtoprotokollat voidaan esittää seitsemänä eri kerroksena. Kerrokset eivät kaikki kommunikoi keskenään, vaan jokainen kerros pystyy kommunikoimaan ainoastaan ylä- ja alapuolellaan olevan kerroksen kanssa. Kerrokset kuvataan siten, että alin kerros on ensimmäinen kerros ja ylin kerros on seitsemäs kerros. Näin ollen datan vastaanottaminen verkosta tapahtuu alhaalta ylöspäin ja datan lähettäminen ylhäältä alaspäin. (Koudata 2010.)

Kerroksia on siis seitsemän, joista jokainen hoitaa omaa tehtävänsä:

1. Fyysinen kerros
2. Siirtokerros
3. Verkkokerros
4. Kuljetuskerros
5. Istuntokerros
6. Esitystapakerros
7. Sovelluskerros.

Kuten edellä mainittiin, jokainen kerros tekee yhteistyötä vain ylä- ja alapuolellaan olevien kerrosten kanssa, joten silloin tällöin mallia verrataan armeijan toimintamalliin. Esimerkkinä olkoon fyysinen kerros sotamiehet, siirtokerros aliupseeristo ja verkkokerros upseeristo. Tällöin sotamiesten ja

upseeriston välissä oleva aliupseeristo (siirtokerros) vastaanottaa komentoja yläpuoleltaan olevalta upseeristolta (verkkokerros) ja antaa omasta puolestaan komentoja alapuoleltaan oleville sotamiehille (fyysinen kerros). (Koudata 2010.)

TAULUKKO 1. OSI-mallin kerrokset tehtävineen (Koudata 2010)

Kerroksen nimi	Kerroksen tehtävä, esimerkkilaitteistot ja -protokollat
7. Sovelluskerros	Verkkoprosessien tarjonta sovelluksille. HTTP, FTP, SMTP.
6. Esitystapakerros	Datan esitysmuoto. JPG, PNG, GIF.
5. Istuntokerros	Tietokoneiden välinen kommunikaatio. AppleTalk, WinSock.
4. Kuljetuskerros	Päästä-päähän-yhteydet. TCP, UDP
3. Verkkokerros	Osoitteet ja reititys. IP, reititin.
2. Siirtokerros	Pääsy siirtotielle. ATM, Ethernet, kytkin.
1. Fyysinen kerros	Kaapelointi ja binäärikoodin lähetys verkkoon. Ethernet, hub, toistin.

Kuten ylläolevasta taulukosta (TAULUKKO 1) voidaan havaita, ensimmäinen kerros hoitaa lähinnä nimensä mukaisesti verkon fyysisiä asioita, kuten kaapelointia. Toinen kerros hoitaa sitä seikkaa, että kytkennöistä voidaan päästä Ethernet-verkon kautta siirtotielle. Kolmas kerros hoitaa IP-liikennettä (Internet Protocol) ja osoitteiden reititystä. Neljännen kerroksen tehtävä on pitää huoli siitä,

että yhteys alkupäästä loppupäähän on toimiva ja neljäs kerros hoitaa muun muassa TCP-protokollan (Transmission Control Protocol) sekä yhteydettömän UDP-protokollan (User Datagram Protocol) käyttöä. Viides kerros on tarkoitettu tietokoneiden välistä kommunikointia varten, eli viidennen kerroksen tehtävä on hoitaa AppleTalk- ja WinSock-yhteysprotokollien käyttöä. Kuudennen kerroksen tarkoitus on erilaisten medioiden, kuten esimerkiksi JPEG-pakattujen valokuvien esittäminen. Viimeinen kerros hoitaa erilaisten verkkoprotokollien tarjontaa ja käyttöä, kuten esimerkiksi SMTP-sähköpostiprotokollan (Simple Mail Transfer Protocol) käyttöä. (Koudata 2010.)

Kuten aiemmin mainittiin, voidaan kerroksia alhaalta ylöspäin tarkastellessa havainnoida kuinka datan vastaanottaminen lähtee fyysisestä kaapeloinnista ylöspäin aina sähköpostiprotokollaan saakka. Alaspäin tultaessa taas voitaisiin esimerkkinä kuvailla, että kun lähetetään esimerkiksi sähköpostiviesti, tullaan kerroksia alaspäin aina siihen asti, kunnes päästään takaisin verkkokaapelointiin, jossa kaikki tiedonsiirto loppujen lopuksi tapahtuu. (Koudata 2010.)

2.2 TCP/IP-malli

Kun tietoverkot alkoivat myöhemmin yleistyä, otettiin käyttöön TCP/IP-malli, joka on samanlainen kuin OSI-malli, mutta koostuu vain neljästä kerroksesta. Kolme ylintä OSI-mallin kerrosta yhdistettiin yhdeksi kerrokseksi ja niin tehtiin myös kahdelle alimmalle kerrokselle (KUVIO 1).



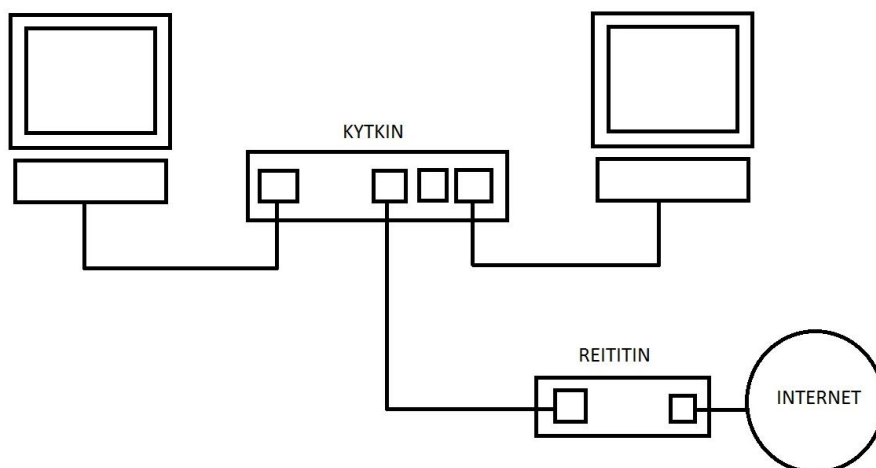
KUVIO 1. TCP/IP-mallin kerrokset

TCP on protokolla, joka vastaa datan hajottamisesta pienemmiksi paketeiksi ennen datan lähetystä. Kun datan lähetys onnistuu ja dataa vastaanotetaan, TCP myös kokoaa paketit takaisin alkuperäiseen muotoonsa. (W3Schools 2015b.)

IP on protokolla, joka vastaa tietokoneiden välisestä kommunikoinnista verkossa. Kommunikointi tapahtuu IP-osoitteiden välityksillä. IP myös huolehtii datapakettien siirtämisestä ja vastaanottamisesta. (W3Schools 2015b.)

2.3 Ethernet-verkko

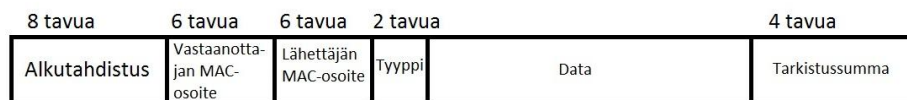
Ethernet on maailman yleisin käytössä oleva LAN-verkkotekniikka (Local Area Network). Tyypillinen Ethernet-verkko voi koostua esimerkiksi kahdesta tietokoneesta, jotka on kytketty kytkinlaitteeseen, josta lähtee kaapeli reitittimeen, joka reitittää tietokoneiden IP-osoitteet Internetiin (KUVIO 2). Ethernet-verkossa laitteet yhdistetään toiseensa kaapelilla, jota kutsutaan Ethernet-kaapeliksi.



KUVIO 2. Piirros tyypillisestä Ethernet-verkon kytkennästä

Ethernet on niin sanotusti kilpailuverkko, jossa vuoronvarausmenetelmänä on käytössä CSMA/CD (Carrier Sense Multiple Access/Collision Detection). Tämä menetelmä perustuu siihen, että tietokoneen verkkokortti kuuntelee, kulkeeko

Ethernet-kaapelissa tietoliikennettä. Kun liikennettä ei kulje, lähetetään matkaan Ethernet-kehys ja kuunnellaan, osuuko lähetetty kehys vastaantulevaan kehykseen. Mikäli ei tapahdu kehysten yhteentörmäystä, on kehyksen lähetys onnistunut. Jos kuitenkin törmäys tapahtuu, niin törmäyskohdassa syntyy jännitepiikki, joka heijastuu takaisin tietokoneen verkkokortille, jolloin verkkokortti tietää, että lähetys ei onnistunut. Tämän jälkeen verkkokortti lähettää signal quality error- eli signaalinlaaturiheviestin sekä heti viestin perään 32 bittiä satunnaista ”sotkua” (jam) toiseen päähän, jotta vastaanottava osapuoli myös huomaa, että lähetys epäonnistui. (Tectarget 2006.)



KUVIO 3. Ethernet-kehys

Ethernet-kehys (KUVIO 3) koostuu kuudesta eri segmentistä. Kehyksen ensimmäinen osuus sisältää kahdeksan tavun mittaisen alkutahdistuksen, minkä jälkeen kehyksestä ilmenee kuuden tavun pituiset vastaanottajan ja lähettäjän MAC-osoitteet (Media Access Control) eli verkkokorttien fyysiset osoitteet. Tämän jälkeen määritellään kehyksen tyyppi kahden tavun mittaisena, jonka jälkeen ilmenee itse kehyksen sisältämä data, joka voi olla 64-1518 tavua pitkä. Lopussa oleva segmentti on neljän tavun pituinen tarkistussumma eli ”checksum”, joka on lähettäjän laskema tarkistussumma lähetetystä kehyksestä, jota vastaanottaja vertaa, onko vastapuolen laskema tulos sama. (Tectarget 2006.)

Kehys toimii siten, että verkkokortti kuuntelee verkkoa koko ajan lukien samalla vastaanottajien MAC-osoitteita. Kun verkkokortti havaitsee oman MAC-osoitteensa, niin verkkokortti tietää, että sille on lähetetty kehys, jolloin kehys ladataan välittömästi koneeseen. (Tectarget 2006.)

3 PALVELIN JA KLUSTERI

Kun mobiilipeliyritys aikoo luoda pelin, jossa pelaajan aloittamat pelisessiot ja myös mahdolliset moninpelitiedot halutaan tallentaa pilvipalveluun, voidaan hyväksikäyttää pelipalvelinklusteria. Pelipalvelinkäytössä klusterilla tarkoitetaan usean tietokoneen yhteisöä, joista yleensä yksi toimii palvelimena, jolloin tietokoneyhteisölle komennetut tehtävät voidaan suorittaa paljon nopeammin kuin vain yhtä tietokonetta käytettäessä.

Usein palvelin luodaan myös kloonautuvaksi, jolloin klusteria voidaan tarvittaessa kasvattaa. Palvelimen kloonauksella voidaan saavuttaa entistä tehokkaampaa ja nopeampaa tehtävien suorituskykyä.

3.1 Palvelimen ominaisuudet

Palvelin eli serveri on palvelinkäyttöön valmistettu ja tarkoitettu tietokone, jolla on useita erilaisia käyttötarkoituksia (KUVIO 4). Palvelin voidaan toki pystyttää myös tavalliselle kotitietokoneelle, mikäli halutaan ylläpitää omaa pientä palvelinta esimerkiksi kotikäytössä. Palvelimelle voidaan tallentaa eri ohjelmien tietoja, sijoittaa esimerkiksi yrityskäytössä yrityksen asiakirjoja tai palvelimelle voidaan asentaa erilaisia sovelluksia, joita muut sovellukset voivat verkon kautta käyttää omassa toiminnassaan.



KUVIO 4. Esimerkki palvelinkaapista, johon on pinottu useita palvelintietokoneita (Server Computer 2014)

Palvelintietokone sisältää itsessään runsaasti kiintolevytilaa ja RAM-muistia (Random Access Memory) eli luku- ja kirjoitusmuistia, jotta tallennuskapasiteetti olisi riittävä ja ohjelmien suoritus olisi nopeaa ja kuormittumatonta. Tehokasta näytönohjainta palvelimessa ei vaadita eikä tarvita, koska näytönohjaimella ei ole palvelimen toiminnan kannalta merkitystä. (Saraibeam 2015.)

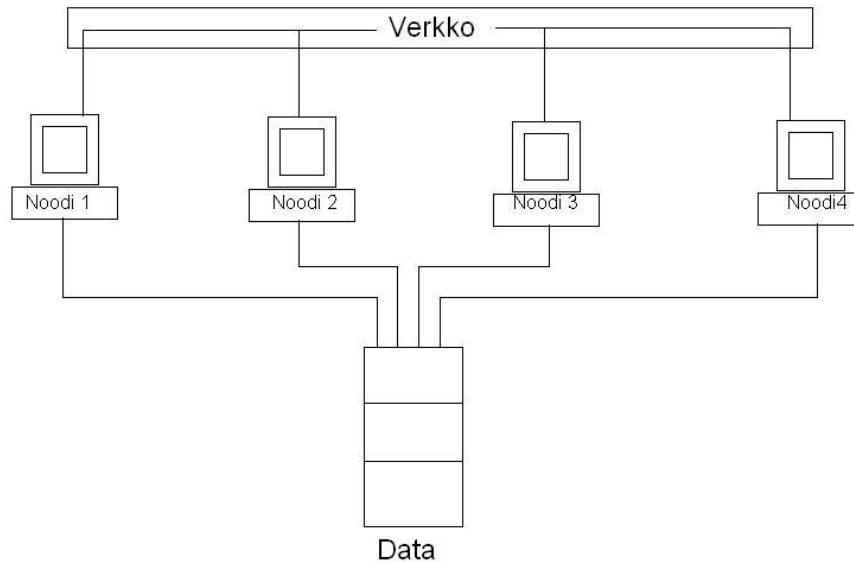
Internet on rakennettu useista toisiinsa yhdistetyistä palvelintietokoneista. Esimerkiksi, kun Internet-käyttäjä haluaa lukea sähköpostiaan, niin tällöin sähköpostisovellus tai selain, mikäli sähköpostia luetaan selaimella, ottaa yhteyden palvelimen sisälle asennettuun palvelinohjelmistoon, jolloin ohjelmisto lähettää palvelupyynnön, että esimerkiksi jokin tietty saapunut sähköpostiviesti halutaan lukea. Tämän jälkeen pyyntö vastaanotetaan ja palautetaan, eli avattavaksi valittu sähköpostiviesti aukeaa ja se voidaan tällöin lukea. Toisena esimerkkinä voidaan ottaa Internetissä ”surffaaminen”, joka tarkoittaa sitä, että eri Internet-sivustojen linkkejä valittaessa käyttäjä voi loikkia palvelimelta toiselle. Linkkiä painamalla voi siis siirtyä palvelimelle, joka sijaitsee aivan eri puolella maailmaa, kuin se sivu, josta linkki alunperin avattiin. (Internetopas 2014.)

Palvelimia voidaan luoda myös virtuaalisena. Tällöin fyysistä palvelintietokonetta ei vaadita, vaan luotu palvelin sijaitsee jossain päin maailmaa palveluntarjoajalta hankitun palvelun kautta verkkokiintolevyllä, eli palvelinta käytetään pilvipalvelun kautta.

3.2 Klusterin ominaisuudet

Klusteri on yleisesti käytettävä termi joukkiosta yhdenmukaisia subjekteja, joka voi kasvaa ja tehdä näin ollen tehokkaampaa työskentelyä. Tietotekniikassa ja asianmukaisesti palvelintekniikassa klusterilla pääsääntöisesti tarkoitetaan usean tietokoneen eli noodin tai suomalaisittain solmun ryhmittymää. Klusterin noodit voivat suorittaa useita prosesseja yhdessä saavuttaen näin runsaamman laskentatehon. Mikäli jossakin klusterin laitteessa syntyy vikatilanne, voidaan laitteen tehtävät ohjata muille klusteriin kuuluville laitteille, jotka hoitavat vioittuneen yksilön tehtäviä, kunnes vikat on korjattu. Tämä mahdollistaa sen, että palvelimen käyttäjillä on aina mahdollisuus käyttää palveluitansa, vaikka jokin osa-alue klusterista olisikin vioittunut.

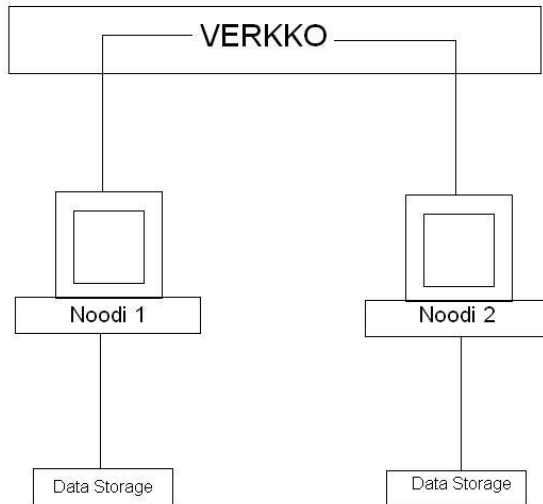
Klusterin käyttö on hyödyllisimmillään sovelluksissa, jotka ovat jatkuvasti käynnissä ja käytössä. Tällaisia sovelluksia ovat esimerkiksi verkkopelit sekä tiedostopalvelimet. (Microsoft Technet 2015.)



KUVIO 5. Piirroskuvio peruskluusterin kytkennästä (Microsoft Technet 2015)

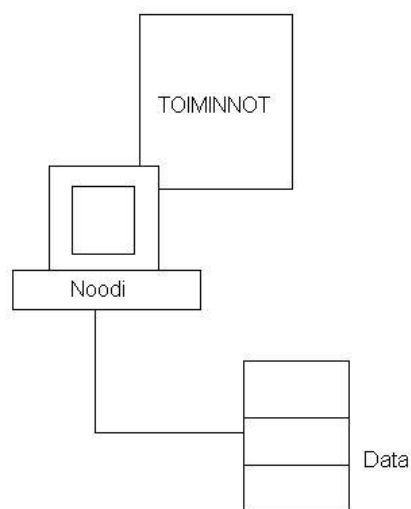
Yleisimmin käytetty klusterityyppi on ”Single Quorum Device Cluster” eli yhden päätäntävallan omaava klusteri (KUVIO 5). Tämän tyyppinen klusteri koostuu useasta noodista, jotka on kytketty yhteen tai useampaan kovalevyjoukkioon. Kovalevyjoukkio on useaan segmenttiin jaettu ja joukkio on koko klusterin yhteinen. Joukkio toimii niin sanottuna liityntälaitteena palvelimien välillä ja sisältää myös klusterin ”quorum”- eli päätäntävaltatiedot. Yksi levyjoukkio on ainoastaan yhden palvelimen hallinnassa. (Microsoft Technet 2015.)

Eräs toinen klusterityyppi on ”Majority Node Set Cluster” (KUVIO 6) eli enemmistönoodiklusteri. Tämän tyyppisessä klusterissa jokaisella noodilla on oma tiedontallennuspaikkansa eikä yhteistä kovalevyjoukkiota, kuten ”Single Quorum Device” -klusterissa. Jokaisen noodin oma tallennuspaikka sisältää klusterin ”quorum”-datan sekä klusterin konfiguraatitiedot. Etuna ”Majority Node Set” -klusterissa on, että vaikka käyttäjä irroittaisi jonkin noodin tallennuspaikan, voidaan klusterin käyttöä silti jatkaa.



KUVIO 6. Majority Node Set -klusterin kytkentäkuvio (Microsoft Technet 2015)

Myös klusterien kokeilukäyttöön on olemassa oma klusterimallinsa. Tämän tyyppistä klusteria kutsutaan ”Local Quorum”- tai ”Single Node”- eli yhden noodin klusteriksi. Nimensä mukaisesti klusteri sisältää vain yhden noodin, joka on kytketty klusteroimattomaan kovalevyjoukkioon. Yhden noodin klusterilla voidaan kokeilla isomman klusterin yhden osa-alueen toimintaa ja toimintoja. (Microsoft Technet 2015.)



KUVIO 7. Local Quorum Cluster (Microsoft Technet 2015)

Palvelimen tavoin myös klusteri voidaan luoda virtuaalisena. Tämä on hyvin kannattava ratkaisu perinteisen mallin sijaan, sillä useasta tietokoneesta muodostava joukkio on usein paljon tilaa vievä ratkaisu.

3.3 Palvelimelle asennettavien sovellusten perusteet

Palvelinta luodessa yksi tärkeimmistä asioista on suunnitella, mitä sovelluksia palvelimelle tulisi alussa asentaa, jotta haluttu toimivuus ja palvelimen käyttötarkoituksen hyödyntäminen olisi täysin taattua. Tällaisia aluksi asennettavia perussovelluksia ovat esimerkiksi sellaiset sovellukset, jotka mahdollistavat palvelimen ymmärtämään erilaisia internetprotokollia sekä, etenkin kun palvelin luodaan peli- ja ohjelmakäyttöön, erilaisia ohjelmointikieliä.

Palvelinohjelmistoja ja -protokollia on olemassa lukematon määrä erilaisia käyttötarkoituksia varten. Koska yleisimmin palvelimet ovat Linux-pohjaisia, keskitytään seuraavassa siihen, millaisia sovelluksia Linux-palvelimelle voidaan asentaa.

3.3.1 MySQL

Tietokanta eli data base on tiedon varastoinnista käytetty termi tietotekniikassa. Tietyn kohteen tiedot voidaan koota yhteen kokoelmaan, joita tietojärjestelmät voivat käyttää omiin tarpeisiinsa sekä muokata niitä. Relaatiolla tarkoitetaan sitä, kun tietokannassa olevien tietojen välille on muodostettu erilaisia suhteita tai yhteyksiä. Kokoelmaa, joka koostuu tiedoista, joilla on jokin luotu yhteys toisiinsa, kutsutaan tauluksi.

MySQL on Oracle Corporationin kehittämä ja maailman yleisimmässä käytössä oleva relaatiotietokantaohjelmisto, joka käyttää kyselykielenään perinteistä SQL:ää (Structured Query Language). MySQL on todella helppokäyttöinen ja soveltuvainen niin pienten kuin suurtenkin sovellusten käytettäväksi. (W3Schools 2015a.)

Palvelinkäytössä MySQL:ää tarvitaan tietojen tallentamista varten tietokantaan. MySQL:n suosio palvelinkäytössä perustuu ilmaiseen lisenssiin.

3.3.2 PHP

PHP eli PHP: Hypertext Preprocessor on ohjelmointikieli, joka soveltuu web-sivustojen luontiin, jossa PHP-kieltä tavanomaisesti käytetään. PHP:tä ei ole kovinkaan vaikea oppia, sillä PHP on hyvin samankaltainen kuin C-, Java- ja Perl-ohjelmointikielien. Kuten monet muut kielet, myös PHP on täysin ilmainen mihin tahansa käyttötarkoitukseen, eli käyttöön ei tarvita minkäänlaista lisenssiä ja näin ollen PHP on avoimen lähdekoodin kieli. (Jyväskylän yliopisto 2015.)

Palvelimelle asennettaessa PHP:tä tarvitaan siinä tapauksessa, kun yksinkertaisesti halutaan ajaa sellaista koodia, joka on kirjoitettu PHP-ohjelmointikielillä. PHP:n asennus on siis todella tarpeellista, sillä noin 95 % web-sovelluksista on kirjoitettu PHP:tä käyttäen.

3.3.3 Apache

Apache on ASF:n eli Apache Software Foundationin luoma HTTP-palvelinsovellus ja maailman suosituin sellainen. ASF on voittoa tavoittelematon yritys, joten Apache on tällöin ilmainen ja lähdekoodikin on vapaa.

Palvelinkäytössä Apache hoitaa tehtävänänsä HTTP-protokollan käyttöä. Tällöin palvelimella voidaan käyttää HTTP-pohjaisia sovelluksia ja palvelin kykenee toimimaan HTTP-pohjan kanssa sekä HTTP-pohjaiset Internet-sivut voidaan avata käyttäjän niitä pyytäessä. (Apache 2015.)

3.3.4 phpMyAdmin

phpMyAdmin on ilmainen työkalu, jossa yhdistyy teoriassa kaksi ohjelmointikieltä: PHP ja MySQL. Työkalu on ohjelmoitu PHP:llä.

Palvelimelle phpMyAdmin asennetaan yleensä siksi, että palvelimella voidaan hoitaa MySQL-tietokannan valvontaa ja hallintaa. Työkalu tukee MySQL:n lisäksi myös muita tietokantoja, kuten MariaDB:tä. (phpMyAdmin 2015.)

3.3.5 Python

Python on ilmaisen lähdekoodin ohjelmointikieli, joka on yksinkertainen ja ensikertalainenkin ohjelmoija oppii Pythonin alkeet nopeasti. Python on hyvin olio-ohjelmointiin soveltuva, joten käyttö soveltuu esimerkiksi peliohjelmointiin. Python on myös useiden eri kielien kanssa yhteensopiva ja toimii missä tahansa ympäristössä. (Python 2015.)

Python asennetaan palvelimelle, jotta palvelin osaisi käyttää Pythonilla luotuja sovelluksia ja palvelin kykenisi käyttämään Pythonia web-pohjalta.

Pelipalvelimella Python on välttämätön sovellus.

3.4 Bash-käskyt

Linux-palvelimen pystyttämisen ja käyttöönoton tärkeimpiä alkeita on opetella käyttämään käskyjä. Bash eli Bourne Again SHell on komentotulkki (shell), jota käytetään useissa eri käyttöjärjestelmissä, kuten esimerkiksi Linuxissa ja Windowsissa. Bash toimii vakiokomentotulkkina Mac OS -käyttöjärjestelmässä.

Kun bash-käskyjä halutaan päästä syöttämään komentoriville graafisessa käyttöjärjestelmässä, tulee ensin valita jokin terminaaliemulaattori eli päätesovellus, kuten esimerkiksi xterm. Monesti käyttöjärjestelmässä on valmiina jokin päätesovellus, mutta niitä saa ladattua myös Internetistä usein ilmaiseksi.

Bash-käskyt ovat jokseenkin samankaltaisia komentoja, kuin ”Microsoft Disk Operating Systemin” eli MS-DOS-käyttöjärjestelmän komennot. Esimerkiksi kansioiden vaihdossa käytetään komentoa cd eli ”change directory” niin bashissa kuin DOSsissakin.

Bash-käskyihin siirtyminen suoraan MS-DOS-käskyjen pohjalta voi olla aluksi hankalaa. Internet on täynnä erilaisia ohjeita, kuinka opetella käyttämään Linuxia ja bash-käskyjä. Itse bash-käskyjä on olemassa satoja. Alla olevassa taulukossa on esitelty muutamia yleisimpiä peruskomentoja (TAULUKKO 2).

TAULUKKO 2. Yleisiä bash-käskyjä (SS64 2015)

<u>Komento</u>	<u>Tehtävä</u>
cd	”Change directory”, hakemiston vaihtaminen.
sudo	Käynnistää ohjelman superkäyttäjän oikeuksilla.
ls	Listaa senhetkisen hakemiston tiedostot.
cp	Kopioi tiedoston.
mv	Tiedoston siirto/nimen muuttaminen.
rm	Tiedoston poistaminen.
mkdir	Uuden hakemiston luonti.

4 VIRTUALISOINTI JA PILVIPALVELUT

Nykyään palvelujen hankkiminen virtuaalisena fyysisten asioiden sijaan on yleistynyt maailmassa. Niin sanottuja virtuaalisia ”leasing”- eli vuokratietokoneita hankitaan palvelinkäyttöön palveluntarjoajilta yhä enemmän, jolloin toimistotilaa vievien fyysisten palvelintietokoneiden hankinta voidaan sulkea pois ja palvelin olisi pilvipalvelun sisällä käytössä. Tällöin myös säästetään energiaa, rahaa sekä elektroniikkajäte vähenee. Pilvipalveluna on mahdollista käyttää myös ohjelmistoja.

Palveluntarjoajat tarjoavat erilaisia räätälöityjä virtualisoituja palveluita. Palveluita voidaan aina myös muokata, mikäli jokin seikka sopisi omaan tarpeeseen paremmin. Suosituimpia palveluntarjoajia ovat muun muassa Amazon, Windows Azure sekä Google.

4.1 SaaS

SaaS (Software as a Service) on pilvipalvelu, jonka avulla voidaan hankkia jokin ohjelmisto, joka toimii pilven sisällä verkossa sen sijaan, että hankittaisiin lisenssipohjainen sovellus. SaaS-sovellusta voidaan käyttää Internet-selaimen kautta toimivalla ”client”- eli asiakasohjelmistolla. (Salesforce 2015.)

SaaS-ohjelmia hyödynnetään yleisesti taloushallintoalalla, esimerkiksi tilien hallinnassa, resurssisuunnittelussa ja asiakassuhteiden hallinnassa. Yhä useampi yritys hyödyntää nykyään SaaS-sovelluksia. (Salesforce 2015.)

4.2 PaaS

PaaS (Platform as a Service) tarkoittaa ulkoistettua palvelualustaa. Tällä tarkoitetaan palvelualustaa, jota käyttäjät voivat itse parantaa sovelluskehitysohjelmin avulla ja lisäksi lisätä omatekemiään ohjelmia osaksi palvelualustaa. Koska käyttäjät voivat itse kehittää palvelualustaa itselleen sopivaksi, ovat yhteensopivuusongelmat mennyttä aikaa skaalautuvuuden vuoksi. (Interoute 2015.)

Pilvipalveluna hankitun palvelualustan käyttäjä saa yleensä käyttöönsä valmiita kehitystyökaluja ja -kirjastoja, joiden avulla omien sovellusten luonti osaksi palvelualustaa on vaivatonta. PaaS soveltuu siis hyvin esimerkiksi ohjelmoijien käyttöön. (Interoute 2015.)

4.3 IaaS

IaaS (Infrastructure as a Service) tarkoittaa sitä, että jokin palveluntarjoaja vuokraa asiakkailleen pilvipalveluna virtuaalitetokoneita. Tällöin esimerkiksi voidaan Windows-käyttöjärjestelmän omaavalla tietokoneella vuokrata virtuaalitetokone ja siihen voidaan asentaa vaikkapa Linux-käyttöjärjestelmä ohjelmiseen. (Techtarget 2015.)

Pilvipalvelun sisäistä tallennustilaa voidaan myös tarvittaessa kasvattaa, mikäli tila pääsee loppumaan. Tallennustilaa voidaan varata muutamasta gigatavusta jopa useaan teratavuun, joka riittää todella hyvin esimerkiksi tietokantojen tallentamiseen. Lisämaksusta voidaan hankkia entistä enemmän tallennustilaa, mikäli tarve niin vaatii. Esimerkiksi tiedostopalvelimella tarvitaan suurta tallennuskapasiteettia. (Techtarget 2015.)

5 PELIPALVELINKLUSTERI MOBIILPELIYRITYKSEN KÄYTTÖÖN

Työn suunnittelu aloitettiin vuoden 2013 loppupuolella, jolloin pidettiin pieniä palaverejä aiheesta, ja itse työ aloitettiin tammikuussa 2014. Tavoitteena oli luoda vakaa palvelinpohja, jota yrityksen työntekijät pystyisivät tarvittaessa muokkamaan, sekä kapasiteettivaatimusten kasvaessa, myös kloonamaan eli monistamaan helposti.

Pelipalvelinkäyttöön tarkoitetun klusterin luonnissa on aluksi tärkeää paneutua kunnolliseen suunnitteluun. Erilaisia ratkaisuja mietittäessä tulee ottaa huomioon yrityksen tarpeet sekä klusterin ja palvelimen tulevaisuus: kuinka esimerkiksi voidaan kasvattaa palvelimen tallennustilaa tai kuinka klusteri voidaan kloonata, kun suoritustehon tarve kasvaa?

5.1 Pelipalvelinklusterin suunnittelu

Työ aloitettiin suunnittelemalla, mitä erilaisia vaihtoehtoja voitaisiin käyttää klusterin toteutuksessa. Vaihtoehtoja oli useita, mutta koska yritys oli sillä hetkellä vasta aloittanut, tuli miettiä, mikä olisi edullisin ja järkevin ratkaisu. Mikäli klusteri toteutettaisiin fyysisenä, jouduttaisiin jostain kohtaa toimistoa varaamaan ylimääräistä tilaa usean tietokoneen joukkioille ja lisäksi hankkimaan vähintään neljä uutta tietokonetta. Koska toimitilat ja budjetti olivat tässä vaiheessa pienet, alettiin selvittämään muita ratkaisuja.

Fyysisen klusterin sijaan pelipalvelinklusteri on mahdollista toteuttaa täysin virtuaalisena pilvipalvelun avulla. Tämä tarkoittaa sitä, että yritys hankkii joltakin pilvipalveluntarjoajalta itselleen pilvipalvelun, johon voidaan luoda täysin virtuaalisia tietokoneita, joita voidaan käyttää palvelimina. Fyysisten laitteiden hankinta, energiankulutus ja tilan vienti voitaisiin samalla poissulkea, kun virtuaalitietokoneet sijaitsevat jossain päin maailmaa palveluntarjoajan palvelimilla.

Sekä fyysisellä että virtuaalisella klusterilla on molemmilla hyvät ja huonot puolensa. Toisaalta, mikäli yritys ei itse ole palveluntarjoaja eli ylläpidä omia fyysisiä palvelimiaan, joita muut voivat vuokrata virtuaalikäyttöön, on virtualisointi huomattavasti parempi ratkaisu.

TAULUKKO 3. Fyysisen ja virtuaalisen klusterin vertailu

<u>Verrattava seikka</u>	<u>Fyysinen</u>	<u>Virtuaalinen</u>
Hinta	Kalliit hankintakulut. Tietokoneita täytyy hankkia useampi.	Palvelun käyttöönotto useilla palveluntarjoajilla jopa ilmaista.
Energiankulutus	Kasvattaa yrityksen energiakustannuksia, kun palvelimia pidetään käynnissä jatkuvasti.	Ei kasvavaa energiankulutusta.
Yhteensopivuus	Päivittäminen täytyy hoitaa kokonaan itse.	Skaalautuu uusia standardeja ja päivityksiä varten palveluntarjoajan toimesta.
Palvelimen sijainti	Palvelimen sijainti omassa toimistossa.	Sijainti usein toisessa maassa.
Tilan vienti	Vie paljon tilaa, mikäli toimiston tilat ovat pienet.	Ei vie ollenkaan toimistotilaa.

Yllä mainittujen seikkojen perusteella (TAULUKKO 3) klusteri päädyttiin toteuttamaan virtuaalisena, sillä virtualisointi soveltui parhaiten yrityksen senhetkiseen tilanteeseen. Kuten aikaisemmin mainittiin, tilaa oli sillä hetkellä vähän ja yritys oli juuri perustettu, joten budjetti oli hyvin pieni. Yrityksen varoja säästyisi enemmän pelien kehittämistä varten, kun fyysisten palvelintietokoneiden hankintaa ei tarvittaisi ja energiakustannukset eivät kasvaisi normaalia suuremmiksi.

5.2 Pilvipalveluntarjoajat

Kun virtualisoitu ratkaisu oli päätetty, tuli seuraavaksi valita jokin pilvipalveluntarjoaja. Vaihtoehtoja on useita, joten tuli tarkasti tutkia, minkä palveluntarjoajan pilvipalvelu olisi sopivin pelipalvelinkäyttöön.

Koska kyseessä on ”leasing”- eli vuokrapalvelintietokone, noudatetaan tällöin IaaS-pilvipalvelumallia. Palvelimen käyttöjärjestelmäksi valittiin Linuxin Ubuntu, koska Linuxin käyttäminen on tuttua lähes jokaiselle tietotekniikka-alan työntekijälle. Linux on lisäksi ilmainen käyttöjärjestelmä.

5.2.1 Amazon Web Services

Amazon Web Services eli AWS tarjoaa kattavan valikoiman erilaisia pilvipalveluita aina mobiilisovelluksista yritysohjelmistoihin. Asiakkaalle asetettu verkkotallennustila mahdollistaa niin omien tietokantojen luomisen kuin virtuaalikoneen käyttämisen.

Amazoniin luodaan yleensä käyttäjän toimesta erilaisia palvelininstansseja, joita tämän palveluntarjoajan tapauksessa kutsutaan EC2-instansseiksi (Elastic Compute Cloud). Luotua instanssia voidaan käyttää esimerkiksi oman palvelimen ylläpitoon. Muun muassa Netflix-videontoistopalvelu käyttää Amazonin EC2-instansseja.

Amazon Web Services tukee IaaS- ja SaaS-tyyppisiä pilvipalveluita. Ohjelmistoalustan luominen eli PaaS-pilvipalvelu ei ole mahdollista. Myös monia MMO-pelejä (Massively Multiplayer Online) eli Internetissä pelattavia massamoninpelejä pelataan AWS:n pilvipalvelun välityksellä.

Amazon Web Servicesin käyttö on ilmaista vuoden ajan Amazonin asettaman ”Free Usage Tierin” eli vapaan käyttöasteen rajoissa. Mikäli käyttäjä haluaa itselleen lisäominaisuuksia, voidaan hankkia maksullisen käyttöoikeuden itselleen. (Amazon Web Services 2015b.)

5.2.2 Windows Azure

Microsoftin Windows Azure tarjoaa käyttäjilleen IaaS-, PaaS- ja SaaS-palveluita. Virtuaalitietokoneiden vuokraus, omien Internet-sivujen tekeminen ja esimerkiksi videoiden suoratoisto ovat siis mahdollisia. Käyttäjä voi luoda pilven sisään myös omatekoisia ohjelmia, joita voidaan skaalata eli muuntaa helposti eri käyttäjien tarpeita varten. Azure on hyvä vaihtoehto niille käyttäjille, jotka haluavat palvelun, jossa kaikki pilvipalvelun toiminnot ovat saatavilla.

Windows Azuren virtuaalitietokonevuokra on maksullinen ja hinta mitataan käyttötunneissa. Azurea on kuitenkin mahdollista käyttää ilmaiseksi kokeiluversiona 30 päivän ajan. (Windows Azure 2015.)

5.2.3 Google Cloud

Myös hakukoneestaan tunnettu Google tarjoaa nykyään erilaisia pilvipalveluita. Google tarjoaa käyttäjilleen omien ohjelmien luontimahdollisuutta sekä omien tiedostojen tallentamista pilvipalveluun. Googlen Cloud Platformilla käyttäjä voi luoda omia Internet-sivuja sekä omia sovelluksia, jotka niin ikään on mahdollista skaalata eri käyttäjiä varten. Käytettävissä ovat siis IaaS- ja PaaS-palvelut.

Google Cloudia voidaan käyttää ilmaiseksi 30 päivän ajan niin sanottuna ”trial”- eli kokeiluversiona. Kokeiluversio sisältää kaikki ominaisuudet, kuin maksullisenkin versio. Mikäli käyttäjä on Google Cloudiin tyytyväinen, voi hän hankkia maksullisen käyttöoikeuden itselleen. (Google Cloud 2015.)

5.2.4 Palveluntarjoajien vertailu ja valinta

Palveluntarjoajavaihtoehtoja on useita, mutta valintatilanteessa päädyttiin valitsemaan kolme eri vaihtoehtoa eli Amazon Web Services, Windows Azure sekä Google Cloud. Valintakriteereinä olivat virtuaalitietokoneen vuokrausmahdollisuus, yhteensopivuus massamoninpelikäytön kanssa sekä hinta.

Jokaisella vaihtoehdolla oli omat hyvät puolensa. Alla olevasta taulukosta (TAULUKKO 4) voidaan havaita työn kannalta tärkeimpien seikkojen vertailu jokaisen vaihtoehdon välillä.

TAULUKKO 4. Palveluntarjoajien vertailu

<u>Verrattava seikka</u>	<u>Amazon Web Services</u>	<u>Windows Azure</u>	<u>Google Cloud</u>
IaaS (virtuaalitetokoneen vuokraus)	On mahdollinen.	On mahdollinen.	On mahdollinen.
SaaS (sovellusten käyttö verkon kautta)	On mahdollinen.	On mahdollinen.	Ei ole mahdollinen.
PaaS (sovellusalustan luonti)	Ei ole mahdollinen.	On mahdollinen.	On mahdollinen.
Käyttö massamonipelissä	On mahdollinen. Käytetään paljon ympäri maailman.	On mahdollinen.	On mahdollinen.
Hinta	Voidaan käyttää ilmaiseksi ”Free Tierin” rajojen sisällä vuoden ajan.	Yhden kuukauden ilmainen kokeilujakso.	Yhden kuukauden ilmainen kokeilujakso.

Parhaaksi vaihtoehdoksi yrityksen tarpeisiin valikoitui Amazon Web Services. AWS:ää voi käyttää vuoden ajan ilmaiseksi ja AWS on suosittu verkossa pelattavien massamonipelien käytössä. Sovelluskehitysmahdollisuuden puuttuminen ei koitunut ongelmaksi.

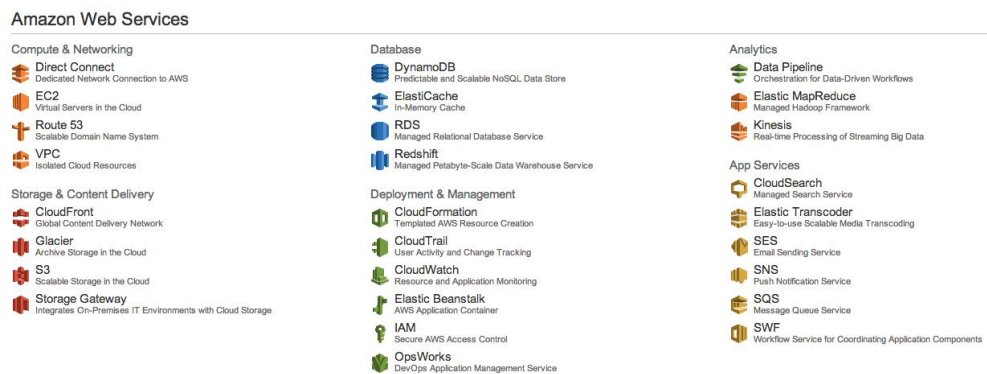
5.3 Palvelun hankkiminen Amazonilta

Amazon Web Servicesin palvelut saa hankittua helposti AWS:n kotisivuilta (<http://aws.amazon.com>) aluksi ilmaisena, josta löytyvät kaikki tarvittavat toiminnot pienen palvelimen ylläpitoon, mutta halutessaan lisäominaisuuksista voi myös maksaa, jolloin palvelimelle saa muun muassa enemmän tallennuskapasiteettia. Palvelun hankintaan vaadittiin ainoastaan yleiset perustiedot, kuten oma ja yrityksen nimi sekä luottokortti, jonka omistajan tulee olla samanniminen kuin palvelun hankkijan. Tässä tapauksessa palvelu otettiin Digital Hammerin toimitusjohtajan nimiin.

Huomioitavaa eurooppalaisilla AWS:n käyttäjillä on siinä, että mikäli haluaa palvelimensa sijaitsevan Euroopassa, tulee sijainniksi valita Irlanti, sillä AWS:n Euroopan pääpalvelin sijaitsee Irlannissa. Sijainti valitaan tilin luontivaiheessa.

5.4 EC2-instanssin luonti

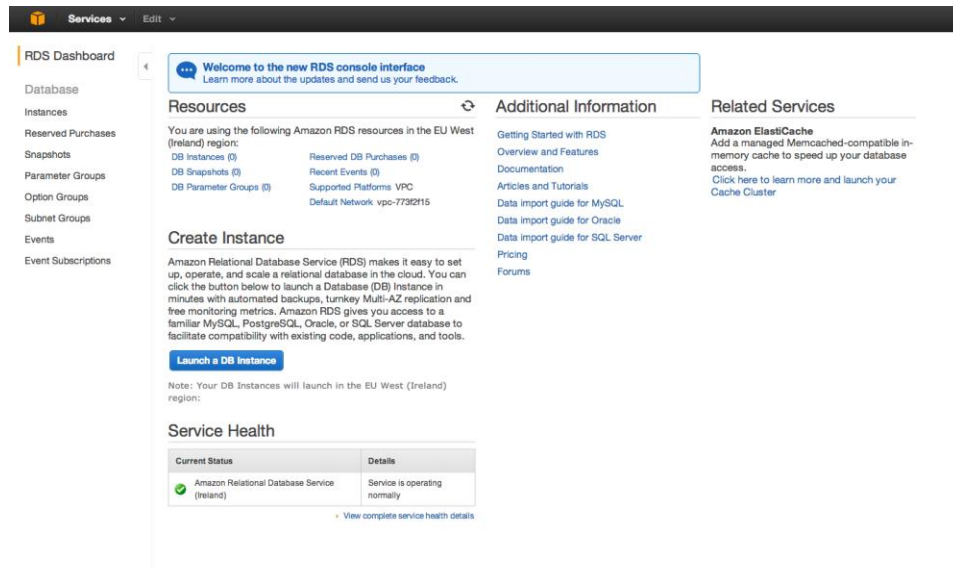
Kun palvelu oli hankittu ja päästiin Amazon Web Servicesin ohjauspaneeliin käsiksi, voitiin aloittaa virtuaalipalvelimen luonti. Ensimmäisenä tuli luoda itse palvelimen pohja, eli Amazonin tapauksessa EC2-instanssi (Elastic Compute Cloud).



KUVIO 8. Näkymä Amazon Web Servicesin ohjauspaneelistä (Amazon Web Services 2014)

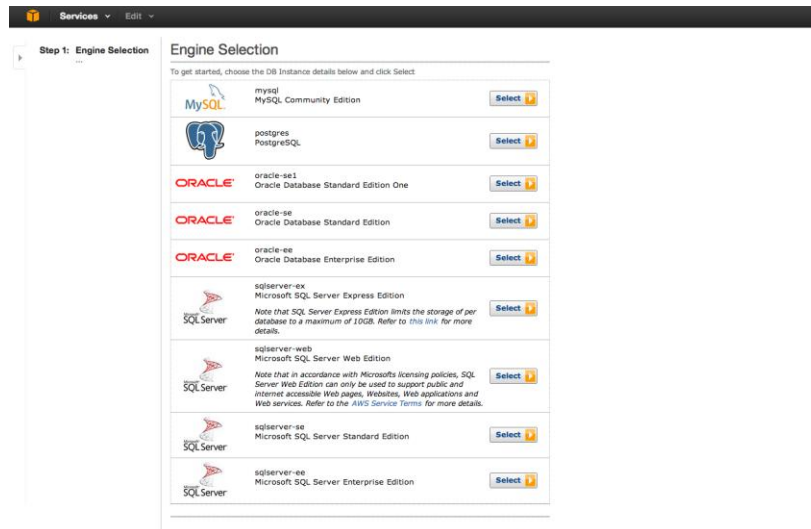
Koska tarkoituksena oli luoda Linuxin Ubuntu-pohjainen palvelin moninpelidatan tallentamista varten, päätettiin luoda tietokanta- eli MySQL-pohjainen palvelin.

Luominen aloitettiin valitsemalla ohjauspaneelin ”Database”-valikon alta kohta RDS eli ”Relational Database Service” (KUVIO 8). Tämän valikon alta siirrytään ”RDS Dashboardiin” eli relaatiotietokantapalvelun kojelautaan, jossa voidaan luoda uusia sekä hallita vanhoja jo olemassa olevia tietokantapohjaisia virtuaalipalvelimia pilvipalvelun sisällä.



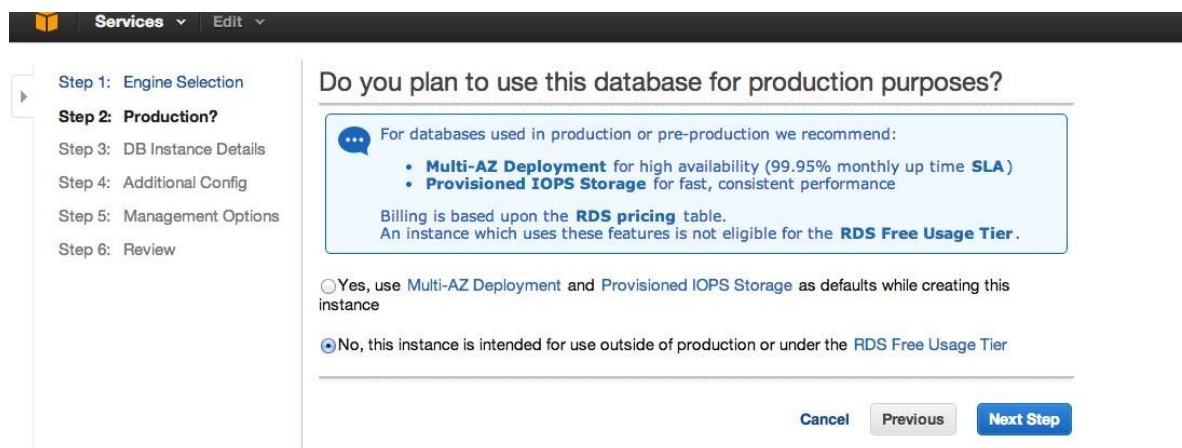
KUVIO 9. ”RDS Dashboard”, josta voidaan tarkastella muun muassa käynnissä olevia instansseja sekä luoda uusia (Amazon Web Services 2014)

”RDS Dashboardia” tutkittaessa voidaan havaita, että kojelauta ilmoittaa myös ”Service Health” –kohdassa palvelun senhetkisen kunnon, eli jos esimerkiksi palveluntarjoajan palvelu on huoltokatkolla tai palvelu on muuten vaan poissa käytöstä. Uutta instanssia päästiin luomaan valitsemalla ”Launch a DB Instance”. (KUVIO 9.)



KUVIO 10. ”Enginen” eli palvelimen pohjalla käyvän ohjelmistomoottorin valinta (Amazon Web Services 2014)

Palvelimen luonti suoritettiin askeleittain (step), joissa tuli tehdä tiettyjä valintoja sen mukaan, millä tavalla palvelimen haluttiin toimivan. Ensimmäisessä askeleessa tuli valita, mitä ohjelmaa palvelimen pohjalla käytetään. Kuten aikaisemmin mainittiin, kyseessä on tietokantapohjainen palvelin, joten ohjelmaksi valittiin listan ylin vaihtoehto eli MySQL (KUVIO 10). Muina vaihtoehtoina on saatavilla muun muassa Java-kielen luoneen Oraclen tietokantapalveluita sekä erilaisia SQL-kyselykielen tietokantapalveluiden variaatiota.



KUVIO 11. Kysely, jossa kysytään, onko instanssin käyttötarkoitus tuotannollinen (Amazon Web Services 2014)

Koska Amazon RDS:än käyttö on tiettyyn rajaan asti ilmaista, seuraavassa askeleessa kysytään, onko palvelininstanssin käyttötarkoitus tuotannollinen vai ei. Koska tässä kyseisessä työssä tarkoitus ei ollut tuotannollinen ja näin ollen pelkät peruspalvelut riittivät, jotka kuuluvat RDS:n ”Free Usage Tieriin” eli vapaan käytön tasoon, valittiin alempi vaihtoehto, joka ei sisällä kehitystyökaluja eikä erillisiä tallennusvarastoja (KUVIO 11). Mikäli tulevaisuudessa halutaan ostaa lisäpalveluita, on mahdollista muuttaa kyseistä valintaa.

KUVIO 12. Tietokantainstanssin yksityiskohdat (Amazon Web Services 2014)

Käyttötarkoituksen valinnan jälkeen päästiin luotavan instanssin yksityiskohtien muokkaukseen. Koska instanssin pohjalla käynnissä oleva tietokantamoottori oli jo aikaisemmin valittu MySQL:ksi ja moottorin alapuolella olevat valinnat tulevat tällöin automaattisesti, voitiin suoraan määritellä instanssin tarvitsema tallennustila. Tallennustilaa voidaan valita 5 gigatavusta aina 3072 gigatavuun asti. Päätettiin, että minimimäärä tallennustilaa riittää, sillä kyseessä on vain muutamien tavujen kokoisista tiedoista, joita palvelimelle varastoitaisiin. Valintaa on toki mahdollista muuttaa myös jälkikäteen suuremmaksi, mikäli tarve niin vaatii.

”DB Instance Identifier”- eli tietokantainstanssin tunniste -kohdassa määritettiin vielä tulevan instanssin nimi. Alemmassa kohdassa voitiin määrittää käyttäjätunnus ja salasana, joilla instanssiin voidaan kirjautua. (KUVIO 12)

Services Edit

Step 1: Engine Selection
Step 2: Production?
Step 3: DB Instance Details
Step 4: Additional Config
Step 5: Management Options
Step 6: Review

Additional Config

Provide the optional additional configuration details below.

Database Name: digitalhammer (e.g. mydb)

Note: If no database name is specified then no initial MySQL database will be created on the DB Instance.

Database Port: 3306

Choose a VPC: Default VPC (vpc-773f2f15)

DB Subnet Group: default

Publicly Accessible: Yes No

Availability Zone: No Preference MultiAZ deployment selected

Option Group: default.mysql-5-6

If you have custom DB Parameter Groups or DB Security Groups you would like to associate with this DB Instance, select them below, otherwise proceed with default settings.

Parameter Group: default.mysql5.6

VPC Security Group(s): launch-wizard-3 (VPC)
launch-wizard-1 (VPC)
default (VPC)
launch-wizard-2 (VPC)

Cancel Previous **Next Step**

KUVIO 13. Lisäasetukset (Amazon Web Services 2014)

Neljännessä askeleessa tuli määrittellä vapaavalintaisia lisäasetuksia. Näissä lisäasetuksissa voidaan määrittellä, mikä pohjalle luotavan tietokannan nimeksi tulee. Jos nimeä ei anneta, ei myöskään tietokantaa tällöin luoda. Päädettiin käyttämään AWS:n tarjoamia vakioasetuksia, koska ne soveltuivat tarpeisiin parhaiten. (KUVIO 13.)

Services Edit

Step 1: Engine Selection
Step 2: Production?
Step 3: DB Instance Details
Step 4: Additional Config
Step 5: Management Options
Step 6: Review

Management Options

Enabled Automatic Backups: Yes No

The number of days for which automated backups are retained.

Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail here.

Backup Retention Period: 1 days

The daily time range during which automated backups are created if automated backups are enabled

Backup Window: Select Window No Preference

Start Time 00 : 00 UTC
Duration 0.5 hours

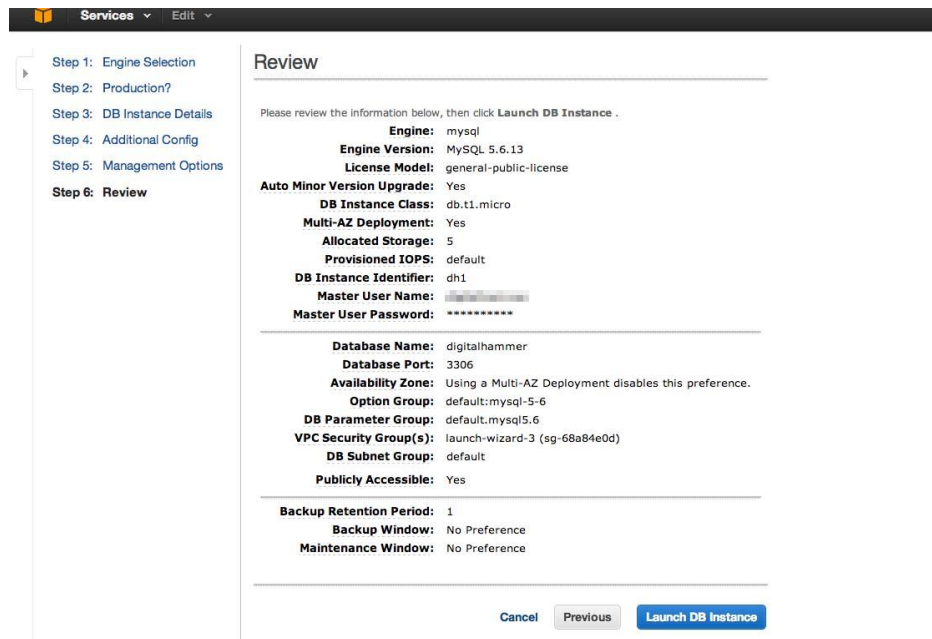
The weekly time range (in UTC) during which system maintenance can occur.

Maintenance Window: Select Window No Preference

Cancel Previous **Next Step**

KUVIO 14. Automaattisten varmuuskopioiden määrittely (Amazon Web Services 2014)

Viidennessä askeleessa kysyttiin, halutaanko instanssista luoda automaattisesti varmuuskopioita tietyn ajanjakson välein. Koska varmuuskopiointi on todella tärkeä asia, valittiin, että varmuuskopiointi tapahtuisi automaattisesti päivittäin kello 12 aamuyöllä ja varmuuskopioita säilytetään yhden päivän ajan. (KUVIO 14.)



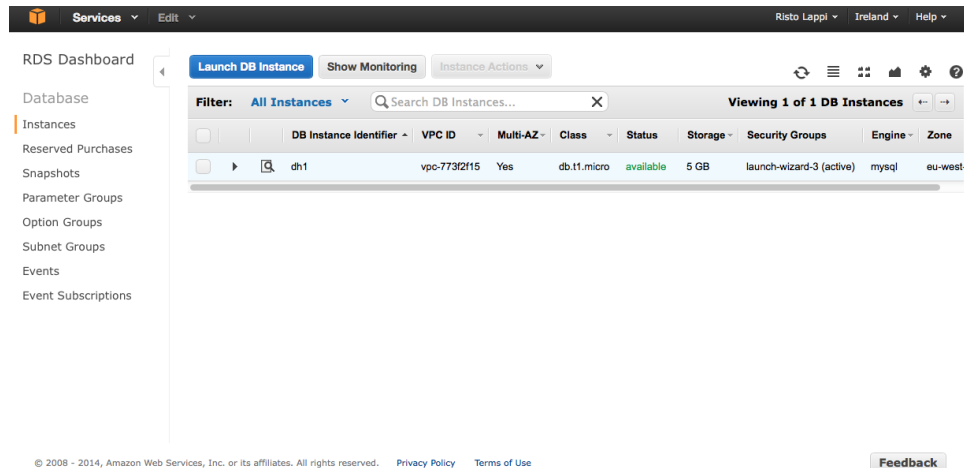
KUVIO 15. Instanssin tiedot ennen käynnistämistä (Amazon Web Services 2014)

Viimeisessä askeleessa AWS näyttää tulevan instanssin kaikki tiedot, jotka edellä oli määritelty. Tästä ikkunasta voidaan nähdä, ovatko kaikki tiedot kunnossa, ja niitä voi myös palata muokkamaan, jos jokin asia unohtui määritellä. Koska tiedot olivat kunnossa, voitiin painaa oikeassa alakulmassa näkyvää sinistä ”Launch DB Instance” –nappia ja instanssi oli nyt valmis ja käynnistyi. (KUVIO 15.)



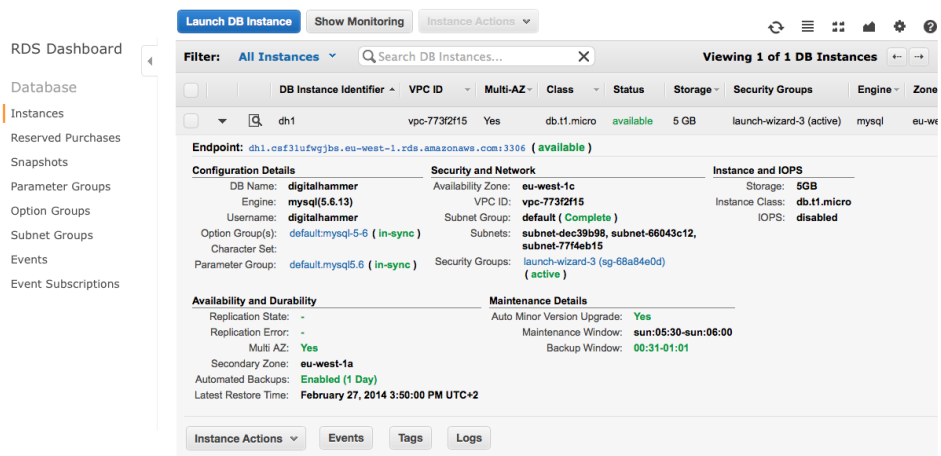
KUVIO 16. AWS ilmoittaa, että instanssi on luotu ja käynnissä (Amazon Web Services 2014)

Kun instanssi oli käynnistynyt, siitä saatiin vielä erikseen ilmoitus. Ilmoituksen alapuolella näkyvästä sinisestä linkistä voidaan tutkia käynnissä olevia instansseja, kuten tätä äsken luotua. (KUVIO 16)



KUVIO 17. RDS:n ”Instances” –valikko, josta nähdään käynnissä olevat instanssit (Amazon Web Services 2014)

Kun edellä mainitun linkin valitsi, päästiin ”RDS Dashboardiin”, josta ohjattiin suoraan ”Instances”-valikkoon. Valikossa näkyi nyt juuri luotu instanssi. Statuskohdassa luki ”available”, eli instanssi oli nyt käynnissä ja valmiina käytettäväksi palvelimessa. (KUVIO 17.)



KUVIO 18. Valmiin instanssin tietojen tarkastelu (Amazon Web Services 2014)

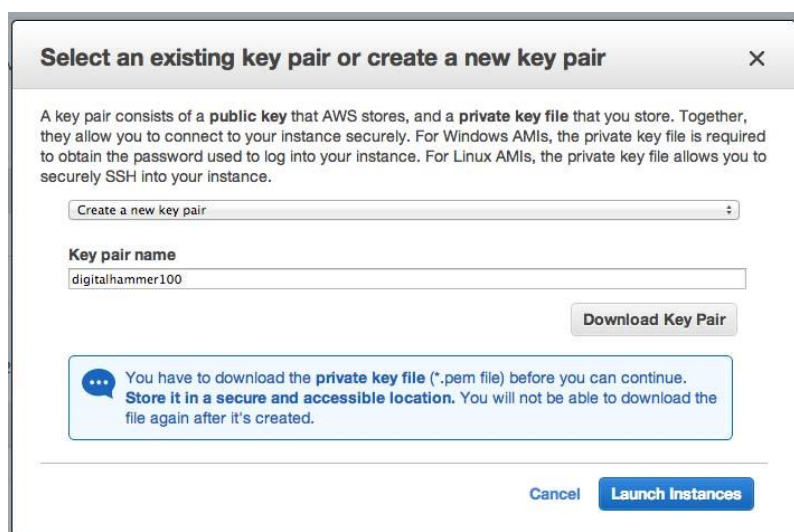
Instanssin ”identifinerin” eli nimekkeen vasemmalla puolella olevaa nuolta näpäyttämällä päästään vielä lähemmin tutkimaan instanssin tietoja, jotka tälle oli asetettu. Tiedoista selviää muun muassa instanssin sijainti, eli Eurooppa, instanssin pohjalla pyörivä ohjelmistomoottori sekä se, milloin viimeksi automaattinen varmuuskopio on tehty. (KUVIO 18.)

5.5 Avainparin luonti

Jotta palvelininstanssia voitaisiin käyttää tietoturvallisesti, tulee tätä varten luoda ”key pair” eli avainpari. Avainparin avulla instanssiin voidaan kirjautua erillisellä pääteohjelmistolla.

Avainpari on erillinen tiedosto, joka luodaan AWS:ssä ja ladataan sitten tietokoneen muistiin. Tiedostoa voidaan jakaa esimerkiksi yrityksen työntekijöiden välillä verkkokiintolevyllä, jotta jokainen työntekijä voi kirjautua palvelimelle lataamalla omalle tietokoneelleen avainparin. Tämä ei kuitenkaan ole tietoturvallista hakkeroinnin vuoksi.

Työvaiheiden alussa esitellystä AWS:ohjauspaneelistä ”Network & Security” -kohdasta voitiin nyt luoda avainpari valitsemalla kohta ”Key Pairs” ja ”Create Key Pair”. Tämän jälkeen avainparille annettiin nimi ja painettiin ”Download Key Pair” -näppäintä. Avainpari oli nyt luotu ja voitiin ladata tietokoneen kovalevylle talteen. (KUVIO 19.)



KUVIO 19. Avainparin luonti (Amazon Web Services 2014)

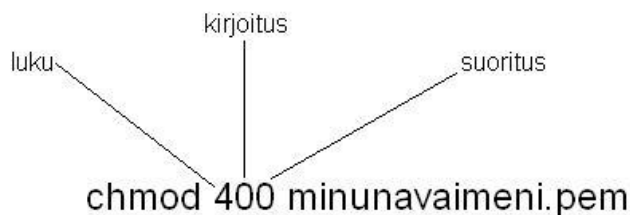
Luotu avainpari ladattiin .pem-tiedostona tietokoneen kovalevylle. Avainpari on varustettu 2048-bittisellä RSA-salauksella, eli avainpari pitää sisällään melko vaikeasti murrettavan matemaattisen algoritmin.

5.6 Palvelimelle kirjautuminen SSH:n avulla

Kun instanssi ja avainpari kirjautumista varten oli luotu, voitiin palvelimelle kirjautua ensimmäistä kertaa. Tässä vaiheessa tarvittiin käyttöön terminaaliemulaattori eli pääteohjelmisto, jonka avulla palvelimelle voitiin kirjautua käyttämällä ”Secure SHelliä” eli SSH:ta. SSH:llä voidaan luoda salattu yhteys juuri luotuun instanssiin.

Ennen pääteohjelman käynnistystä tuli selvittää AWS:stä instanssin IP- eli ”Internet Protocol” –osoite. Osoite löytyi jo aiemmin esitellyn käynnissä olevan instanssin lisätietoja tutkimalla (KUVIO 18).

Kun pääteohjelma oli käynnistetty, täytyi cd- eli ”Change Directory” –komentoa käyttäen hakeutua oikeaan hakemistoon tietokoneen kovalevyllä, jossa sijaitsi äskeittäin luotu avainpari, jotta pääteohjelma tietää, mistä kirjautumiseen vaadittava avainpari löytyy. Ennen kirjautumista täytyi kuitenkin muokata avainparin käyttöoikeuksia, ettei avainpari olisi täysin vapaasti muokattavissa tai nähtävillä. Tämä tapahtui sen jälkeen, kun pääteohjelmalla oli hakeuduttu oikeaa hakemistoon, jossa avainpari sijaitsee. Komennolla chmod eli ”Change Mode” voitiin muokata tiedostojen luku-, kirjoitus- sekä suoritusoikeuksia. Komennon perään kirjoitetaan kolminumeroinen luku, jossa jokainen luku määrää erikseen, mikä on tiedoston käyttöoikeus (KUVIO 20). Päätettiin, että avainparista tehdään julkisesti näkymätön, joten annettiin komento ”chmod 400 minunavaimeni.pem”. Tällöin kirjoitus- ja suoritusoikeudet on nolliä, jolloin tiedostoa ei voi muokata eikä suorittaa. Numero neljä, joka määrittelee lukuoikeutta, määrää tiedoston vain ja ainoastaan luettavaksi ja julkisesti näkymättömäksi.



KUVIO 20. Tiedoston käyttöoikeuksien muuttaminen

Kun halutaan kirjautua salaisesti käyttämällä SSH:ta, ilmoitetaan ensin, millä protokollalla kirjaututaan, avainparin tiedostonimi sekä IP-osoite ja tässä tapauksessa Linux-versio, joka oli Ubuntu, eli esimerkiksi ”ssh –i minunavaimeni.pem ubuntu@111.111.111.111”. Edellä mainitulla komennolla päästiin kirjautumaan palvelimelle, ja oli aika alkaa muokkaamaan palvelimen rakennetta enemmän pelipalvelimen malliseksi.

5.7 Palvelimen perusasetukset ja sovellusten asentaminen

Sisäänkirjaututtaessa kirjaututaan pääkäyttäjänä, ”rootina” eli niin sanotusti juurena. Ensimmäiseksi tuli määritellä rootille uusi salasana, sillä kun palvelimelle kirjaututaan, pääkäyttäjän salasana on sama kuin pääkäyttäjän nimi, eli ”root”. Salasana voitiin muuttaa passwd-komennolla eli esimerkiksi ”passwd salasana”. Yleisiä nykyajan tietoturvasääntöjä noudattaen salasanasta kannattaa tehdä mahdollisimman vaikeasti arvattava.

Seuraavaksi päätettiin, että palvelimelle luodaan pääkäyttäjän lisäksi kolme kappaletta peruskäyttäjiä, joilla on hallintaoikeudet. Käyttäjien luominen suoritettiin sudo- eli ”super do” –komennolla, jolloin käyttäjät saavat niin sanotut sudo-oikeudet. Komento käyttäjän luomiseksi Ubuntussa on hyvin yksinkertainen: sudo adduser käyttäjänimi sudo, eli jos halutaan luoda peruskäyttäjä hallintaoikeuksilla nimimerkillä ”Emil”, kirjoitetaan ”sudo adduser Emil sudo”. Tällöin luotu käyttäjä saa hallintaoikeudet ja käyttäjä lisätään ”sudo grouppiin” eli sudo-ryhmään. Käyttäjien luominen ja komennot vaihtelevat hieman Linux-versioittain, kuten esimerkiksi Ubuntu vanhemmissa versioissa hallintaoikeuden omaava käyttäjä luodaan komennolla ”sudo adduser käyttäjä admin”. Sudo-ryhmän nimi on siis aikaisemmin ollut ”admin”-ryhmä.

Kun perusasetukset oli saatu kuntoon ja käyttäjät luotu, oli aika asentaa palvelimelle sovelluksia. Päätettiin asentaa perussovellukset parhaimman toimivuuden saavuttamiseksi eri ohjelmointikielillä ohjelmoitujen sovellusten kesken, joita palvelin mahdollisesti joutuisi käyttämään. Asennettaviksi sovelluksiksi valittiin MySQL, PHP, Apache, phpMyAdmin sekä Python. Ubuntuun sovellusten asennus on yksinkertaista ja asennukseen yleisesti käytettävä komento on ”sudo apt-get”, jolla haetaan sovellus Linuxin sovelluslistalta ja perään kirjoitetaan ”install” sekä sovelluksen nimi, jolloin järjestelmä asentaa sovelluksesta uusimman version.

MySQL-tietokantapalvelin voitiin asentaa käyttämällä komentoa ”sudo apt-get install mysql-server”, jolloin palvelin saatiin käsittelemään muiden tietokantapalvelimien kesken ja etenkin ymmärtämään MySQL-tietokantoja. Asennuksen aikana tuli asettaa ”root”-käyttäjän salasana sovellukselle. Kun asennus oli valmis, sovellus käynnistettiin varmuuden vuoksi uudelleen komennolla ”sudo service mysql restart”. (Ubuntu Documentation 2015a.)

PHP on verkkoprotokolla, joka on ehtinyt jo viidenteen versioonsa, joten palvelimelle asennettiin PHP5. Tämä tapahtui komennolla ”sudo apt-get install php5”. Palvelimelle PHP on elintärkeä protokolla, sillä suurin osa verkossa olevista sivustoista on luotu PHP:llä. (How-To Geek 2006.)

Apachen asennus oli myös yksinkertaista. Komennolla ”sudo apt-get install apache2” voitiin asentaa Apachen uusin versio. Nyt palvelimen HTTP-protokollan käyttöä voitiin hoitaa Apachen avulla. Apache käynnistettiin asennuksen päätteeksi uudelleen komennolla ”sudo etc/init.d/apache2 restart”. (How-To Geek 2006.)

Koska tietokantoja haluttiin hallinnoida pääkäyttäjän tasolta, asennettiin palvelimelle seuraavaksi phpMyAdmin. Tämä tapahtui niin ikään komentoa ”sudo apt-get install phpmyadmin” käyttäen. Nyt erilaisten, kuten esimerkiksi MySQL- ja MariaDB-tietokantojen hallinta ja valvonta oli mahdollista. (Ubuntu Documentation 2015b.)

Oli aika asentaa viimeinen sovellus, joka oli Python-ohjelmointikielen tuki. Komennolla ”sudo apt-get install python-software-properties” saatiin tarvittavat

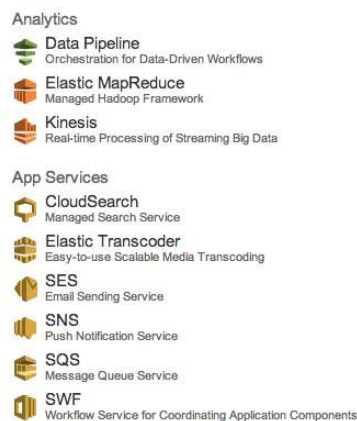
työkalu, jotta palvelin osaisi käsitellä Pythonilla ohjelmoitujen sovellusten läpikulkevaa tietoa. (LinuxG 2015.)

5.8 Klusterin luonti

Kun sovellukset saatiin asennettua, oli aika siirtyä siihen, mistä tässä koko työssä oli kyse, eli klusterin luontiin. Asetukset tehtiin melko lailla Amazonin suositusten mukaan, mutta halutessaan asetuksia voi toki muuttaa myös jälkikäteen. Alustaksi tietopankin hallintaan päätettiin käyttää Apachen Hiveä, sillä Hive toimii käsi kädessä SQL-tyyppisten kyselykielten kanssa. Hive oli alun perin sivuroolissa Apachen Hadoop –projektissa, mutta vähitellen Hivestä tuli täysin oma sovelluksensa. Hadoop on sovellus, jonka avulla voidaan liikuttaa suuria datamääriä klustereiden välillä. (Apache Hive TM 2014.)

Jotta Hiveä voitaisiin käyttää AWS:n kanssa, tuli ensimmäisenä luoda jälleen uusi avainpari. Avainpari luotiin täsmälleen samalla tavalla, kuten kohdassa 3.3, tosin eri nimellä, kuin palvelimen avain.

Avainparin luonnin jälkeen palattiin takaisin AWS:n ohjauspaneeliin. ”Analytics”-valikon alta löytyy kohta nimeltään EMR eli ”Elastic MapReduce”, josta klusteri voidaan luoda (KUVIO 21).



KUVIO 21. Analytics-valikko (Amazon Web Services 2014)

EMR-kohdan alta painettiin ”Create cluster” –nappia, josta valittiin kohta ”Cluster configuration”. ”Cluster configurationin” alta avautui jälleen uusi ikkuna, jossa klusterin perustiedot määriteltiin.

The screenshot shows the 'Cluster Configuration' window in the AWS Management Console. The settings are as follows:

- Cluster name:** My cluster
- Termination protection:** Yes (selected with a blue radio button)
- Logging:** Enabled (selected with a blue checkbox)
- Log folder S3 location:** s3:// (with a dropdown menu showing the template s3://<bucket-name>/<folder>/)
- Debugging:** Enabled (selected with a blue checkbox)

KUVIO 22. Klusterin konfigurointi (AWS Documentation 2015a)

Klusterin konfigurointi-ikkunasta (KUVIO 22) määriteltiin ensin klusterille nimi. Kohdassa ”Termination protection” voitiin valita, haluttiinko klusterin pysyvän käynnissä, mikäli sattuisi virhe. Tähän kohtaan valittiin ”Yes”-vaihtoehto. ”Logging”-kohdassa valittiin asetukseksi ”Enabled”, sillä haluttiin, että EMR tallentaisi sisäänkirjautumistiedot joka kerta, kun käyttäjä kirjautuu. Alempana olevaan kohtaan määriteltiin polku, johon niin sanottu ”debug data” eli instanssien testaustiedot tallentuvat. Käytettäväksi tallennuspaikaksi valittiin Amazonin S3 eli ”Amazon Simple Storage Service”, joka on Amazonin tarjoama sovelluskehittäjäläheinen verkkotallennuspalvelu. Viimeiseen kohtaan valittiin, haluttiinko käyttää ”debuggausta”, eli halutaanko yksityiskohtaiset testaustiedot tallentaa hakemistona Amazonin SimpleDB:n sisälle. Amazon SimpleDB on tietokantojen tallennuspalvelu, joka ei toimi relaatiotietokantana, eli kehittäjän ei tarvitse kuin tehdä tietokantakysely ja Amazon SimpleDB hoitaa loput (Amazon Web Services 2015a).

Software Configuration

Hadoop distribution Amazon Use Amazon's Hadoop distribution. [Learn more](#)

AMI version Determines the base configuration of the instances in your cluster, including the Hadoop version. [Learn more](#)

MapR Use MapR's Hadoop distribution. [Learn more](#)

Applications to be installed	Version			
Hive	0.11.0.2			
Pig	0.12.0			

Additional applications

KUVIO 23. Ohjelmiston konfigurointi (AWS Documentation 2015a)

Seuraavaksi voitiin siirtyä ”Software configuration”-kohdan alle (KUVIO 23). ”Hadoop distribution” -kohta yksinkertaisesti määrittelee, kenen Hadoop-pohjaista sovellusta käytetään. Koska käytettiin Amazon Web Servicesin palveluita, vaihtoehtoja ei ollut muita kuin valita ”Amazon”. Kohdasta ”AMI version” voitiin valita, mikä Hadoopin versio haluttiin asentaa ja minkä mukaan klusterin asetukset määräytyvät. Seuraavaksi vielä kysyttiin, mitä sovelluksia halutaan lisäksi asentaa. Päätettiin asentaa sekä Hive että Pig. Pigin avulla voidaan suorittaa analyysyjä ja siirtoja suurille datamäärille (AWS Documentation 2015e).

Hardware Configuration

i Specify the networking and hardware configuration for your cluster. If you need more than 20 EC2 instances, complete this form. [Request Spot instances](#) (unused EC2 capacity) to save money.

Network Use a Virtual Private Cloud (VPC) to process sensitive data or connect to a private network. [Create a VPC](#)

EC2 Subnet [Create a Subnet](#)

	EC2 instance type	Count	Request spot	
Master	<input type="text" value="m1.medium"/>	<input type="text" value="1"/>	<input type="checkbox"/>	The Master instance assigns Hadoop tasks to core and task nodes, and monitors their status.
Core	<input type="text" value="m1.medium"/>	<input type="text" value="2"/>	<input type="checkbox"/>	Core instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS).
Task	<input type="text" value="m1.medium"/>	<input type="text" value="0"/>	<input type="checkbox"/>	Task instances run Hadoop tasks.

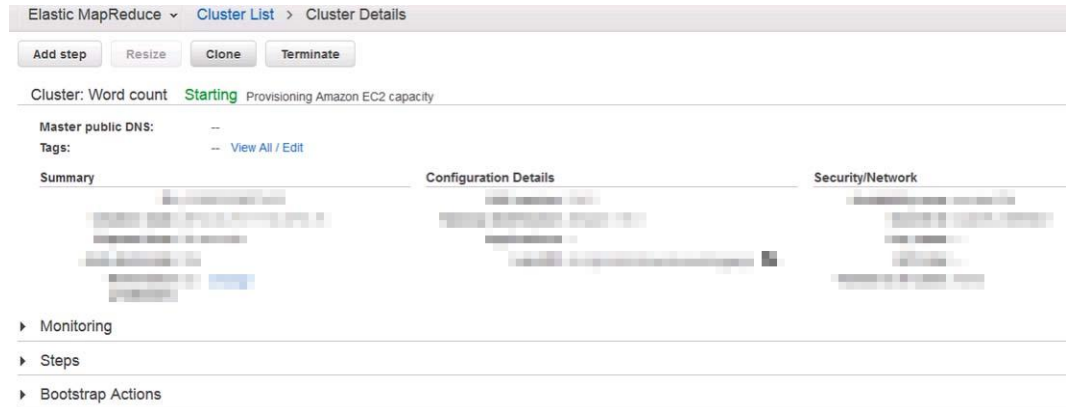
KUVIO 24. Laitteiston konfigurointi (AWS Documentation 2015a)

Kohdasta ”Hardware configuration” päästiin muokkaamaan laitteiston asetuksia (KUVIO 24). ”Network” määrittelee, halutaanko klusteria käyttää jossain tietyssä verkossa. Koska meillä oli valmiiksi aiemmin luotu mikroinstanssi, voitiin valita käytettäväksi instanssin verkko, jolloin klusteri käynnistyy siinä virtuaalikoneessa, johon instanssi oli luotu. Alempana oleva kohta ”EC2 subnet” määrittelee käytettävän aliverkon, joka jätettiin randomiksi eli satunnaiseksi. Kohdat ”Master”, ”Core” sekä ”Task” jätettiin mediumiksi, eli näillä asetettiin hallinta-, ydin- ja työnoodien toimintatapa. Äsken mainittujen noodien lukumäärä määriteltiin kohtaan ”Count”, jossa hallintanoodeja on vakiona yksi kappaletta ja koska toimitettiin Amazonin suositusten mukaisesti, päätettiin luoda kaksi kappaletta ydinnoodeja ja työnnoodeja ei toistaiseksi luotu. Jokaisen noodityypin ”Request spot” –laatikko jätettiin vielä lopuksi valitsematta, sillä AWS:ää käytettiin tässä vaiheessa ilmaiseksi, eikä ollut tarvetta varata maksullisia ”Spot”-instansseja. ”Spot”-instanssien avulla voidaan alentaa klusterien hintoja, mikäli ne ovat vajaakäytöllä (AWS Documentation 2015d).

Seuraavassa kohdassa määriteltiin vielä klusterin ”Security and Access” -asetukset eli turvallisuutta ja klusteriin käsiksi pääsemiseen liittyviä oikeuksia. Tässä vaiheessa valittiin tämän osion alussa luotu avainpari ”Key pair” -pudotusvalikosta. ”IAM user access” (Identity and Access Management) määrittää, pääsevätkö kaikki AWS-tilillä olevat IAM-käyttäjät, eli käyttäjät, jotka kykenevät hallitsemaan AWS:n käyttäjiä ja käyttöehtoja, klusteriin käsiksi ja näkykö se heille. Tähän valittiin ”No other IAM users”, eli klusteri ei näkyisi muille IAM-käyttäjille (AWS Documentation 2015c). Tämän jälkeen määriteltiin vielä, että klusterissa ei käytetä IAM-rooleja, eli liittyen edellä asetettuun ”No other IAM users” –valintaan pohjautuen, valittiin ”Proceed without roles”, jolloin muut IAM-käyttäjät eivät pääse käsiksi klusterin resursseihin (AWS Documentation 2015b).

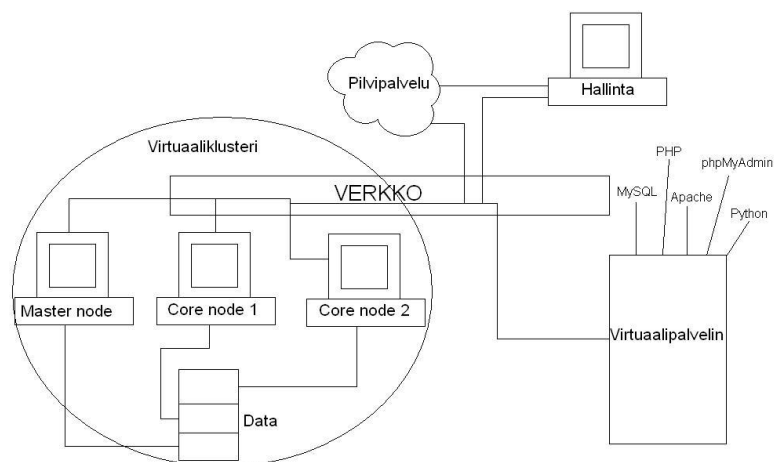
Kun turvallisuusasetukset oli tehty, saatiin vielä nähdä lopullinen näkymä tulevasta klusterista, jossa näkyi äsken tehdyt asetukset. Kaikki oli tarkastastelun jälkeen kunnossa, joten voitiin painaa ”Create cluster” –painiketta. Klusteri käynnistyi ja ilmestyi EMR-valikon ”Cluster listiin”, josta tietoja voitiin tarkastella ja mahdollisesti myös sammuttaa klusteri esimerkiksi päivitystöiden vuoksi. Tulevaisuudessa klusteria voitaisiin myös suoraan kloonata helposti

”Clone”-nappia painamalla siltä varata, että klusterin toiminnan raja tulee vastaan ja tarvittaisiin enemmän prosessointitehoa. Tällöin ”Clone”-painike loisi rinnalle uuden, täysin samanlaisen klusterin. (KUVIO 25.)



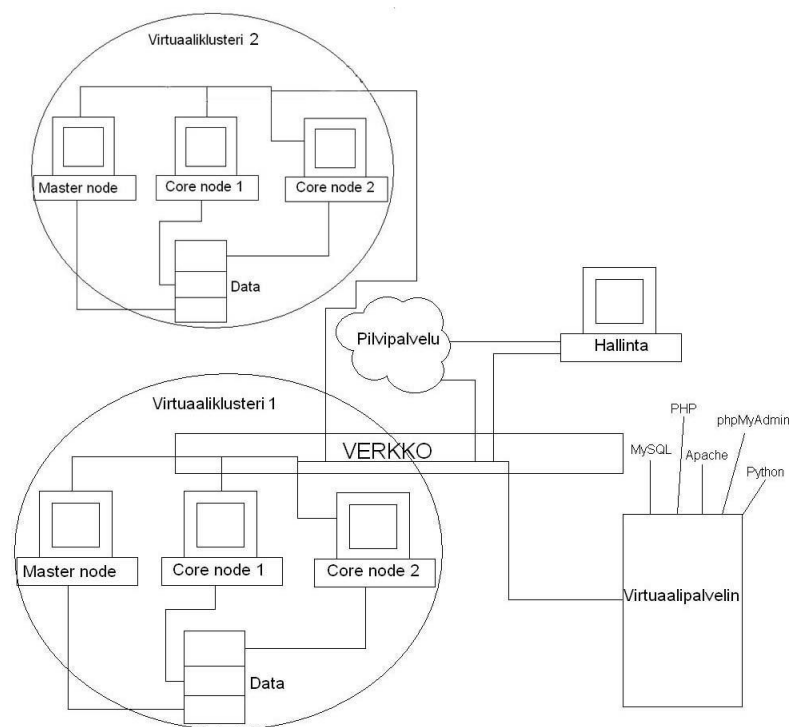
KUVIO 25. Käynnissä olevan klusterin yksityiskohdat (AWS Documentation 2015a)

Nyt klusteri oli valmis yrityksen työntekijöiden käyttöön pelipalvelimen klusterointia varten. Samalla kun klusteri käynnistyi, niin AWS asensi työvaiheiden alussa luotuun instanssiin Apache Hiven, jota voitaisiin käyttää suurten datamäärien hallintaan klusterissa.



KUVIO 26. Valmiin työn toimintakuva

Toimintakuvasta (KUVIO 26) voidaan havaita, kuinka virtuaalinen palvelin ja klusteri toimivat ja miltä kytkentä näyttää. Kaikki osapuolet on kytketty samaan verkkoon. Kuten aikaisemmin mainittiin, sekä palvelin että klusteri sijaitsevat pilvipalvelun sisällä. Hallintatietokone, joka tässä tapauksessa on mikä tahansa tietokone, jolla on kirjaututtu pilvipalveluun, lähettää käskyjä palvelimelle ja klusterille. Palvelimelle asennetut sovellukset (MySQL, PHP, Apache, phpMyAdmin ja Python) hoitavat kukin niille tarkoitettuja tehtäviään aina kun jotakin niiden ohjaamaa protokollaa tarvitsee käyttää palvelimella. Klusteri on perinteinen yhden päätäntävällän eli ”Single Quorum Device” -klusteri, jossa on yksi ”Master”- eli hallintanoodi ja kaksi ”Core”- eli ydinnoodia. Jokainen noodi on kytketty kovalevyjoukkioon. Mikäli vaaditaan runsaasti laskentatehoa, voidaan klusteria käyttää, jolloin useampi klusterin noodi tekee samanaikaisesti yhteistyötä raskaan prosessin suorittamiseksi. Jos tarvitaan vielä enemmän laskentatehoa, pilvipalvelusta voidaan käydä kloonaamassa uusi klusteri vanhan klusterin rinnalle (KUVIO 27).



KUVIO 27. Klusterin kloonaus

Työn valmistuttua klusteri ja palvelin jätettiin pilvipalvelun sisään käyntiin. Myöhemmin, kun peliin oli luotu moninpelimahdollisuus, voitiin palvelin liittää pelin monipelitietojen tallennuspaikaksi. Palvelimen liittämisen itse pelin tallennuspaikaksi tekivät yrityksen ohjelmoijat.

6 YHTEENVETO

Työ onnistui kaiken kaikkiaan hyvin ja tavoitteet tulivat täytetyiksi. Tarkoituksena oli luoda helposti kloonattava ja kasvatettava klusteri pelipalvelimen käyttöön. Nyt yrityksellä olisi helposti komentotulkin avulla hallittava tallennusvarasto monipelidataa varten, johon voitaisiin varastoida pelitilien tallennuksia sekä tilastoja siitä, kuinka hyvin pelaaja on pelannut eli niin kutsuttuja ”ranking listejä”. Työstä oli yritykselle suurta hyötyä, sillä ohjelmoijien aikaa säästy ja he saivat keskittyä enemmän itse pelinkehitykseen.

Myöhemmin, kun peli alkoi muokkautua lopulliseen muotoonsa, kävi ilmi, että Hive ei ollut tarpeeksi yhteensopiva alusta yrityksen pelin tarpeisiin. Peli on tehty käyttämällä Unity3D-peliohjelmointityötä, jolle sopivampi vaihtoehto oli Photon-niminen sovellus, joka hoitaa monipelitilastointia. Photon oli myös paljon toimivampi ratkaisu, sillä yritys lähti myöhemmin markkinoimaan peliään myös ulkomaille ja Photon mahdollisti pätkimättömän pelaamiskokemuksen myös ulkolaisille. Tässä työssä luotu palvelin on kuitenkin edelleen käytössä ja sitä on muokattu yrityksen tarpeille sopivaksi jälkepäin työntekijöiden toimesta.

Tulevaisuudessa yritys voi ehostaa klusteria esimerkiksi luomalla skriptin eli ohjelman, joka tekee sille asetettuja toimenpiteitä aina, kun jotakin tiettyä tietokoneen näppäintä painetaan. Skriptiä voitaisiin käyttää esimerkiksi automaattisten varmuuskopioiden luontiin. Mikään ei ole tärkeämpää kuin vankka tietoturvallisuus ja varmuuskopioiden ottaminen usein.

Sekä instanssin että palvelimen avainpareja varten voitaisiin kehittää jokin tietoturallinen tallennusratkaisu, jotta avainpareja voitaisiin säilöä verkossa ja tarvittaessa yrityksen työntekijä voi esimerkiksi kotona työskennellessään ladata avainparit tietokoneellen ja päästä näin ollen käsiksi klusteriin ja palvelimeen. Yksi vaihtoehto olisi tietysti jokaiselle työntekijälle hankittava muistitikku, joka kryptataan esimerkiksi Keypass-ohjelmiston avulla.

Nykyaikana, kun virtualisointi alkaa saada näkyvyyttä yhä enemmän tietoliikennetekniikassa, voidaan niinkin suuria asioita kuin usean tietokoneen joukkiota hoitaa täysin virtuaalisesti ja hallita verkon kautta vain yhdellä tietokoneella. Tämä säästää energiaa sekä klustereita ja virtuaalisia ”leasing”- eli

vuokratietokoneita tarvitsevien yritysten pääomaa. Virtualisointi on siis todellista ”Green ICT:tä” eli vihreitä arvoja omaksuvaa tietotekniikkaa. Virtualisointi vähentää myös näkyvästi elektroniikkaromun määrää kaatopaikoillamme. Virtualisoinnissa on epäilemättä vihreän tietotekniikan tulevaisuus.

LÄHTEET

Amazon Web Services. 2014. AWS [viitattu 27.2.2014]. Saatavissa:

<http://aws.amazon.com/>

Amazon Web Services. 2015a. Amazon SimpleDB [viitattu 23.4.2015]

Saatavissa: <http://aws.amazon.com/simplydb/>

Amazon Web Services. 2015b. What is AWS? [viitattu 12.5.2015] Saatavissa:

<http://aws.amazon.com/what-is-aws/>

Apache. 2015. Apache [viitattu 27.5.2015]. Saatavissa: <http://apache.org/>

Apache Hive. 2014. Apache Hive TM [viitattu 23.4.2015]. Saatavissa:

<https://hive.apache.org/>

AWS Documentation. 2015a. Amazon Elastic MapReduce [viitattu 23.4.2015].

Saatavissa:

http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/EMR_CreateJobFlow.html

AWS Documentation. 2015b. Configure IAM Roles for Amazon EMR [viitattu 23.4.2015] Saatavissa:

<http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-iam-roles.html>

AWS Documentation. 2015c. Configure IAM User Permissions [viitattu 23.4.2015]. Saatavissa:

<http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-plan-access-iam.html>

AWS Documentation. 2015d. Lower Costs with Spot Instaces [viitattu 23.4.2015].

Saatavissa:

<http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-plan-spot-instances.html>

AWS Documentation. 2015e. Pig and Amazon EMR [viitattu 23.4.2015].

Saatavissa:

<http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-pig.html>

Google Cloud. 2015. Google Cloud [viitattu 12.5.2015]. Saatavissa: <https://cloud.google.com/>

How-To Geek. 2006. Installing PHP5 and Apache on Ubuntu [viitattu 22.4.2015]. Saatavissa: <http://www.howtogeek.com/howto/ubuntu/installing-php5-and-apache-on-ubuntu/>

IGN. 2015. Kaakkois-Aasian pelimarkkinat tuplaantuvat vuoteen 2017 mennessä [viitattu 22.4.2015]. Saatavissa: <http://fi.ign.com/kaikki/68236/news/newzoo-kaakkois-aasian-pelimarkkinat-tuplaantuvat>

Internetopas. 2014. Internetin rakenne ja toimintaperiaate [viitattu 16.10.2014]. Saatavissa: <http://www.internetopas.com/yleistietoa/rakenne/>

Interoute. 2015. What is Paas? [viitattu 7.5.2015]. Saatavissa: <http://www.interoute.fi/what-paas>

Jyväskylän yliopisto. 2015. PHP:n perusteet [viitattu 15.1.2015]. Saatavissa: http://users.jyu.fi/~kolli/ITK215_05/php/

Koudata. 2010. OSI-malli [viitattu 28.4.2015]. Saatavissa: <http://www.koudata.fi/node/598>

LinuxG. 2015. How to install Python [viitattu 22.4.2014]. Saatavissa: <http://linuxg.net/how-to-install-python-3-3-on-ubuntu-13-04-12-10-and-12-04/>

Microsoft Technet. 2015. What is a server cluster? [viitattu 17.4.2015]. Saatavissa: <https://technet.microsoft.com/en-us/library/cc785197%28v=ws.10%29.aspx>

phpMyAdmin. 2015. About phpMyAdmin [viitattu 7.4.2015]. Saatavissa: http://www.phpmyadmin.net/home_page/index.php

Python. 2015. What is Python? [viitattu 7.4.2015]. Saatavissa: <https://www.python.org/doc/essays/blurb/>

Salesforce. 2015. SaaS [viitattu 7.5.2015]. Saatavissa:

<https://www.salesforce.com/saas/>

Saraibeam. 2015. Palvelin [viitattu 7.5.2015]. Saatavissa:

<http://www.saraibeam.fi/ratkaisu/palvelin>

Server Computer. 2014. Server Computer [viitattu 15.10.2014]. Saatavissa:

<http://www.servercomputer.org>

SS64. 2015. An A-Z Index of the Bash command line for Linux [viitattu

13.5.2015]. Saatavissa: <http://ss64.com/bash/>

Talouselämä. 2015. Jättimenestys: Supercell teki 1,5 miljardin liikevaihdon [viitattu 7.4.2015]. Saatavissa:

<http://www.talouselama.fi/uutiset/jattimenestys+supercell+teki+15+miljardin+liikevaihdon/a2298697>

Taloussanomat. 2014. Rovio Entertainment Oy [viitattu 14.10.2014]. Saatavissa:

<http://yritys.taloussanomat.fi/y/rovio-entertainment-oy/espoo/1863026-2/>

Techtarget. 2006. What is Ethernet? [viitattu 7.5.2015]. Saatavissa:

<http://searchnetworking.techtarget.com/definition/Ethernet>

Techtarget. 2015. What is IaaS? [viitattu 7.5.2015]. Saatavissa:

<http://searchcloudcomputing.techtarget.com/definition/Infrastructure-as-a-Service-IaaS>

Ubuntu Documentation. 2015a. MySQL [viitattu 22.4.2015]. Saatavissa:

<https://help.ubuntu.com/12.04/serverguide/mysql.html>

Ubuntu Documentation. 2015b. phpMyAdmin [viitattu 22.4.2015]. Saatavissa:

<https://help.ubuntu.com/12.04/serverguide/phpmyadmin.html>

W3Schools. 2015a. PHP MySQL Database [viitattu 8.1.2015]. Saatavissa:

http://www.w3schools.com/php/php_mysql_intro.asp

W3Schools. 2015b. What is TCP/IP? [viitattu 28.4.2015]. Saatavissa:

http://www.w3schools.com/website/web_tcpip.asp

Windows Azure. 2015. Windows Azure [viitattu 12.5.2015]. Saatavissa:
<http://www.windowsazure.com/en-us/home/features/overview>

