Haaga-Helia
ammattikorkeakoulu Oy

IT-infrastruktuurin valvonta AWS-pilvipalvelussa

Tomas Vermilä

Opinnäytetyö
Tietojenkäsittelyn koulutusohjelma
2019

Tekijä(t)
Tomas Vermilä

Koulutusohjelma
Tietojenkäsittelyn koulutusohjelma

Opinnäytetyön otsikko
IT-infrastruktuurin valvonta AWS-pilvipalvelussa

Sivu- ja liitesivumäärä
25 + 3

Opinnäytetyön otsikko englanniksi
Monitoring AWS cloud infrastructure

IT-infrastruktuurin valvonta on tärkeässä osassa vikatilanteiden ratkaisemisessa. Sen ansiosta vikatilanteisiin voidaan reagoida nopeasti. Valvonnasta saatavien raporttien perusteella voidaan myös löytää infrastruktuurissa olevia ongelmakohtia.

Tässä työssä tutkittiin, kuinka valvontaympäristö luodaan AWS-pilvipalveluun ja mitä työkaluja AWS tarjoaa valvonnan toteutukseen. Työn empiirisessä osassa rakennettiin pienimuotoinen IT-infrastruktuuri AWS-pilvipalveluun. Se suunniteltiin vikasietoiseksi AWS:n parhaiden käytäntöjen mukaisesti, mikä parantaa palveluiden saatavuutta ja vikasietoisuutta. Sen jälkeen testattiin eri valvontapalveluiden toimintaa luomalla vikatilanteita tahallisesti ja tutkittiin, kuinka vikatilanteisiin pystytään reagoimaan.

Valvontaympäristö ja testit saatiin suoritettua onnistuneesti. Todettiin kuitenkin, että toimivan valvonnan toteuttaminen vaatii usean eri AWS-palvelun yhteiskäyttöä, mikä hankaloittaa valvonnan toteuttamista verrattuna kolmannen osapuolen valvontasovelluksiin, kuten N-Centraliin, mikä tarjoaa kaiken yhdessä ja samassa paketissa. Varsinaista vertailua kolmannen osapuolen valvontasovelluksiin ei tässä työssä kuitenkaan tehty, vaikka lopuksi tutkimuksessa rakennettua valvontaratkaisua lyhyesti vertaillaan N-Central-valvontasovellukseen. Erityisen positiivisena asiana koettiin CloudWatchin tuottama laaja valvontastatistiikka, mikä helpottaa infrastruktuurissa esiintyvien ongelmakohtien löytämistä.

Asiasanat
Pilvipalvelut, pilvipalveluiden valvonta, IT-infrastruktuuri

Sisällys

Käsitteet ja termit
1 Johdanto 2
2 Pilvipalvelut 3
2.1 Pilvipalvelun hyödyt 3
2.2 Pilvipalvelumallit 4
2.2.1 SaaS – Sovellukset palveluna 4
2.2.2 PaaS – Sovellusalusta palveluna 4
2.2.3 IaaS – Infrastruktuuri palveluna 4
2.3 Pilvimallit 5
3 Amazon Web Services -pilvipalvelu 6
3.1 Historiaa 6
3.2 AWS-palveluiden valvonta 7
3.2.1 Amazon CloudWatch 8
3.2.2 AWS CloudTrail 8
3.2.3 AWS Config 9
3.2.4 Amazon Simple Notification Service (SNS) 9
4 Valvontaympäristön rakentaminen 10
4.1 VPC – Virtual Private Cloud 10
4.2 VPC-ympäristön perustaminen 10
4.3 Aliverkot 10
4.4 Internet-yhdyskäytävä (IGW) 11
4.5 Reititystaulut 11
4.6 Palomuurit (Security Group) 11
4.7 EC2-instanssit 12
4.8 RDS-palvelun käyttöönotto 13
4.9 Kuormantasaaja (ELB) 14
4.10 Skaalautumisryhmä EC2-instansseille 14
5 Valvonnan toteuttaminen 17
5.1 EC2-instanssien valvonta 17
5.2 RDS-ympäristön testaus ja valvonta 18
5.3 Kuormantasaajan valvonta 19
5.4 Automaattisen skaalautumisryhmän valvonta 19
5.5 Ympäristön valvonta Configilla 20
6 Pohdinnat 22
Lähteet 24

Käsitteet ja termit

FAILOVER Ohjelman tai palvelun suoritus siirtyy varalla olevaan järjestelmään vikaantuneesta järjestelmästä.

NACL Network Access Control List – verkon pääsynvalvontalista. Rajoitetaan pääsyä eri verkon osiin.

S3 Simple Storage Service. Amazonin pilvitallennustilaa tarjoava palvelu.

S3-ÄMPÄRI Kansio S3-palvelussa, mikä sisältää tallennetun data.

SNS Simple Notification Service. Amazonin palvelu, jonka avulla voidaan lähettää viestejä tekstiviestillä tai sähköpostilla.

SSD Solid State Drive – puolijohdelevy. Käytetään tietokoneen massamuistina ja on huomattavasti nopeampi kuin perinteinen magneettinen kiintolevy.

SYSTEMS MANAGER

AWS-palvelu, jonka kautta on mahdollista mm. automatisoida toimintoja ja suorittaa niitä useisiin instansseihin kerralla.

VIKASIETOINEN JÄRJESTELMÄ

Järjestelmä, joka jatkaa toimintaansa suorituskyvyn kärsimättä, vaikka jokin sen komponenteista vikaantuisi.

1 Johdanto

Nykypäivänä tuotanto voi pysähtyä yrityksessä kokonaan, jos tietojärjestelmä ei toimi. Tämän takia IT-infrastruktuurin vikasietoisuus on erittäin tärkeässä roolissa. Pilvipalvelun avulla pystytään ympäristöä skaalaaman tarpeen mukaan ja järjestelmien saatavuus pitämään korkeana.

Olen viimeiset kolme vuotta tehnyt IT-ympäristön valvomotyötä, jossa on keskitytty häiriövalvontaan, virtuaalisen ja fyysisen IT-ympäristön ylläpitoon. Lisäksi tuotetaan asiakkaillemme erilaisia raportteja heidän ympäristöstään ja ratkaistaan siihen liittyviä ongelmia. Asiakkaillemme tarjoama konesalikapasiteetti ei kuitenkaan ole verrattavissa nykyisiin suuriin pilvipalveluihin siinä, että se ei ole yhtä helposti ja nopeasti skaalattavissa, eikä asiakas itse voi tehdä kapasiteettiinsa muutoksia. Halutessaan lisää resursseja asiakas tekee siitä palvelupyynnön, jonka toteutamme tietyn vasteajan puitteissa. Minua kuitenkin kiinnosti nähdä kuinka vastaavantyyppisen valvontaympäristön toteuttaminen onnistuisi pilvipalveluun. Työkokemuksestani on varmasti apua tämän tutkimuksen suorittamisessa. Uskon tämän tutkimuksen auttavan myös muita, jotka harkitsevat IT-infrastruktuurinsa siirtämistä pilveen ja pähkäilevät ympäristönsä valvonnan kanssa.

Kolmannen osapuolen tarjoamia ohjelmistoja pilvipalveluiden valvontaan on saatavilla, mutta tässä työssä keskityn IT-ympäristön valvontaan, ongelmien ehkäisyyn ja havaitsemiseen käyttäen vain AWS:n tarjoamia palveluita.

Tässä tutkimuksessa AWS-pilvipalveluun rakennetaan pienimuotoinen IT-ympäristö. Vikatilanteisiin varaudutaan jo rakennusvaiheessa kahdentamalla ympäristö eri saatavuusalueille. Kun ympäristö on saatu luotua, sen komponentteihin kohdistetaan erilaisia rasitustestejä ja luodaan keinotekoisia vikatilanteita, minkä jälkeen tutkitaan, kuinka häiriötilanteisiin pystytään reagoimaan ja miten mahdolliset palvelun saatavuuskatkot pystytään minimoimaan. Rakennettava ympäristö tehdään siis vain tätä tutkimusta varten, eikä se ole tuotantokäytössä, mutta periaatteessa se voisi sellaiseen soveltua, vaikkakin on kooltaan hyvin pieni. Tähän tutkimukseen se on kuitenkin täysin riittävä ja sillä onnistuu erilaisten häiriötilanteiden testaus.

2 Pilvipalvelut

Pilvipalveluiden suosio on kasvanut viime vuosikymmenen lopusta lähtien räjähdysmäisesti. Pilvipalvelussa palveluntarjoaja tarjoaa konesaliresursseja käytettäväksi Internet-yhteyden yli. Tällöin yrityksessä ei tarvita lainkaan omaa fyysistä konesalia, vaan kaikki laskenta ja tiedon varastointi tapahtuu palveluntarjoajan konesalissa. Pilvipalvelun tarjoaja omistaa kaiken palvelinraudan ja vastaa sen ylläpidosta. Yritys vain vuokraa haluamiaan resursseja.

2.1 Pilvipalvelun hyödyt

Pilvipalvelussa joutuu maksamaan vain sen mukaan, mitä resursseja käyttää ja kuinka paljon. Resurssit on määritetty usein skaalautumaan tarpeen mukaan. Hiljaisina aikoina on vähemmän resursseja käytössä, minkä vuoksi laskut palvelusta ovat pienempiä. Sen ansiosta ei myöskään tarvitse tehdä suuria etukäteisinvestointeja konesaliin ja palvelimiin ilman, että on tarkkaan edes tiedossa, kuinka paljon resursseja todellisuudessa tarvitsee. Tämän ansiosta ei tule turhaan ylimitoitettua konesalikapasiteettia, minkä takia osa resursseista olisi tyhjän panttina ja niistä olisi maksettu turhaan.

Jos tarvetta konesalikapasiteetin kasvattamiselle ilmenee, niin pilvipalvelussa sen kasvattaminen käy helposti ja nopeasti, jopa muutamassa minuutissa. Normaalissa konesaliympäristössä kapasiteetin kasvattamiseen, esim. palvelimen hankintaan ja asentamiseen, menee usein viikkoja. AWS-pilvipalvelussa on myös täysin Amazonin hallitsemia palveluita, kuten S3 ja DynamoDB-tietokantapalvelu, jotka skaalautuvat automaattisesti, ilman käyttäjän toimenpiteitä. Amazon vastaa myös näiden järjestelmien päivittämisestä ja ylläpidosta.

Pilvipalveluiden ansiosta yritys ei välttämättä tarvitse lainkaan omaa fyysistä konesalia. Täten sen ei tarvitse myöskään keskittyä niin paljon konesaliympäristön kehittämiseen ja ylläpitoon, vaan voi keskittyä olennaiseen, eli omien ohjelmistojen ja palveluiden kehittämiseen.

Palveluiden tarjoaminen maapallon laajuisesti matalilla viiveillä käy myös pilvipalvelun ansiosta helposti ja nopeasti. Omiin konesaliresursseihin pääsee helposti käsiksi mistä vaan Internet-yhteyden avulla.

2.2 Pilvipalvelumallit

Pilvipalveluiden suosion kasvun myötä erilaisia pilvipalvelumalleja on tullut täyttämään erilaisia käyttäjien tarpeita. Jokainen näistä palvelumalleista antaa käyttäjälle eri tasoista hallittavuutta ja joustavuutta palveluun. Näiden pilvipalvelumallien ymmärtäminen auttaa yritystä löytämään juuri sille sopivan ratkaisun.

2.2.1 SaaS – Sovellukset palveluna

SaaS-palvelulla tarkoitetaan pilvessä sijaitsevaa ohjelmistoa, jonka ylläpidosta vastaa palveluntarjoaja. SaaS-palvelut välitetään verkkoselaimen kautta, sovelluksena tai näiden sekoituksena. Verkkoselaimen kautta välittäminen on suosituin tapa näistä kolmesta. (Pilvi.com, 2019) Käytännössä SaaS on vuokrattava ohjelmisto, josta maksetaan käytön mukaan. Palvelun ansiosta ohjelmiston käyttäjän ei tarvitse huolehtia ohjelmiston ylläpidosta, eikä fyysisestä palvelinraudasta ja pystyy keskittymään täysin ohjelmiston käyttämiseen. Hyvä esimerkki SaaS-pohjaisesta palvelusta on verkossa toimiva sähköpostipalvelu.

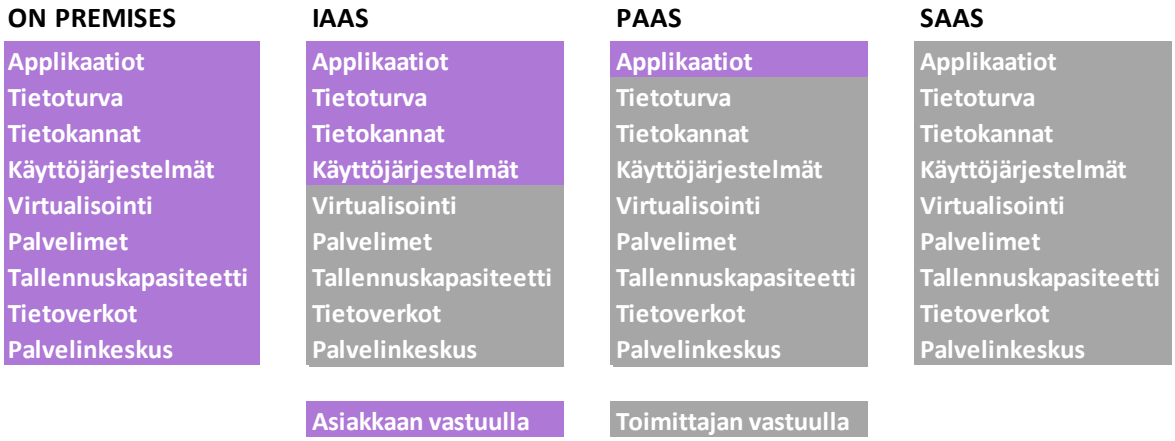
2.2.2 PaaS – Sovellusalusta palveluna

PaaS tarkoittaa nimensä mukaisesti alustapalvelua ohjelmistojen kehittämiseen, testaamiseen ja julkaisuun. IaaS-kerroksen lisäksi PaaS sisältää tarvittavat sovellukset, joiden avulla voidaan esimerkiksi kehittää web- ja mobiilisovelluksia, tarjota tietokantapalveluja tai erilaisia julkaisualustoja loppukäyttäjien käyttöön. (Telia Inmics-Nebula, 2019)

PaaS-palvelussa palvelunkäyttäjän tulee huolehtia vain tuottamastaan sisällöstä ja palveluntoimittaja vastaa alla olevan palvelukerroksen toiminnasta.

2.2.3 IaaS – Infrastruktuuri palveluna

IaaS-palvelumallissa palvelun toimittaja vastaa vain palvelun mahdollistavan IT-infrastruktuurin ylläpidosta ja niihin liittyvistä kustannuksista. Täten Palvelun ostaja vastaa itse virtuaalisten tai fyysisten palvelimien, käyttöjärjestelmien ja niiden sovellusten ylläpidosta, kuten kuvasta 1 havaitaan. Tässä mallissa ostajalla pitää olla palvelimien käyttöjärjestelmiin ja pilviympäristöön liittyvää osaamista. (Telia Inmics-Nebula, 2019)



Kuva 1. Eri palvelumallien vastuukuvaus. (mukaillen Telia Inmics-Nebula, 2019)

2.3 Pilvimallit

Pilvipalvelut voidaan jakaa pilvimalleihin sen turvallisuusvaatimusten ja hallintamahdollisuuksien mukaan. Yleisesti ne jaetaan neljään eri kategoriaan: julkinen pilvi, yksityinen pilvi, hybridipilvi ja yhteisöpilvi.

Julkisessa pilvessä sen tarjoamat palvelut ovat julkisesti saatavilla. Myös verkko, jonka kautta sitä käytetään, on julkinen. Koko IT-infrastruktuuri sijaitsee palveluntarjoajan tiloissa. Nimensä mukaisesti yksityinen pilvi on vain yhdelle asiakkaalle rakennettu IT-ympäristö. Jos yhdistetään pilvessä oleva IT-infrastruktuuri olemassa olevaan fyysisessä konesalissa olevaan infrastruktuuriin, sitä kutsutaan hybridipilveksi, minkä ansiosta konesalin kapasiteettia pystytään helposti lisäämään ilman, että hankitaan lisää fyysistä palvelinrautaa konesaliin. Tässä mallissa yksi pilvi-infrastruktuuri on jaettu useamman yrityksen kesken. Tämä on hyödyllinen ratkaisu silloin, kun eri yritykset jakavat samanlaisia tarpeita infrastruktuurin suhteen. (Salmio, 2012)

3 Amazon Web Services -pilvipalvelu

Amazon Web Services on Amazonin kehittämä pilvialusta, joka tarjoaa mm. laskentatehoa, tiedon varastointiratkaisuja, tietokantapalveluita sekä palveluita, joiden avulla näitä hallitaan. Näiden palveluiden avulla yrityksen IT-infrastruktuuri on mahdollista ulkoistaa kokonaan pilveen.

Resursseista maksetaan oletuksena vain käytön mukaan. Erilaisista virtuaalipalvelimista maksetaan sen tehokkuuden ja kapasiteetin mukaan. Kun palvelimen sammuttaa, niin laskutus pysähtyy. Virtuaalipalvelimia on mahdollista tilata myös 1-3 vuodeksi kerrallaan. Tällä voidaan saada säästöjä, jos kyseiselle laitteelle on varmasti käyttöä koko tilausjakson ajan.

Amazonin pilvipalvelussa on yli 140 palvelua ja lisää tulee koko ajan. (Mathew, 2018) Palveluiden saatavuus vaihtelee hieman maantieteellisestä sijainnista riippuen.

3.1 Historiaa

Vuonna 2006 Amazon Web Services (AWS) alkoi tarjota IT-infrastruktuuripalveluita yrityksille verkkopalveluina. Nykyään tämä malli tunnetaan pilvipalveluna. AWS-alusta julkaistiin kuitenkin alun perin jo vuonna 2002 ilmaisena palveluna. Tuolloin sen oli vain tarkoitus auttaa ohjelmistokehittäjiä rakentamaan applikaatioita ja työkaluja hyödyntäen Amazon.comin erityisiä piirteitä. (Carey, 2019)

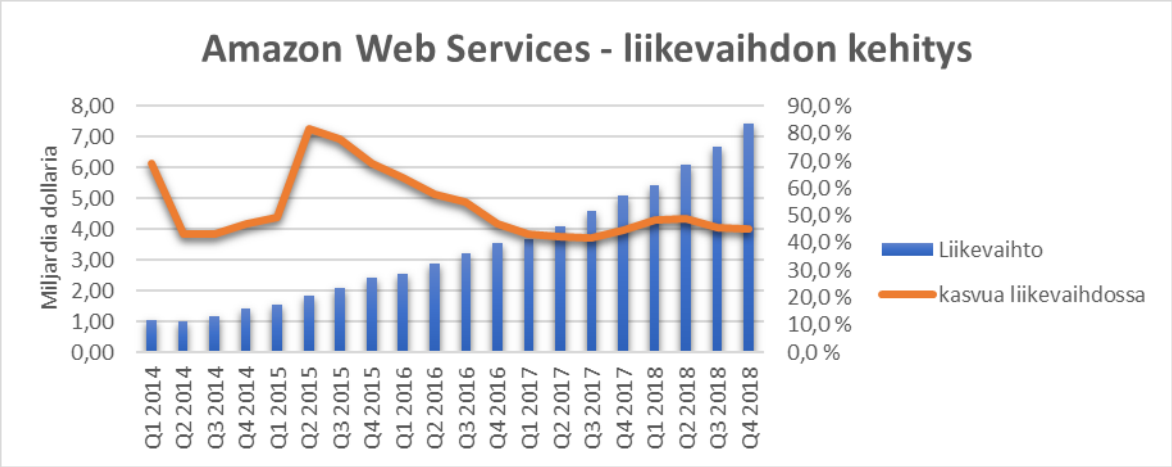
Vuonna 2006 julkaistiin ensimmäiset varsinaiset pilvipalvelut, joiden avulla yritysten oli mahdollista siirtää ohjelmistojen kehitys kokonaan Amazonin pilvi-infrastruktuuriin. Julkaistut palvelut olivat S3 (Simple Storage Service) ja EC2 (Elastic Compute Cloud). S3 on tiedonvarastoinnin pilveen mahdollistava palvelu, ja EC2 on virtuaalipalvelin, josta on nykyisin olemassa suuri määrä erilaisia variaatioita. (Carey, 2019)

Vuonna 2008 mukaan kilpailuun tuli myös Google ja vuotta myöhemmin Microsoft omilla pilvipalveluillaan. (Carey, 2019)

Amazonin pilvipalvelussa kaikkien yritysten IT-ympäristöt olivat toimineet samassa verkossa vuoteen 2009 asti, jolloin Amazon julkaisi virtuaalisen pilviympäristön (Virtual Private Cloud, VPC), jonka avulla IT-ympäristöt eristettiin tehokkaasti toisistaan. (Carey, 2019)

Amazon ei normaalisti itse rakenna asiakkaiden IT-infrastruktuuria, mutta vuonna 2013 se teki poikkeuksen, kun se teki $600 miljoonan arvoisen sopimuksen CIA:n kanssa.

Vuoteen 2015 asti Amazon ei ollut paljastanut tietoja siitä kuinka paljon rahaa se teki pilvipalveluillaan, mutta tuolloin se kertoi saavuttaneensa vuonna 2014 $4,6 miljardin liikevaihdon. Vuonna 2018 sen liikevaihto oli kasvanut jo $25,7 miljardiin (Griswold, 2019). AWS-pilvipalvelun liikevaihdon kehitystä on kuvattu alla olevassa kaaviossa 1.



Kaavio 1. AWS-pilvipalvelun liikevaihdon kehitys kvartaaleittain vuodesta 2014. (Mukaillen Griswold, 2019)

Vuonna 2018 AWS solmi $10 miljardin arvoisen sopimuksen Yhdysvaltojen puolustusministeriön Pentagonin kanssa. Liikevaihdossa mitattuna se on selkeä johtaja pilvipalvelumarkkinoilla 40% osuudellaan. Perässä seuraa Microsoft 17%, Google 8% ja Alibaba 5% osuuksillaan. (Carey, 2019)

3.2 AWS-palveluiden valvonta

IT-infrastruktuurin valvonnan avulla voidaan seurata eri resurssien käyttöasteita. Tallennetuista tiedoista voidaan luoda erilaisia raportteja asiakkaan hyödyksi. Valvonnan avulla pystytään myös seuraamaan ohjelmistojen ja palveluiden toimivuutta sekä huomaamaan mahdolliset vikatilanteet nopeasti. Sen ansiosta niihin pystytään reagoimaan nopeasti ja mahdolliset käyttökatkokset saadaan pidettyä lyhyinä ja palveluiden saatavuus korkeana.

Valvontaan voidaan määrittää erilaisia raja-arvoja eri resursseille, joiden ylittyessä valvontaohjelmisto hälyttää viasta. Se voidaan myös määrittää tekemään tiettyjä korjaustoimenpiteitä automaattisesti.

Seuraavaksi kerron hieman Amazonin pilvipalveluiden valvonnan keskeisistä palveluista.

3.2.1 Amazon CloudWatch

Amazon CloudWatch on valvonta- ja hallintatyökalupalvelu, joka tuottaa tilastoitavaa dataa AWS-pilviympäristöstä. Sen avulla voidaan valvoa myös resursseja, jotka ovat asiakkaan tiloissa, mutta ovat yhdistetty AWS-pilvipalveluun. CloudWatchin ansiosta koko ohjelmistoalustan valvonta onnistuu lähes reaaliajassa helposti yhdellä sovelluksella. Näitä ovat esim. palvelimet, verkot ja tietokannat. Vikatilanteisiin voidaan liittää automaattisia reagointitapoja, minkä ansiosta vikatilanteiden kestot lyhenevät. Sen avulla saadaan tietoa eri resurssien käyttöasteista. Näitä tietoja säilytetään jopa 15 kuukautta. Tämän ansiosta niistä on helppo luoda erilaisia historiallisia analyyseja tai muita raportteja. Myös infrastruktuurissa esiintyviä palveluun hidastavasti vaikuttavia kohteita pystytään löytämään helpommin. (Amazon Web Services)

CloudWatch valvoo yli 70 resurssia ja palvelua automaattisesti. (Amazon Web Services) Oletuksena esim. EC2-virtuaalipalvelimesta valvotaan prosessorin käyttöastetta, levyoperaatioita ja verkkoliikenteen datamäärää. Lisäksi valvontaan, järjestelmän toimivuutta yleisesti. Oletusmittausväli on viisi minuuttia, mutta se voidaan lyhentää yhteen minuuttiin. Lisäksi palveluun pystyy luomaan omia mittareitaan ja lähettämään dataa omista sovelluksistaan esim. CloudWatch agentin avulla.

Kaikesta kerätystä datasta pystytään myös luomaan graafisia mittareita omaan koottuun näkymäänsä. Tämän kaltaisen datan visualisoinnin ansiosta oman IT-infrastruktuurin tilan seuranta helpottuu huomattavasti.

3.2.2 AWS CloudTrail

CloudTrail valvoo AWS-hallintakonsolin kautta tehtyjä toimintoja sekä ohjelmistorajapinnan kautta tehtyjä toimintoja. Sen avulla pystytään selvittämään, kuka teki, mitä teki ja milloin. (Amazon Web Services) Sen avulla myös erilaisia tietoturvariskejä pystytään havainnoimaan helpommin. Se tuottaa dataa lokitiedostoina, joita voidaan lähettää suoraan CloudWatchiin tai ne voidaan tallentaa S3-ämpäriin. S3:ssa olevien lokitiedostojen dataa pystytään analysoimaan helposti Amazon Athena -palvelun avulla. Siellä olevat lokit myös oletuksena salataan automaattisesti. (Amazon Web Services)

CloudTrail on automaattisesti päällä kaikilla AWS-tileillä. Tämän ansiosta se ei vaadi minkäänlaisia konfigurointeja eikä sitä tarvitse erikseen kytkeä päälle.

3.2.3 AWS Config

AWS Config valvoo ja nauhoittaa kaikki pilviympäristössä ilmenevät konfiguraatiomuutokset. Sen avulla voidaan pitää huolta, että kaikki muutokset ovat yrityksen käytäntöjen mukaisia. Myös eri resurssien konfiguraatiotiedot voidaan helposti kerätä talteen Configin avulla. Se kertoo kuinka jonkin resurssin konfiguraatiomuutokset vaikuttavat toisiin resursseihin, mikä vähentää muutoksista aiheutuvia vikatilanteita. (Amazon Web Services)

Kun Config havaitsee jonkin konfiguraatiomuutoksen, mikä ei ole sallittu, se voi esim. lähettää siitä ilmoituksen sähköpostilla SNS-palvelun avulla. Näin järjestelmän ylläpitäjä saa siitä heti tiedon ja voi suorittaa korjaavat toimenpiteet. Config voidaan määrittää suorittamaan korjaavat toimenpiteet myös automaattisesti.

3.2.4 Amazon Simple Notification Service (SNS)

Amazon SNS ei varsinaisesti ole valvontaan liittyvä palvelu. Sen avulla voidaan kuitenkin helposti lähettää ilmoituksia erilaisista vikatilanteista esim. tekstiviestin tai sähköpostin avulla. Tämän ansiosta automaattiset ilmoitukset esim. yrityksen tiketöintijärjestelmään vikatilanteiden sattuessa onnistuvat helposti.

4 Valvontaympäristön rakentaminen

Tässä osiossa käydään läpi valvontainfrastruktuurin perustaminen AWS-pilvipalveluun. Siihen liittyvät palvelut käydään läpi pintapuolisesti, jotta lukija ymmärtää niiden merkityksen.

4.1 VPC – Virtual Private Cloud

Ennen kuin Amazon vuonna 2009 julkaisi VPC:n, kaikkien pilvipalveluympäristöt sijaitsivat samassa julkisessa verkossa. VPC:n avulla IT-infrastruktuuri on mahdollista eristää muista AWS-pilvessä olevista ympäristöistä. Se on virtuaalinen verkkoympäristö, joka muistuttaa läheisesti perinteistä fyysistä konesaliympäristöä. Suurena hyötynä kuitenkin fyysiseen konesaliin: sen kapasiteettiä on erittäin helppo laajentaa. Toisin kuin perinteisessä konesalissa, jossa laajennus vaatii huomattavasti useamman asian huomioon ottamista, kuten sähkönkulutuksen kapasiteetin kasvun huomioimisen sekä ihan fyysisen palvelinraudan tilaamista ja asentamista, johon usein kuluu kuukausia. AWS-pilvipalvelussa uusien resurssien käyttöönotto onnistuu minuuteissa.

AWS-pilvipalvelu on jaettu alueisiin. Jokaiselle alueelle tulee automaattisesti valmiina oletus-VPC ja siihen aliverkko, Internet-yhdyskäytävä, palomuuri, pääsynvalvontalista ja reititystaulu. Tässä työssä kuitenkin rakennetaan oma ympäristö VPC:stä lähtien. Näin luodaan käsitys siitä, mitä nämä eri palvelut ovat, kuinka niitä konfiguroidaan ja kuinka niitä pystytään valvomaan.

4.2 VPC-ympäristön perustaminen

VPC perustetaan North-Virginian alueelle. Se on ensimmäisiä AWS-pilvipalveluun perustettuja alueita. Tämän etuna, tälle alueelle tulee yleensä aina ensimmäisenä käyttöön uudet palvelut, joita AWS julkaisee.

Uudelle VPC:lle annettiin nimeksi MonitoringVPC. AWS-pilvipalvelussa määritettävän verkon aliverkon on oltava 16-28 bittinen. IP-osoitealueeksi määriteltiin 10.0.0.0/16, mikä on sisäverkon käyttöön varattu IP-osoitealue.

4.3 Aliverkot

Tämän jälkeen perustettiin aliverkot, jotka jakavat VPC:n pienempiin verkkosegmentteihin, joista kaksi sijaitsee julkisessa verkossa, mutta eri saatavuusalueilla (us-east-1a & us-east-1b). Kaksi muuta sijaitsevat myös eri saatavuusalueilla, mutta ne ovat yksityisessä

verkossa, josta ei ole pääsyä Internetiin. Julkisten aliverkkojen osoitealueiksi määriteltiin 10.0.1.0/24 ja 10.0.4.0/24. Yksityisten aliverkkojen vastaavat alueet olivat 10.0.2.0/24 ja 10.0.3.0/24. Julkisiin aliverkkoihin määriteltiin lisäksi päälle asetus, joka jakaa automaattisesti kaikille siinä verkossa oleville laitteille julkisen IP-osoitteen. Alla vielä havainnollistava taulukko 1 luotujen aliverkkojen asetuksista.

Aliverkot

Saatavuusalue	Yksityisyys	IPv4 CIDR	IP-osoitteita
us-east-1a	Julkinen	10.0.1.0/24	251
us-east-1b	Yksityinen	10.0.2.0/24	251
us-east-1a	Yksityinen	10.0.3.0/24	251
us-east-1b	Julkinen	10.0.4.0/24	251

Taulukko 1. Aliverkkojen tiedot.

4.4 Internet-yhdyskäytävä (IGW)

Jotta kahdesta aiemmin luodusta julkisesta aliverkosta päästään Internetiin, tarvitaan yhteyden mahdollistava Internet Gateway. Yhdellä VPC:llä voi olla vain yksi IGW-palvelu käytössä, mutta se on Amazonin hallinnoima, minkä takia se on automaattisesti vikasietoinen. Sen nimeksi annettiin monitoring-IGW, jonka jälkeen se liitettiin osaksi rakennettavaa VPC:tä.

4.5 Reititystaulut

Aina kun uusi VPC luodaan, se saa automaattisesti oletusreititystaulun. Sen ansiosta kaikki siinä VPC:ssä olevat laitteet pystyvät kommunikoimaan keskenään ilman sen suurempia konfigurointeja. Se ei kuitenkaan vielä mahdollista Internetiin pääsyä laitteille. Tätä varten julkisten aliverkkojen reititystauluihin määriteltiin vielä uudet reitit. Reitti tehtiin Internet-yhdyskäytävästä kaikkiin kyseisen aliverkon IP-osoitteisiin. Nyt julkisiin aliverkkoihin lisättäville laitteille tulee automaattisesti julkinen IP-osoite ja niiltä on pääsy Internetiin.

4.6 Palomuurit (Security Group)

Amazonin Security Group on palomuuri, jonka porttisääntöjen avulla voidaan sallia liikenne halutuille porteille. Sen kautta ei pystytä kuitenkaan estämään liikennettä minkään tietyn portin kautta, vaan siihen tarkoitukseen käytetään verkon pääsynvalvontalistoja (NACL).

Tutkimusta varten luotiin kaksi Security Group -säännöstöä, yksi julkisia aliverkkoja varten ja toinen yksityisille aliverkoille. Julkisiin aliverkkoihin sallittiin pääsy TCP-liikenteelle porttien 22 (SSH), 80 (HTTP) ja 443 (HTTPS). Yksityisiin verkkoihin pääsy sallittiin vain julkisen aliverkon osoitteista ja ainoastaan porteille 22 (SSH) ja 3306 (MYSQL). Lisäksi sallittiin ICMP-protokolla. Alla palomuurien konfiguraatioita havainnollistavat taulukot 2 & 3.

Web-DMZ

Tyyppi	Protokolla	Portti	Lähde
SSH	TCP	22	80.220.78.176/32
SSH	TCP	22	91.123.133.2/32
SSH	TCP	22	10.0.0.0/16
HTTP	TCP	80	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0

Taulukko 2. Julkisille aliverkoille tarkoitettu palomuuri.

Private-SG

Tyyppi	Protokolla	Portti	Lähde
SSH	TCP	22	10.0.0.0/16
MYSQL/Aurora	TCP	3306	10.0.0.0/16
All ICMP - IPv4	All	N/A	10.0.0.0/16

Taulukko 3. Yksityisille aliverkoille tarkoitettu palomuuri.

4.7 EC2-instanssit

Seuraavaksi luotiin EC2-instanssit. Ne ovat virtuaalipalvelimia Amazonin pilvessä. Niitä on useita eri tyyppejä, jotka sopivat erilaisiin käyttötarkoituksiin. Usein erot liittyvät laitteen suorittimien ja keskusmuistin nopeuteen sekä määrään. Tämä vaikuttaa myös laitteen hintaan. Oletuksena EC2-instansseista maksetaan niiden käytön mukaan, mutta myös määräaikaisia vuoden tai kolmen vuoden sopimuksia on mahdollista tehdä, jolloin keskimääräinen tuntiveloitus putoaa huomattavasti käytön mukaan veloitettaviin nähden.

Ensimmäisenä uutta instanssia luodessa kysytään palvelimeen asennettavaa käyttöjärjestelmää, joka asennetaan siihen valmiista levykuvasta. Amazon kutsuu näitä nimellä AMI (Amazon Machine Image). Käyttöjärjestelmäksi valittiin Amazon Linux 2, mikä on Amazonin itse tekemä Linux-versio, mikä on optimoitu EC2-instansseille.

Seuraavaksi valittiin instanssin tyyppi. Tässä tutkimuksessa kaikkien EC2-instanssien tyypiksi valittiin t2.micro, mikä sisältyy AWS:n ilmaiseen ensimmäisen vuoden kokeilusopi-

mukseen. Sen suorituskyky on kuitenkin hyvin rajoitettu sisältäen vain yhden virtuaalisen suorittimen ja yhden gigatavun keskusmuistia. Käytössä on kuitenkin suorittimen hetkellisen ylisuorittamisen mahdollistava ominaisuus, minkä ansiosta palvelimesta saadaan hetkellisiin tehotarpeisiin enemmän tehoa irti. Seuraavaksi määriteltiin mm. luotavien instanssien määrä, VPC, mihin ne sijoitetaan sekä haluttu aliverkko.

Kovalevyn kooksi määriteltiin kahdeksan gigatavua ja tyypiksi General Purpose SSD. Tämän jälkeen laite nimettiin ja valittiin jompikumpi aiemmin luoduista palomuureista, riippuen kumpaan aliverkkoon se sijoitettiin.

4.8 RDS-palvelun käyttöönotto

Tietokantainstanssien käyttöönotossa hyödynnettiin Amazonin tarjoamaa RDS-palvelua. Se tekee tietokantojen hallinnasta helpompaa, koska Amazon vastaa järjestelmän varmistuksista, päivittämisestä, vikatilanteiden havainnoimisesta ja korjaamisesta. Tämän haittapuolena on, että tietokantainstansseille ei saa muodostettua hallintayhteyttä komentorivin kautta, kuten normaaleille instansseille. Varmistuksia voi ottaa manuaalisesti tai automatisoidusti aikataulutettuna, joiden avulla tietokannan palautus vikatilanteista onnistuu luotettavasti ja tehokkaasti. Palvelu tarjoaa myös korkean saatavuuden ratkaisut helposti, jolloin luodaan kaksi tietokantainstanssia, jotka sijoitetaan eri saatavuusalueille. (What Is Amazon Relational Database Service (Amazon RDS)?) Toinen näistä instansseista toimii ensisijaisena ja toinen toissijaisena, joka otetaan automaattisesti käyttöön vikatilanteen sattuessa. Molemmilla instansseilla on omat tietokannat, jotka synkronoidaan automaattisesti keskenään.

RDS-palvelun tuetut tietokantatuotteet ovat MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server, joista kolme ensimmäistä tukevat korkean saatavuuden ratkaisua. Tietokantatyypiksi valittiin MySQL, joka kuului ilmaisen kokeilun piiriin. Halusin kuitenkin tietokantaympäristön tukevan korkeaa saatavuutta, minkä vuoksi luotiin kaksi instanssia, jotka sijaitsivat eri saatavuusalueilla. Se ei enää kuulunut ilmaiseen kokeiluun, vaan aiheutti pieniä kuluja. Tietokantamoottoriksi valittiin MySQL v. 5.6.40 ja instanssityypiksi db.t2.micro, joka vastaa tehoiltaan web-palvelimien instanssia. Kovalevyksi valittiin jälleen General Purpose SSD ja kooksi 20GiB, mikä oli minimikoko uutta tietokantaa luodessa. Maksimikoko olisi ollut 16384GiB. Kovalevyn koko vaikuttaa myös levyn suorituskykyyn: mitä suurempi levy, sen tehokkaampi se on.

Instanssit sijoitettiin aiemmin luotuun VPC-ympäristöön. Tietokantainstansseja varten luotiin aliverkkoryhmä, johon valittiin aiemmin luodut kaksi yksityistä aliverkkoa. Pääsy VPC:n

ulkopuolelta kiellettiin eivätkä instanssit näin ollen saa julkisia IP-osoitteita. Palomuuriksi valittiin aiemmin luodut privaatin aliverkon säännöstöt. Tietokannat voi halutessaan myös suojata salaamalla data, mutta valitut instanssityypit eivät tukeneet sitä, joten salausta ei otettu käyttöön. Varmistukset määritettiin otettavaksi seitsemän päivän välein klo 02 yöllä UTC-aikaan. Vain virhelokit määritettiin lähetettäväksi CloudWatchiin. Päivitysten ajankohdaksi valittiin klo 04 UTC sunnuntai-öisin.

4.9 Kuormantasaaja (ELB)

Seuraavaksi perustettiin kuormantasaaja web-palvelimia varten. Tyypiksi valittiin Application Load Balancer, mikä soveltuu parhaiten web-sovelluksille jakamaan HTTP- ja HTTPS-liikennettä. Web-sovelluksille tarkoitettu liikenne ohjataan ensin kuormantasaajalle, joka jakaa liikenteen tasaisesti julkisien aliverkon rekisteröidyille EC2-instansseille, jotka muodostavat yhdessä kohderyhmän (target group). Koska aliverkot sijaitsevat eri saatavuusalueilla, parantaa se myös palvelun saatavuutta. Kuormantasaajalle määritettiin myös kohteiden tilan tarkistus (health check), joka tehdään http-protokollaa käyttäen. Kohteeksi valittiin web-applikaation polku /index.html. Jos, kohdepalvelimella ei ole web-palvelu toiminnassa kahden peräkkäisen tarkistuksen ajan, jotka tapahtuvat 30 sekunnin välien, määrittelee kuormantasaaja sen vialliseksi, eikä enää jaa liikennettä kyseiselle instanssille. Kuormantasaaja suorittaa kumminkin tarkistuksia laitteelle jatkossakin tasaisesti, eli jos palvelin palaa toimintaan, alkaa kuormantasaaja taas jakaa sille liikennettä.

4.10 Skaalautumisryhmä EC2-instansseille

Automaattiset skaalautumisryhmät ovat yksi pilvipalveluiden hyödyistä. Sen ansiosta saavutetaan järjestelmän parempi vikasietoisuus. Automaattisen skaalauksen myötä pystytään automaattisesti käynnistämään uusia EC2-instansseja korvaamaan vialliset instanssit, kun sellainen havaitaan. Skaalautumisryhmä voidaan myös määritellä useammalle saatavuusalueelle. Jos yhdelle saatavuusalueelle tulee vikatilanne, niin uusi instanssi voidaan käynnistää automaattisesti toisella saatavuusalueella.

Automaattinen skaalautumisryhmä parantaa myös järjestelmän suorituskykyä, koska sen ansiosta voidaan automaattisesti säätää järjestelmän palvelinkapasiteettia sen hetkisen kuorman mukaan. Palvelimia voidaan automaattisesti lisätä, kun kuorma on raskaampi ja vähentää, kun kuormitus on kevyempää.

Automaattista skaalautumisryhmää käyttämällä hyödytään myös kustannushyötyjä. Useampia palvelimia käytetään vain silloin, kun sille on tarvetta ja palvelimien määrää vähen-

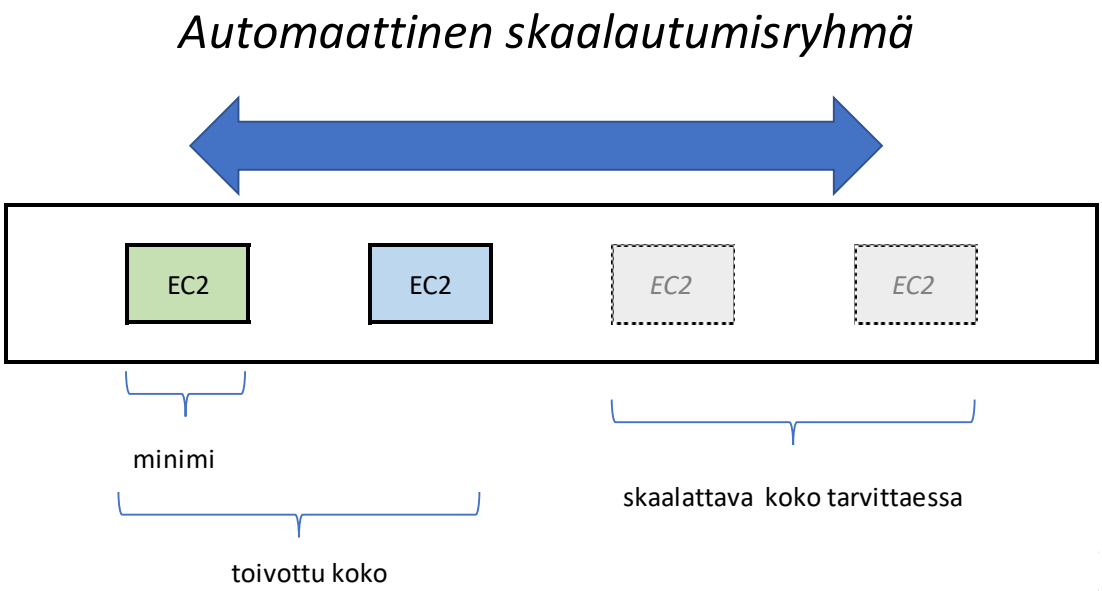
netään automaattisesti, kun niitä ei tarvita, minkä ansiosta kustannukset ovat pienemmät kiinteään palvelinmäärään verrattaessa. (Benefits of Auto Scaling. 2019)

Tätä tutkimusta varten perustettiin skaalautumisryhmä, jossa oli kaksi julkisessa aliverkossa toimivaa web-palvelinta, jotka toimivat eri saatavuusalueilla vikasietoisuuden ja saatavuuden parantamiseksi. Jotta skaalautumisryhmä saatiin otettua käyttöön, piti sitä varten ensin tehdä levykuva jo olemassa olevasta web-palvelimesta. Lisäksi piti tehdä käynnistyskonfiguraatio. Seuraavaksi käynnistyskonfiguraatio määriteltiin käyttämään tehtyä levykuvaa automaattisesti käynnistetyissä EC2-instansseissa. Ne määritettiin myös käyttämään julkisen verkon palomuuria.

Tämän jälkeen luotiin varsinainen automaattinen skaalautumisryhmä. Se määritettiin käyttämään äsken luotua uusien instanssien käynnistyskonfiguraatiota. EC2-instanssien aloitusmääräksi asetettiin yksi kappale. Toivotuksi instanssien määräksi asetettiin myös yksi kappale ja maksimimääräksi kaksi kappaletta. Verkoksi valittiin tutkimusta varten luotu VPC, ja skaalautumisryhmän käytössä oleviksi aliverkoiksi määritettiin aiemmin tehdyt kaksi julkista aliverkkoa.

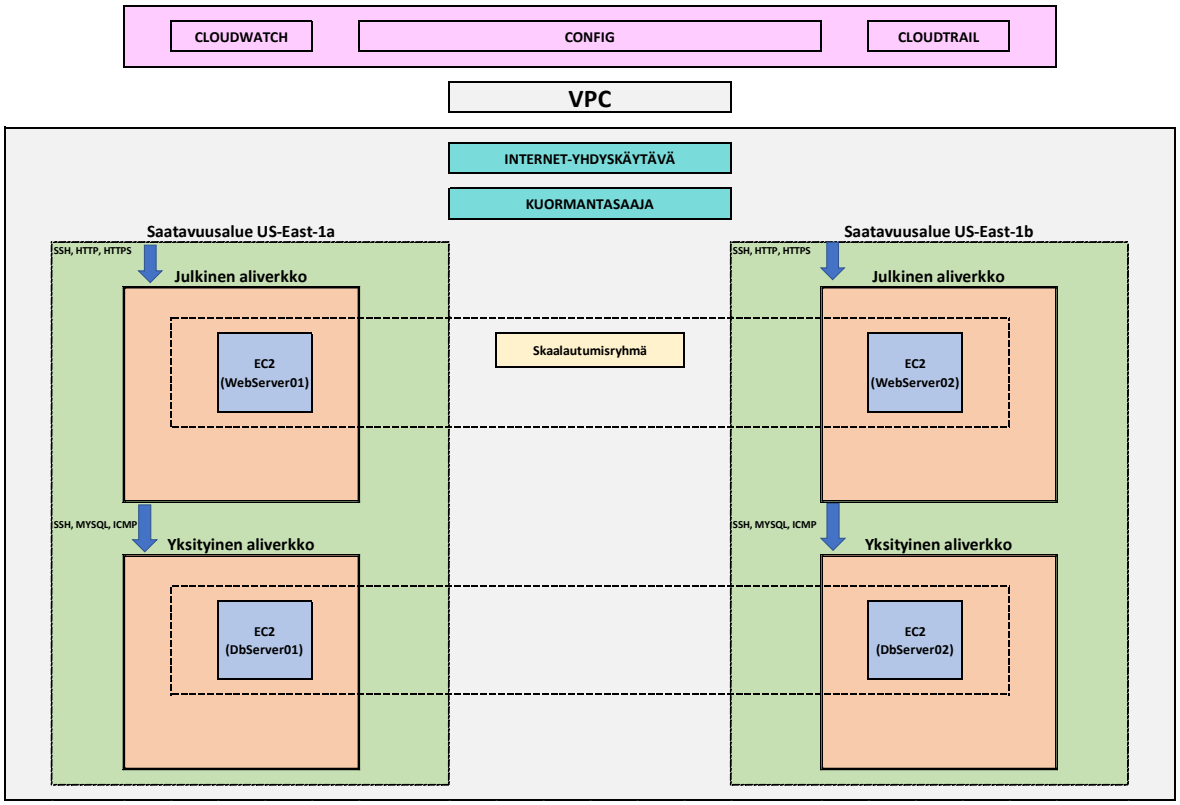
Kun skaalautumisryhmä oli luotu, sille määriteltiin skaalautumiskäytännöt. Tätä tutkimusta varten tehdyt asetukset eivät välttämättä ole optimeja tuotantoympäristössä, mutta ne nopeuttivat testauksen suorittamista. Skaalautumiskäytäntö tehtiin suoritinkuorman perusteella. Jos kuorma on yli 80% viiden minuutin seurantajakson ajan, lisätään skaalautumisryhmään yksin uusi EC2-instanssi. Palvelinmäärän automaattiseksi vähentämiseksi määriteltiin sääntö, joka poistaa käytöstä yhden EC2-instanssin, kun suoritinkuorma on ollut alle 30% viiden minuutin seurantajakson ajan.

Alla vielä selventävä kuva, kuinka skaalautumisryhmä toimii. Kuvassa on käytetty neljän palvelimen ryhmää, jossa palvelimien minimimäärä on yksi, toivottu palvelinmäärä kaksi ja kuorman kasvaessa voidaan vielä lisätä kaksi uutta instanssia (Kuva 2).



Kuva 2. Automaattisen skaalautumisryhmän toiminta.

Alla havainnollistava kuva 3 luodusta testiympäristöstä.



Kuva 3. Luotu testiympäristö.

5 Valvonnan toteuttaminen

Seuraavaksi toteutettiin valvonta aiemmin luotuun virtuaaliseen verkkoympäristöön ja siihen luotuihin komponentteihin. Mitattiin komponenttien toimivuutta erilaisilla mittareilla ja asetettiin niihin SNS-notifikaatiot sähköpostiin. Hälytyksistä on mahdollista lähettää sähköpostit mm. tiketöintijärjestelmään, jolloin saadaan luotua niistä automaattisesti tikettejä. Näin tikettien luonti on automatisoitu mm. nykyisellä työpaikallani. Meidän infrastruktuuri ei tosin sijaitse julkisessa pilvessä ja valvonta on toteutettu kolmannen osapuolen sovelluksella, mutta periaatteet ovat samat.

Eri kohteiden valvontaa saa helpotettua luomalla sitä varten kustomoituja näkymiä, missä näkyvät oleelliset valvottavat kohteet.

5.1 EC2-instanssien valvonta

Oletuksena EC2-instanssi lähettää dataa CloudWatchiin viiden minuutin välein. Lähetysten väliä on mahdollista lyhentää yhteen minuuttiin ottamalla käyttöön yksityiskohtauksen valvonnan kyseisessä instanssissa. Hyvä kuitenkin huomioida, että siitä aiheutuu ylimääräisiä kustannuksia. Kun uusi EC2-instanssi luodaan, siitä valvotaan automaattisesti mm. suorittimen käyttöastetta, levyn kirjoitus- ja lukuoperaatioiden määrää sekä verkkoliikenteen määrää. Lisäksi valvotaan virtuaalipalvelimen yleistä tilaa, mikä saattaa mennä vikatilaan, jos fyysiseen alustapalvelimeen tulee vikaa. Tällöin yleensä korjaustoimenpiteenä toimii EC2-instanssin uudelleenkäynnistys, koska se siirtyy silloin automaattisesti toiselle alustapalvelimelle.

Aiemmin mainittujen valvottavien kohteiden lisäksi olisi tärkeää valvoa myös ainakin palvelimen kiintolevyn ja keskusmuistin käyttöasteita, sillä näiden kapasiteetin loppuminen saattaa jumittaa palvelimen. Näiden valvomiseksi on kuitenkin asennettava valvottavaan instanssiin CloudWatch-agentti tai tehtävä palvelimelle erilliset skriptit, jotka lähettävät dataa CloudWatchiin.

Päädyin asentamaan palvelimille CloudWatch-agentit, joka mahdollistaa myös monen muun valvottavan kohteen, jos sille näkee tarvetta. Ensin instansseille piti vielä luoda uusi rooli, jotta CloudWatch-agentin asennus Systems Managerin kautta ja datan lähetys CloudWatchiin onnistuisi. Roolille annettiin kaksi oikeuskäytäntöä, jotka löytyivät valmiiksi luotuna AWS-pilvipalvelusta: CloudWatchAgentServerPolicy ja AmazonEC2RoleforSSM. Tämän lisäksi yhteen instanssiin liitettiin vielä CloudWatchAgentAdminPolicy, jotta se sai oikeudet lähettää CloudWatch-konfiguraation Systems Managerin Parameter Storeen tal-

teen. Sen jälkeen suoritettiin CloudWatch-agentin asennus Systems Managerin Run Command -toimintoa käyttäen, josta löytyi jo valmiiksi siihen soveltuva toiminto AWS-ConfigureAWSPackage. Tämän jälkeen luotiin CloudWatchin konfiguraatiotiedosto, jota muut instanssit voivat sen jälkeen käyttää. Konfiguraatiotiedosto luotiin siihen tarkoitetulla sovelluksella suorittamalla komento:

sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard

Se kyseli joukon kysymyksiä valvontaan liittyen. Valvonnan tasoksi määriteltiin standard, joka mahdollistaa mm. kiintolevyn vapaan tilan määrän sekä keskusmuistin käyttöasteen valvonnan. Tämän jälkeen agentin lähettämää valvontadataa alkoikin ilmestyä CloudWatchiin. Sen jälkeen asetettiin hälytys levyntäyttymiselle, joka aiheuttaa SNS-notifikaation, kun levyn täyttöaste ylittää 90%. Tämän jälkeen levylle kopioitiin dataa, niin että hälytysraja ylittyi. Hälytysviesti täyttymisestä tulikin sähköpostiin hieman yli viisi minuuttia myöhemmin.

5.2 RDS-ympäristön testaus ja valvonta

RDS-ympäristön tapahtumien seurantaan luotiin uusi tilaus (event subscription), joka määritettiin lähettämään sähköpostilla ilmoitus, jos failover tapahtuu tai jokin aiemmin luoduista tietokantainstansseista menee vikatilaan. Tämän jälkeen tietokantainstanssi käynnistettiin manuaalisesti, jonka pitäisi laukaista sähköpostihälytys sekä failover-toiminnon käynnistää automaattisesti uusi instanssi toisella saatavuusalueella. Ennen uudelleenkäynnistystä tietokantainstanssi sijaitsi us-east-1b saatavuusalueella. Uudelleenkäynnistyksen jälkeen lokien perusteella nähtiin, että palvelun siirto toiselle saatavuusaluelle aloitettiin heti uudelleenkäynnistyksen jälkeen ja tietokanta oli taas toiminnassa n. minuutin kuluttua. Uusi saatavuusalue (us-east-1a) päivittyi näkymään kuitenkin vasta n. 10min kuluttua. Sähköpostihälytystä palvelun siirrosta ei saatu. Tein kokonaan uuden SNS-aiheen tätä varten ja kun siitä tuli hyväksymispyyntö sähköpostiin, niin samalla kaikista aiemmista failover-tapahtumistakin tuli ilmoitukset sähköpostiini. Tässä oli ilmeisesti ollut jokin katkos Amazonin järjestelmässä, koska muut viestit olivat kuitenkin tulleet perille sähköpostiini tällä välin.

RDS-ympäristöön on mahdollista lisätä hälytyksiä helposti sitä varten valmiiksi tehdyistä hälytyksistä, kuten tietokannan täyttymisestä. CloudWatchin kautta RDS-näkymässä on automaattisesti 17 kohdetta valvonnassa, mutta yhteensä statistiikkaa on saatavilla peräti 363 eri kohteesta. Näiden kautta onnistuu myös mahdollisten suorituskykyongelmia aiheuttavien pullonkaulojen selvitys.

5.3 Kuormantasaajan valvonta

Kuormantasaajasta valvotaan automaattisesti 63 eri kohdetta. Näistä oleellisimmat ovat kuitenkin esim. viallisen kohdepalvelinten määrä, joille kuormantasaaja jakaa liikennettä. Kohdepalvelimissa valvottiin index.html sivuston saatavuutta. Lisäksi, kun on kyse web-applikaatiosta, niin oleellista HTTP 5xx vastausten määrä. Ne viittaavat, palvelinpuolen virhetilaan ja sen saatavuuteen, minkä vuoksi näiden määrää on hyvä pitää silmällä ja määrittää niille hälytykset, jotka lähettävät automaattiset SNS-notifikaatiot. Lisäksi on hyvä valvoa myös HTTP-pyyntöjen viivettä ja asettaa hälytys, jos se nousee kovin korkeaksi. Sitä ei tässä tutkimuksessa tosin testattu, mutta HTTP 5xx ja viallisten kohdepalvelimien määrälle asetettiin hälytykset. Sen jälkeen ainoa päällä ollut web-palvelininstanssi käynnistettiin uudelleen, mikä aiheutti katkoksen web-sivuston saatavuuteen. Samaan aikaan yritettiin sivustoa avata selaimella, mihin tuli vastauksena 502 Bad Gateway -virheilmoitus ja myöhemmin vielä 504 Gateway Time-out. Sähköpostiin tuli myös lähes välittömästi ilmoitus HTTP 5xx vastausten hälytysrajan ylittymisestä sekä viallisesta kohdepalvelimesta.

Kuormantasaajalle tulevia HTTP-pyyntöjä on mahdollista seurata myös pääsylokien kautta. Niistä näkee mm. pyynnön ajankohdan, pyynnön tehneen koneen IP-osoitteen, Internet-selaimen sekä käyttöjärjestelmäversion.

5.4 Automaattisen skaalautumisryhmän valvonta

Ensimmäiseksi testattiin, kuinka nopeasti uusi toimiva instanssi on käytössä, siitä kun instanssi sammuu syystä taikka toisesta. Tämä testi suoritettiin yksinkertaisesti sammuttamalla ainut päällä ollut Web-palvelininstanssi. Sen jälkeen otin aikaa, kuinka kauan menisi, että web-sivusto on jälleen toiminnassa. Lisäksi määrittelin SNS-notifikaatiot aina, kun skaalautumisryhmä käynnistää uuden instanssin, poistaa instanssin tai jompikumpi aiemmista epäonnistuu.

Testi suoritettiin muutamaan otteeseen, mutta tulokset olivat lähes identtisiä. Siitä, kun web-palvelin sammutettiin, meni n. kolme minuuttia ja 30 sekuntia, että web-sivusto oli jälleen toiminnassa ja korvaava web-palvelininstanssi toimintakunnossa. Tästä voi päätellä, että jos halutaan varmistaa, että web-sivusto olisi koko ajan saatavilla, niin olisi syytä pitää vähintään kaksi palvelininstanssia päällä minimissään, jotka sijaitsevat eri saatavuusalueilla. Se luonnollisesti kasvattaa myös kuorman määrää, minkä web-palvelin kestää eikä se näin ollen ole niin altis hetkittäisille kävijäpiikeille sivustolla. Jos kuitenkin ollaan tiukalla budjetilla ja muutaman minuutin katkokset sallitaan, niin yhdelläkin palvelimella pärjätään.

Seuraavaksi testattiin skaalautumisryhmän automaattisen skaalautumisen toimivuutta palvelimen suorittimen kuormaa keinotekoisesti kasvattamalla. Tämä tehtiin Stress-ohjelmaa käyttämällä. Ohjelma vaati ensiksi epel-kirjaston käyttöönoton, jonka jälkeen asennettiin itse sovellus seuraavalla komennolla:
sudo amazon-linux-extras install epel -y && yum install stress -y

Tämän jälkeen suoritinta kuormitettiin komennolla:
sudo stress -t 600 -c 8

Koska tässä uuden instanssin luomiseen käytettiin CloudWatch-hälytyksiä, jonka tarkistusväli oli viisi minuuttia, meni uuden instanssin käynnistämiseen hieman yli viisi minuuttia. Tämän jälkeen suorittimen kuormittaminen lopetettiin ja tutkittiin, kuinka kauan toisen instanssin terminointiin kuluu aika. Skaalautumisryhmä havaitsi kuorman laskeneen palvelimilla jälleen hieman yli viiden minuutin kuluttua, mutta toisen instanssin poistoon kului vielä toiset viisi minuuttia, joten yhteensä koko operaatioon kului aikaa n. 10 minuuttia. Jotta valvonta toimisi mahdollisimman tehokkaasti, onkin suositeltavaa määrittää CloudWatchiin yhden minuutin tarkistusväli.

SNS-ilmoitukset toimivat hyvin testien osalta ja uuden instanssin luomisesta ja poistamisesta tuli ilmoitukset välittömästi. Skaalautumisryhmä teki myös uudet instanssit joka kerta oikeaoppisesti toiselle saatavuusalueelle.

CloudWatchissa skaalautumisryhmälle on kahdeksan mittaria, jotka näkyvät konsolissa skaalautumisryhmän monitor-välilehdellä. Näitä ovat mm. skaalautumisryhmän instanssien kokonaismäärä, poistettavien ja toivottu instanssien määrä.

5.5 Ympäristön valvonta Configilla

Config ei ole oletuksena päällä, vaan se on saatavilla maksullisena lisäpalveluna. Sen toimintaa testattiin kytkemällä se päälle. Kytkennän yhteydessä voi valita yksittäisiä resursseja, mitä valvotaan tai sitten määrittää se valvomaan koko infrastruktuuria. Tutkimuksessa valittiin jälkimmäinen vaihtoehto. Testaukset suoritettiin ottamalla käyttöön sääntö, jonka mukaan SSH-portti avoimuus pitää olla rajoitettu palomuurisäännöissä. Jos näin ei ole, siitä lähetetään jälleen SNS-ilmoitus sähköpostilla.

Testissä määritettiin yksityisen aliverkon palomuurin SSH-portin lähdeosoitteeksi 0.0.0.0/16 ja katsottiin, kauanko menee, että se aiheuttaa hälytyksen Configissa. 20 minuutin odottelun jälkeen Config näytti konsolissa virheellisesti konfiguroidusta palomuurista, jossa oli myös tarkemmat tiedot resurssista sekä muutoksen ajankohta. Ajankohdaksi tosin oli merkitty aika, jolloin Config teki kyseisen havainnon, eikä todellinen muutoksen tekohetki. Tämän jälkeen SSH-asetus muutettiin entiseen rajoitettuun tilaan ja testattiin, kuinka kauan muutoksen havaitsemiseen jälleen menee. Vaikka virheellisesti määritellyn resurssin löytymisen jälkeen siihen olikin mahdollista kohdistaa suoraan säännön uudelleenarviointi, niin siltikin muutoksen havaitseminen kesti sen saman 20 minuuttia.

Muutosten havaitseminen oli siis aika hidasta. Erittäin hyödyllinen ominaisuus Configissa on kuitenkin tuo muutoksentekohetken näkeminen, vaikka se ei tismalleen oikea olekaan. Sen kautta on myös helppo katsoa kaikkien resurssien muutoshistoriaa pidemmältäkin aikaväliltä. Halutessaan voi myös kirjoittaa vaikka lambda-funktion, mikä korjaa havaitut epäkohdat automaattisesti.

6 Pohdinnat

Valvonnan toteutus kattoi toteutuksen pienimuotoiseen infraan. Kaikkien palveluiden valvonnan toteutusta ei ollut mahdollista toteuttaa tässä tutkimuksessa ajan puutteen takia, koska valvottavia palveluita on niin suuri määrä. Pääsääntöisesti valvonta kuitenkin muihinkin palveluihin toteutetaan samoja työkaluja käyttäen, joten muidenkin palveluiden valvonnan toteutukseen tästä tutkimuksesta on varmasti hyötyä. Amazonin pilvipalveluun tulee tiuhaan tahtiin uusia palveluita ja vanhoihin palveluihin tulee uusia ominaisuuksia, minkä vuoksi tästä tutkimuksesta saadut tulokset saattavat vaatia päivittämistä jo lähitulevaisuudessa.

Käytetyt tutkimusmenetelmät palvelivat hyvin tutkimusta ja niiden avulla saatiin kerättyä tietoa, kuinka eri palvelut toimivat vikatilanteen sattuessa ja miten niihin on mahdollista reagoida. EC2-instanssien eri komponentteja olisi voinut vielä testata monipuolisemmin, mutta en usko sen tuovan kovinkaan paljoa lisäarvoa, sillä tutkimustavat olisivat olleet silti samat kuin nyt suoritetuissa testeissä.

Yleisesti ottaen Amazonin pilven omat valvontatyökalut tuottivat pienoisen pettymyksen. Toimivaan valvontaratkaisuun joutui yhdistelemään useita eri palveluita, mikä hankaloittaa valvonnan käyttöönottoa verrattuna työssäni käyttämään N-Central-valvontasovellukseen, joka tarjoaa kaiken tarvittavan yhdessä sovelluksessa. Valvonnan toteuttaminen Amazonin tarjoamilla työkaluilla vaatii myös huomattavaa paneutumista niihin etukäteen, jotta tietää, mitä palveluita eri komponenttien valvontaan kannattaa käyttää ja valvonnasta saa kaiken hyödyn irti. EC2-instanssien yksityiskohtainen valvonta vaatii aika paljon manuaalista työtä, koska esim. yksittäisiä palveluja ei ole mahdollista valvoa, ellei jokaiselle luo skriptiä, mikä sitä valvoo ja lähettää dataa CloudWatchiin. N-Centralissa on suuri määrä valmiita pohjia, jotka sisältävät valvonnan tietyn roolin palvelimille ja niitä voi määrittää useampia yhdelle palvelimelle. Lisäksi agentti valvoo palvelimen kaikkia palveluita ja prosesseja automaattisesti ja niiden lisääminen valvottavien palveluiden listalle onnistuu yhdellä hiiren klikkauksella.

Positiivista oli kuitenkin CloudWatchin tarjoama lukematon määrä erilaisia mitattavia kohteita, joiden avulla esim. laitteen suorituskyvyn analysointi onnistuu helposti ja infrastruktuurin mahdolliset ongelmakohdat on helpompi havaita. Palvelimien uudelleenkäynnistykset eivät jää huomaamatta, koska palvelimen valvonta ei perustu tietyin aikavälein tehtävään kyselyyn, toisin kuin N-Centralissa. Siinä usein jää esim. palvelimen uudelleenkäyn-

nistyminen havaitsematta valvonnassa, koska virtuaalipalvelimet käynnistyvät niin nopeasti uudelleen, että esim. kahden minuutin skannausvälillä sitä ei välttämättä huomata. Lähtökohtaisesti kuitenkin suosittelisin jonkin kolmannen osapuolen valvontasovelluksen käyttöä niiden helpomman käyttöönoton takia. Jos kuitenkin virittelyhaluja riittää, niin myös Amazonin omilla työkaluilla on mahdollista rakentaa toimiva valvonta IT-infrastruktuurille.

Lähteet

Amazon Web Services. Amazon CloudWatch Features. Luettavissa: https://aws.amazon.com/cloudwatch/features/ Luettu: 17.3.2019

Amazon Web Services. AWS Config concepts. Luettavissa: https://docs.aws.amazon.com/config/latest/developerguide/config-concepts.html Luettu: 17.5.2019

Amazon Web Services. Benefits of Auto Scaling. Luettavissa: https://docs.aws.amazon.com/autoscaling/ec2/userguide/auto-scaling-benefits.html Luettu: 21.4.2019

Amazon Web Services. What is Amazon CloudTrail? Luettavissa: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-user-guide.html Luettu: 17.5.2019

Amazon Web Services. VPC Network Management and Monitoring. Luettavissa: https://aws.amazon.com/answers/networking/vpc-network-management-and-monitoring/ Luettu: 25.2.2019

Amazon Web Services. What Is Amazon Relational Database Service (Amazon RDS)? Luettavissa: https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Welcome.html Luettu: 22.4.2019

Carey, S. 2019. The history of AWS: A timeline of defining moments from 2002 to now. Luettavissa: https://www.computerworlduk.com/galleries/cloud-computing/aws-defining-moments-for-the-cloud-giant-3636947/ Luettu: 17.3.2019

Griswold, A. 2019. Amazon Web Services brought in more money than McDonald's in 2018. Luettavissa: https://qz.com/1539546/amazon-web-services-brought-in-more-money-than-mcdonalds-in-2018/ Luettu: 17.3.2019

Lucifredi, F., Ryan, M. 2018. AWS System Administration. 1. painos. O'Reilly Media.

Markoff, S. 2018. Pilvipalveluiden valvonnan kehittäminen yrityksessä. Luettavissa: https://www.theseus.fi/bitstream/handle/10024/146711/Opinnaytetyo_Markoff.pdf?sequence=1 Luettu: 25.2.2019

Mathew, S. 2018. Overview of Amazon Web Services. Luettavissa: https://docs.aws.amazon.com/aws-technical-content/latest/aws-overview/introduction.html Luettu: 17.3.2019

Pilvi.com. Mikä on SaaS-palvelu? Luettavissa: https://www.pilvi.com/fi/mika-on-saas-palvelu/ Luettu: 17.3.2019

Salmio, P. 2012. Pilvipalvelut. Luettavissa: https://www.theseus.fi/bitstream/handle/10024/41634/Salmio_Petri.pdf?sequence=1&isAllowed=y Luettu: 17.5.2019

Telia Inmics-Nebula. Pilven monet kasvot – IaaS, PaaS ja SaaS. Luettavissa: https://www.inmicsnebula.fi/fi/blogi/pilven-monet-kasvot-iaas-paas-ja-saas?language_content_entity=fi Luettu: 17.3.2019