jamk.fi

Wazuh SOC-ympäristössä Linux-näkyvyyden lisäämiseen

Tomi Särkisaari

Opinnäytetyö
Maaliskuu 2020
Tekniikan ala
Insinööri (AMK), Tieto- ja viestintätekniikka

Jyväskylän ammattikorkeakoulu
JAMK University of Applied Sciences

jamk.fi

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Särkisaari, Tomi	Opinnäytetyö, AMK	Maaliskuu 2020
	Sivumäärä 48	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x

Työn nimi
Wazuh SOC-ympäristössä Linux-näkyvyyden lisäämiseen

Tutkinto-ohjelma
Tieto- ja viestintätekniikka, Kyberturvallisuus

Työn ohjaaja(t)
Ari Rantala, Esa Salmikangas

Toimeksiantaja(t)
Telia Cygate Oy

Tiivistelmä

Opinnäytetyön tavoitteena oli selvittää toisiko Wazuh security platform tarvittavaa lisäarvoa SOC-ympäristössä Linux näkyvyyden lisäämiseen.

Lisäksi tavoitteena oli saada riittävästi tietoa Wazuhin tuomasta mahdollisesta lisäarvosta otettaessa huomioon Wazuhin käyttöönotto, luotettavuus, skaalautuvuus ja mahdollinen tarve ylläpidolle. Opinnäytetyössä perehdyttiin Wazuhin ominaisuuksiin myös siltä kannalta, olisiko Wazuhilla mahdollista tuottaa tarvittavaa lisänäkyvyyttä Linux-laitteisiin ja -järjestelmiin Telia Cygate Oy:n omassa SOC-ympäristössä ja mahdollisesti tulevaisuudessa myös asiakasympäristöissä.

Näkyvyyden lisääminen Linux-käyttöjärjestelmiin on tärkeää, sillä monien yritysten verkoissa on nykyään paljon Linux-laitteita, joiden tilasta, haavoittuvuuksista sekä niihin kohdistuvista hyökkäyksistä ei ole tehokasta lokitus- ja tai seurantajärjestelmää. Opinnäytetyössä arvioitiin, kuinka Wazuh security platformia voitaisiin käyttää juurikin Linux-laitteiden lokien keräämiseen, lokin jäsentämiseen sekä lokissa esiintyvien epäkohtien nostamiseen herätteiksi.

Opinnäytetyön tuloksena saatiin arvio Wazuhista sen lisäarvoa tuovien ominaisuuksien kannalta. Lopuksi myös avataan opinnäytetyötä tehdessä törmättyihin ongelmiin niin testauksen kuin laitteiston tarvittavan kapasiteettiin liittyen ja siihen, miten nämä ongelmat saataisiin ratkaistua tulevaisuudessa.

Opinnäytetyö antaa pohjaa Wazuhin käyttöönottamiseksi ja sitä voidaankin hyödyntää Wazuhia käyttöönottaessa.

Avainsanat (asiasanat)

Wazuh, SOC, kyberturvallisuus, tietoturvapoikkeama, näkyvyys

Muut tiedot (Salassa pidettävät liitteet)

jamk.fi

Description

Author(s)	Type of publication	Date
Särkisaari, Tomi	Bachelor's thesis	March 2020
		Language of publication: Finnish
	Number of pages: 48	Permission for web publication: x

Title of publication

Wazuh in SOC environment for Linux visibility enhancement

Degree programme

Information and Communication Technology, Cybersecurity

Supervisor(s)

Rantala Ari, Salmikangas Esa

Assigned by

Telia Cygate Oy

Abstract

The bachelor's thesis was assigned by Telia Cygate Oy. The aim was to find out if the Wazuh security platform would add the necessary value in the SOC environment for increasing the visibility of Linux.

In addition, the purpose to gain enough knowledge of the potential added value of Wazuh, considering Wazuh's deployment, reliability, scalability and possible maintenance needs. Wazuh's features in terms of enabling Wazuh to provide the necessary additional visibility into Linux devices and systems in Telia Cygate Oy's own SOC environment and possibly in the future in client environments as well.

Increasing visibility into Linux operating systems is important because many enterprise networks today have a large number of Linux devices without an effective an effective logging or monitoring system for their status, vulnerabilities, and attacks. The thesis evaluated how the Wazuh security platform could be used to collect logs for Linux devices, parse the log, and to trigger log alerts that SOC analysts can then investigate.

The thesis resulted in an evaluation of Wazuh for its value-adding features. Finally, the problems encountered while working on the thesis will be opened, both in terms of testing and hardware capacity and how these problems can be resolved in the future.

This thesis provides a basis for deploying Wazuh and can be utilized in case deploying Wazuh.

Keywords/tags (subjects)

Wazuh, SOC, Cyber security, Security incident, visibility

Miscellaneous (Confidential information)

Sisältö

Kuviot

1 Lähtökohdat

1.1 Toimeksiantajana Telia Cygate Oy

Telia Cygate Oy on tietotekniikan alan yritys, joka on erikoistunut tietoturvallisten palveluiden sekä laitteiden ylläpitoon ja myyntiin. Yritys tarjoaa esimerkiksi asiakasratkaisuja ohjelmistoihin, lohkoketjuihin, identiteetinhallintaan, tietoverkkoihin, tietoturvaan sekä konesali- ja pilvipalveluihin. Telia Cygate Oy tarjoaa vuorokauden ympäri päivystävää palvelua esimerkiksi ylläpidossa olevien laitteiden toiminnan seuraamiseksi sekä security operations center eli SOC-palvelua tietoturvapoikkeamien seuraamiseen ja niistä raportoimiseen. (Telia Cygate Oy lyhyesti n.d.a.)

Telia Cygatella on yli 400 työntekijää, joilla on lähes 600 erilaista IT-alan sertifikaattia. Toimipisteitä on ympäri Suomea: Helsingissä, Jyväskylässä, Kouvolassa, Lappeenrannassa, Oulussa, Tampereella, Lahdessa, Kotkassa ja Turussa. Telia Cygate on arvostettu IT-ratkaisujen tarjoaja ja merkittävä osa Teliaa. (Telia Cygate Oy lyhyesti n.d.b.)

1.2 Toimeksianto

Toimeksiantona oli testata ja analysoida avoimen lähdekoodin tietoturvaratkaisua, Wazuhia, sen mahdollisten lisäarvoa tuovien ominaisuuksien kannalta Telia Cygate Oy:n SOC:in toiminnassa.

Opinnäytetyössä käydään läpi teoriaa liittyen SOCiin, Wazuhiin sekä Wazuhin asennusprosessi ja lopuksi arvioidaan kyseisen työkalun ominaisuuksia SOC-toimintaa silmällä pitäen.

1.3 Opinnäytetyön tavoitteet

Opinnäytetyön tavoitteena oli saada riittävästi tietoa Wazuhin tuomasta mahdollisesta lisäarvosta, käyttöönotosta, luotettavuudesta, skaalautuvuudesta ja ylläpidosta. Opinnäytetyössä arvioitiin myös sitä, onko Wazuh security platformin avulla

mahdollista tuottaa tarvittavaa lisänäkyvyyttä Linux-laitteisiin ja -järjestelmiin Telia Cygate Oy:n omassa SOC-ympäristössä sekä mahdollisesti tulevaisuudessa myös asiakasympäristöissä.

Näkyvyyden lisääminen Linux-käyttöjärjestelmiin on tärkeää, sillä useiden yritysten verkoissa on paljon Linux-laitteita, joiden tilasta, haavoittuvuuksista sekä suorien hyökkäysten toiminnasta ei ole olemassa tehokasta lokitusjärjestelmää. Opinnäytetyön tavoitteena oli tutkia, kuinka Wazuh security platformia voitaisiin käyttää Linux laitteiden lokin keräämiseen, jäsentämiseen sekä siinä esiintyvien epäkohtien ja tietoturvapoikkeamien herätteiksi nostamiseen.

Opinnäytetyössä käydään läpi erityisesti sitä, kuinka suuri lisäarvo Wazuhin käyttöönottoon ja ylläpitoon liittyy suhteessa mahdolliseen lisäarvoon. Lisäksi opinnäytetyön tavoitteena oli sisällyttää arvio siitä, onko Wazuhia mahdollisuus ottaa käyttöön tai hyödyntää kustannustehokkaalla tavalla myös asiakasympäristöissä. Lopputuloksena saadaan arvio Wazuhin mahdollisesta sopivuudesta yhdeksi SOC-ympäristöä tukevaksi järjestelmäksi. Opinnäytetyössä ei kuitenkaan käyty Linux-asennusta tai siihen liittyvää teoriaa läpi.

2 Security Operations Center (SOC)

2.1 Security Operations Center yleisesti

Security Operations Center (SOC) on keskitetty tietoturvan ja kyberturvallisuuden palveluihin erikoistunut valvomo, jossa seurataan organisaatioiden tietoturvaa niin yleisellä kuin syvemmälläkin tasolla. Uhkia pyritään aktiivisesti tunnistamaan, seuraamaan sekä tarvittaessa puuttumaan niihin. Näin pyritään takaamaan organisaatioiden toiminnan jatkuvuus myös kyber- tai tietoturvapoikkeamatilanteissa. (Combitech Finland 2017a.)

Yleisimmin tunnetut haittaohjelmat ja hyökkäysrajapinnat, jotka on jo kirjattu yleisiin uhkatietokantoihin, ovat yleensä estettävissä uuden sukupolven eli niin kutsutuilla Layer 7 -palomuureilla sekä yleisimmillä virustentorjuntaohjelmistoilla.

Next generation -palomuurilla eli uuden sukupolven palomuurilla tarkoitetaan palomuurityyppiä, joka pystyy esimerkiksi applikaatiotasolla puuttumaan ja reagoimaan yleisessä dataliikenteessä oleviin poikkeamiin. Tämän tyyppisiä palomuuriratkaisuja on nykypäivänä yleisesti käytössä, sillä ne pystytään asentamaan siten, että palomuurit toimivat myös IDS- tai IPS-järjestelmän tavoin.

IDS- ja IPS -järjestelmä -lyhennelmät pohjautuvat sanoihin Intrusion Detection System (IDS) ja Intrusion Prevention System (IPS). Alun perin IDS -ja IPS-järjestelmät luotiin haavoittuvuuksiin kohdennettujen hyökkäyksien, kohdesovellusten tai suorien tietokoneita vastaan toteutettujen hyökkäysten seurantaan. Karkeasti määriteltynä IDS-järjestelmä eroaa IPS-järjestelmästä siten, että IDS-järjestelmä tunnistaa edellä mainittuja tietoturvapoikkeamia verkkoliikenteestä niihin kuitenkaan reagoimatta. IPS-järjestelmä tunnistaa tietoturvapoikkeamia samalla tavalla kuin IDS-järjestelmäkin, mutta pystyy tämän lisäksi myös puuttumaan niihin. Tietoturvapoikkeamiin puuttuminen tapahtuu esimerkiksi siten, että järjestelmä automaattisesti estää hyökkääjän käyttämästä IP-osoitteesta tulevan liikenteen kohteeseen. (What is an Intrusion Detection System? n.d.)

Nykypäivän tietoturva- ja kyberuhkat kuitenkin muuttuvat ja kehittyvät jatkuvasti. Uhkatietokantoihin ei aina pystytä raportoimaan uusimmista hyökkäyskanavista tai mahdollisista haavoittuvuuksista, mikäli kyseistä hyökkäys- tai hyväksikäyttömenetelmää ei ole vielä havaittu tai tunnistettu tarpeeksi laajalti ja tarkasti. Lisäksi monissa tapauksissa uhkatietokantaan esimerkiksi haavoittuvuudesta merkinnän saaminen voi hidastua, jos hyökkääjä tai haavoittuvuuden löytäjä ei ole tyytyväinen kyseisen haavoittuvan tuotteen tai laitteen luoneelta yritykseltä tai organisaatiolta saamaansa palkkioon. Näissä tapauksissa on valitettavan usein maailmalla nähty niin sanottujen nollapäivä-haavoittuvuuksien hyväksikäyttöä, mikä on huomattu vasta liian myöhään. Mustassa pörssissä eli esimerkiksi Tor-verkossa sijaitsevilla kauppapaikoilla

hakkerit voivat saada haavoittuvuuksista huomattavastikin suurempia palkkioita kuin tuotteen tai laitteen tehneeltä yritykseltä.

Nollapäivä-haavoittuvuudella tarkoitetaan haavoittuvuutta, jolle ei toistaiseksi ole vielä olemassa korjausta, mutta jolle kuitenkin löytyy olemassa oleva hyväksikäyttömenetelmä. Nollapäivä -haavoittuvuuden nimi viittaa siihen, montako päivää korjauksen jälkeen haavoittuvuuden hyväksikäyttömenetelmä julkaistaan. Toisin sanoen, nollapäivä -haavoittuvuuden hyväksikäyttömenetelmän tietoon tulon jälkeen tai sen yhteydessä saadaan tieto itse haavoittuvuudesta. Vasta tämän jälkeen voidaan aloittaa korjaustoimet haavoittuvuuden korjaamiseksi, mikäli valmistaja ei ole siitä ollut vielä tietoinen. (Rouse 2019a.)

Tietomurtojen, uusien haavoittuvuuslöytöjen ja aktiivisesti toimivien hakkereiden takia kaikki organisaatiot tarvitsevat jatkuvaa ympäristön valvontaa sekä uhkatilanteiden seurantaa. Lisäksi organisaatiot tarvitsevat koulutettuja ammattilaisia havaitsemaan ja torjumaan erilaiset tieto- ja kyberturvallisuuden uhat. Vain tällä tavoin voidaan saavuttaa paras mahdollinen kyky puolustautua nykypäivän kyberrikollisia vastaan. (Combitech Finland 2017b.)

2.2 Toiminta

Palvelun kattavuus ja siellä olevan toiminnan laajuus riippuu usein esimerkiksi asiakkaan tarpeesta palvelulle tai siitä, mitä SOC:issa on tarkoitus seurata. Mikäli SOC -palvelu on ostettu esimerkiksi vain sähköpostiliikenteen seurantaan ja siellä liikkuvien haitallisten sähköpostien seuraamiseen, ei tarvetta kellon ympäri vuoden jokaisena päivänä toimivalle (24/7) SOC -palvelulle välttämättä ole.

24/7 SOC -palvelua tarjottaessa SOC -analyytikot työskentelevät vuoroissa, jotta palvelu voi toimia keskeytyksettä vuorokauden jokaisena hetkenä vuoden jokaisena päivänä. Yleensä niin yö- kuin päivävuoroissakin on tarpeen mukaan vähintään yksi tai useampi SOC -analyytikko.

Vuorokierrossa toimivat SOC -analyytikot kuuluvat yleensä niin sanottuun Layer 1 -tasoon. Tämä tarkoittaa sitä, että SOC -analyytikot ovat suorassa kontaktissa sisään tulevien herätteiden kanssa ja analysoivat, ovatko herätteet niin kutsuttuja false positive -herätteitä (FP) vai true positive -herätteitä (TP). Toisin sanoen, ovatko herätteet aiheellisia vai aiheettomia. Mikäli ensimmäisen tason (Layer 1) analyytikko kokee herätteen olevan aiheellinen, siirretään se yleensä se käsiteltäväksi ylemmän tason analyytikolle eli L2 -tason analyytikolle (layer 2), joka yleensä vastaa tietoturvapoikkeaman kohteena olevan organisaation ympäristöstä ja käsittelee herätteen tehden sille vaadittavat toimenpiteet.

Monissa nykypäivän SOC -ympäristöissä myös niin sanottu taso kolme eli L3 (layer 3), joka taas vastaa käytettyjen ympäristöjen, ohjelmistojen sekä laitteiden toimivuudesta yhdessä asiakkaiden kanssa sekä siitä, mitä herätteitä eri ympäristöistä pystytään saamaan.

Paras hyöty SOC:n toiminnasta saadaan, kun näkyvyys laitteisiin ja hallinnoitaviin ympäristöihin on riittävän hyvä sekä silloin, kun palvelu on käytettävissä aina, eli kellon ympäri vuoden jokaisena päivänä. Tämä siksi, että hyökkäysten ajankohdat eivät välttämättä sijoitu vain Suomen aikavyöhykkeen päiväaikaan. Esimerkiksi kyberturvallisuusrintamalla aktiivisesti työllistävien maiden, kuten Venäjän, Kiinan ja USA:n aikavyöhykkeet ovat sen verran kaukana Suomen aikavyöhykkeestä tunneissa mitattuna, että siellä päivällä tehdyt hyökkäykset voivat näkyä Suomessa esimerkiksi yöllä tai arkipyhinä. Toki myös strategisesti voi joissain tapauksissa hyökkääjän kannalta ajateltuna olla parempi ajastaa hyökkäys siihen ajankohtaan, jolloin puolustuskyky ei oletettavasti ole parhaimmillaan monissa organisaatioissa eli yöllä tai pyhäpäivinä.

Tämän opinnäytetyön tarkoituksena olikin arvioida avoimen lähdekoodin tietoturvajärjestelmää Wazuhia sen näkyvyyttä lisäävien ominaisuuksien, käytettävyyden, skaalautuvuuden sekä luotettavuuden kannalta. Tarkoituksena oli juuri Linux -laitteisiin liittyvän näkyvyyden lisääminen.

2.3 Tehtävä

Security Operations Center:in tehtävä on toimia osana organisaatioiden riskinhallintaa. Nykypäivänä yrityksiin ja organisaatioihin kohdistuu jatkuvasti hyökkäyksiä, sekä hyväksikäyttöyrityksiä. SOC:n tehtävänä onkin näiden mahdollisten uhkatekijöiden sekä riskien minimointi tunnistamalla uhat ajoissa ja reagoimalla niihin mahdollisimman nopeasti, jolloin voidaan säästää huomattavia määriä rahaa tai aikaa.

3 Avoimen lähdekoodin tietoturvaympäristö Wazuh

3.1 Wazuh yleisesti

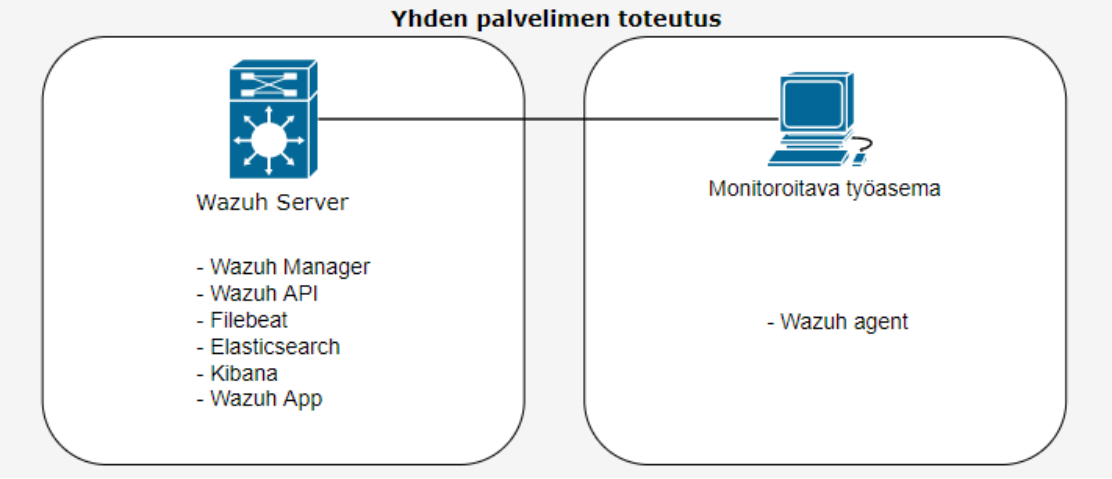
Wazuh on ilmainen sekä avoimeen lähdekoodiin pohjautuva alusta uhkien havaitsemiseen, tietoturvapoikkeamien seuraamiseen ja niihin reagointiin sekä uuden tietoturva lainsäädännön noudattamisen tueksi. Wazuhia voidaan käyttää päätelaitteiden seurantaan, mutta myös pilvipalveluiden ja konttien seurantaan sekä yhdistämään ja analysoimaan ulkoisista lähteistä saatua uhkatietoa. (Welcome to Wazuh n.d.)

Wazuh vaatii kohtuullisen vähän palvelinkapasiteettia verrattuna useimpiin SIEM- eli keskitettyihin tietoturvanhallintaratkaisuihin, mikä tekeekin siitä kustannustehokkaan vaihtoehdon etsittäessä tietoturvaympäristöä. Wazuh voidaan asentaa useimmille Linux-käyttöjärjestelmille, kuten esimerkiksi Ubuntulle, CentOS:lle ja Debianille. Tässä opinnäytetyössä Wazuh asennettiin CentOS-käyttöjärjestelmää käyttävälle virtuaalikoneelle.

Wazuhia käytetään laitteiden tietoturvan kannalta kriittisten tietojen eli lokien keräämiseen, aggregointiin, indeksointiin ja analysointiin. Wazuh auttaa organisaatioita havaitsemaan tunkeutumisia, uhkia ja käyttäytymisen poikkeavuuksia. Näillä ominaisuuksilla Wazuh osaltaan auttaa ehkäisemään tietoturvapoikkeamien huomaamatta jäämistä. (Security Analytics n.d.)

Wazuh-järjestelmän voidaan ajatella koostuvan kahdesta pääelementistä. Managerista sekä Agentista. Manager toimii koko pakkaa ylläpitävänä elementtinä, jolla voidaan nähdä ja ylläpitää ympäristöä uhkien sekä muiden tapahtumien varalta. Wazuh Agent on vain pieni osa koko järjestelmäkokonaisuudesta, eli se toimii itse seurattavalla päätelaitteella keräten lokitapahtumia ja lähettäen ne Wazuh Managerille analysoitavaksi jatkotoimenpiteitä varten. Wazuhissa on Elastic Stack -integraatio tapahtumien tallentamista sekä visualisointia varten.

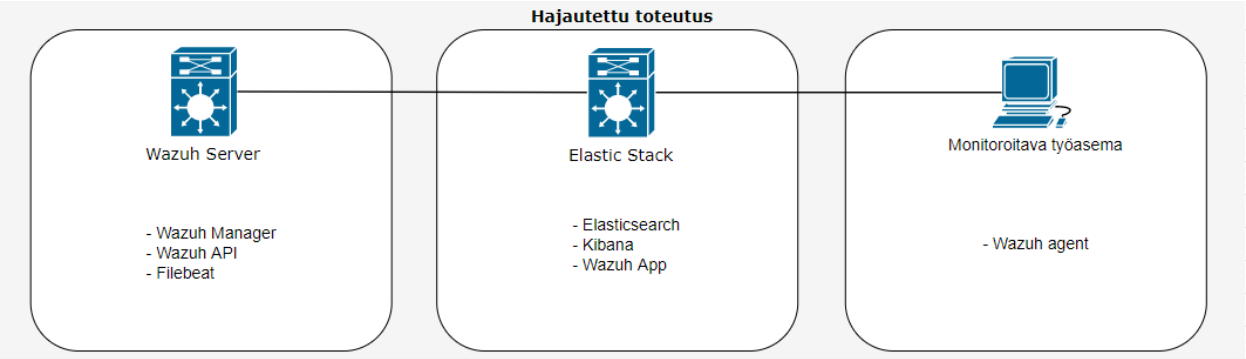
Wazuhin asennusprosessiin Wazuhin oma dokumentaatio suosittaa kahta erityyppistä toteutusvaihtoehtoa: yhden palvelimen toteutus ja hajautettu toteutus. Koska Wazuh perustuu avoimeen lähdekoodiin ja siihen on saatavana kattava dokumentaatio ja ohjeistus, on se helposti lähestyttävä vaihtoehto tietoturvatason lisäämiseksi jo pieniinkin yrityksiin tai ympäristöihin. Sellaisessa tapauksessa, jossa monitoroitavien työasemien määrä on kohtuullisen pieni, voi olla järkevämpää asentaa Wazuh -ympäristö yhdelle keskitetylle palvelimelle kuvion 1 mukaisesti. (Installation guide n.d.)



Kuvio 1. Wazuh yhden palvelimen toteutuksena

Wazuh voi olla myös varteenotettava vaihtoehto keskisuuriin ja jopa suuriinkin yrityksiin ja ympäristöihin. Mikäli monitoroitavia työasemia on paljon, voi olla parempi

jakaa kuormaa eri prosesseista eri palvelimille kuvion 2 mukaisesti. Tällä tavoin voidaan varmistua siitä, että suurenkin datamäärän tullessa sisään on tarvittava palvelinkapasiteetti saatavilla kaikille prosesseille.



Kuvio 2. Wazuh hajautettuna toteutuksena

3.2 Filebeat

Filebeat on kevyt lokitietojen välittämiseen tehty työkalu. Filebeat tarkkailee sekä kerää lokitietoja, jotka se myöhemmin lähettää joko Elasticsearchille tai Logstashille indeksointia varten. Indeksoinnilla tarkoitetaan esimerkiksi raakalokista saatavien samantyyppisten tietojen (samantyyppisten tietokenttien) erittelyä omiin indeksiryhmiinsä, joita tarvitaan Kibanan esittäessä tiedon sisältöä (Indexing n.d).

Wazuh siis käyttää Filebeatia turvalliseen herätteiden sekä muiden tapahtumien lähettämiseen Elasticsearchille (Filebeat overview n.d).

3.3 Elasticsearch

Elasticsearch on työkalu, johon virtaa raakadataa useista eri lähteistä, mukaan lukien lokilähteet, järjestelmätiedot sekä verkkosovellukset. Raakadatatietojen saapuessa

Elasticsearchiin joudutaan raakadata ensin parsimaan, normalisoimaan sekä rikastamaan, ennen kuin se voidaan indeksoida Elasticsearchissa. Käytännössä tämä tarkoittaa sitä, että raakadata muokataan sellaiseen muotoon, että käyttäjät pystyvät suorittamaan kyselyjä lokitietokantaan sekä käyttämään kyselyissä tai hauissa erityyppisiä asiasanoja(indeksejä). Indeksointi helpottaa suurien lokimäärien käsittelyä ja ymmärrystä, jotta esimerkiksi Kibanassa käyttäjät pystyvät tehokkaasti luoda erityyppisiä visualisointeja haluamiensa tietojen esittämiseen tai tutkimiseen. (How does Elasticsearch work? n.d.a.)

Lyhyesti sanottuna Wazuh siis käyttää Elasticsearchia juuri lokien parsimiseen sekä niiden indeksoimiseen, jotta lokeista saadaan kerättyä vain tarvittavat tiedot Kibanan visualisointia varten. (How does Elasticsearch work? n.d.b.)

3.4 Kibana

Kibana on avoimen lähdekoodin analysointi- ja visualisointiympäristö, joka on suunniteltu toimimaan Elasticsearchin kanssa. Kibanaa käytetään yleisesti Elasticsearchin indeksoimien lokien ja tallennettujen tietojen hakuun, tarkasteluun sekä lokien visualisoimiseen niiden luettavuuden helpottamiseksi. Kibanassa on mahdollista tilastoida esimerkiksi piirakkakuvioon, jossa osoitteissa yrityksen asiakkaat eniten vierailevat. (Introduction. n.d.a.)

Kibanassa on selainpohjainen käyttöliittymä, josta erilaisia konfiguroituja näkymiä voidaan tarkastella ja tutkia ilman ylimääräisiä konfiguraatioita. Wazuh käyttääkin Kibanaa juurikin herätteiden visualisoimiseen, jotta lokit olisivat selkolukuisempia. (Introduction. n.d.b.)

3.5 Tunkeilijan havaitsemisjärjestelmä (IDS)

IDS (Intrusion Detection System) eli tunkeilijan havaitsemisjärjestelmää käytetään havaitsemaan tietoturvapoikkeamia kohdesovellusta tai tietokonetta vastaan. IDS järjestelmä eroaa IPS (Intrusion prention system) -järjestelmästä siten, että IDS hälyttää

tietoturvapoikkeamista niihin kuitenkaan reagoimatta. IPS -järjestelmä taas myös poistaa uhat esimerkiksi verkkoliikenteestä.

Wazuh ei suoranaisesti ole IDS -järjestelmä, mutta siitä on löydettävissä samankaltaisia ominaisuuksia. Wazuh -agentit tarkistavat järjestelmiä etsien haittaohjelmia, rootkittejä, sekä muita epäilyttäviä tietoturvapoikkeavuuksia. Wazuh -agentit voivat havaita piilotetut tiedostot, peitetyt prosessit tai ulkopuoliset verkkokuuntelijat sekä työaseman ollessa hyökkääjän hallinnassa, sen ”takaisin soiton” eli CnC (Comman and Control) -liikenteen hyökkääjälle. CnC liikenteellä hyökkääjä voi mahdollisesti etäältä komentaa kohdekonetta haluamallaan tavalla. (Intrusion Detection. n.d.a.)

Agenttiominaisuuksien lisäksi palvelinkomponentti käyttää allekirjoituspohjaista tunkeutumisen estämistä käyttämällä säännöllisen lausekkeen moottoria kerättyjen lokitietojen analysoimiseksi sekä mahdollisten uhka indikaattoreiden etsimiseen. (Intrusion Detection. n.d.b.)

3.6 Lokien analysointi

Wazuh -agentit keräävät käyttöjärjestelmän ja sovellusten lokit ja toimittavat ne turvallisesti edelleen keskitetylle Wazuh manager palvelimelle sääntöpohjaista analyysiä ja tallennusta varten. Wazuhiin on mahdollista määrittää säännöt, joiden mukaan lokia kerätään. Lokit ovat äärimmäisen tärkeitä tietoturvapoikkeamien selvittämisessä, sillä kattavasti kerätystä lokista voidaan nähdä suoraan tietoa sovellus- tai järjestelmävirheistä, virheellisistä määrityksistä, haitallisten toimien yrityksistä tai onnistumisista, käytäntöjen rikkomisista ja monista muista turvallisuus- ja operatiivisista ongelmista, sekä näiden ajankohdista. (Log Data Analysis n.d.)

3.7 Tiedostojen eheyden seuranta

Wazuh tarkkailee tiedostojärjestelmää tunnistaen muutokset sisällössä, käyttöoikeuksissa, työasemassa ja tiedostojen määritteissä, jotka voivat olla muuttuneet esimerkiksi tiedoston saastuessa. Lisäksi se identifioi alkuperäiset käyttäjät ja sovellukset, joita käytetään tiedostojen luomiseen tai muokkaamiseen. Tiedostojen eheyden

valvontaominaisuuksia voidaan käyttää yhdessä uhkatiedon kanssa uhkien tai haitallisten tiedostojen tunnistamiseen. (File Integrity Monitoring n.d.)

3.8 Haavoittuvuuksien havaitseminen

Wazuh -agentit hakevat uhkatietokannasta tiedot uusista haavoittuvuuksista sekä uhista ja lähettävät nämä tiedot palvelimelle, missä ne korreloivat jatkuvasti päivitettävien CVE (Common Vulneramissions and Exposure) -tietokantojen kanssa tunnettujen haavoittuvien ohjelmistojen tunnistamiseksi. Automaattinen haavoittuvuuksien havainnointi auttaa löytämään ympäristölle kriittiset haavoittuvuudet ajoissa ja nopeuttaa ryhtymistä korjaaviin toimenpiteisiin ennen kuin hyökkääjät kerkeävät hyödyntää haavoittuvuutta. Esimerkiksi hyökkäys yrityksen laitteita vasten voisi pahimmassa tapauksessa aiheuttaen luottamuksellisia tietoja vuotamisen ulkopuolisille tai tietojen katoamisen. (Vulnerability Detection n.d.)

3.9 Kokoonpanon arviointi

Wazuh tarkkailee järjestelmän ja sovelluksen kokoonpanoasetuksia varmistaakseen, että ne ovat turvallisuuskäytäntöjen ja standardien mukaisia. Agentit suorittavat säännöllisiä skannauksia havaitakseen sovelluksia, joiden tiedetään olevan haavoittuvia, korjaamattomia tai muutoin epäilyttäviä. Lisäksi kokoonpanotarkastuksia voidaan muuttaa tarvittaessa siten, että ne vastaavat kunkin organisaation tietoturvakäytäntöjä. Wazuh myös generoi hälytyksiä, jotka sisältävät suosituksia esimerkiksi paremmasta kokoonpanosta tai haavoittuvan ohjelmiston päivittämisestä. (Configuration Assessment n.d.)

3.10 Tietoturvapoikkeamaan vastaaminen (Incident Response)

Incident responsella tarkoitetaan organisoitua tapaa vastata hallitusti tietoturvapoikkeamiin, kuten esimerkiksi kyberhyökkäykseen. Tavoitteena on käsitellä tietoturvapoikkeama siten, että minimoidaan mahdollisten haittojen aiheuttamat kustannukset

niin ajallisesti kuin laitteiston korjauskustannuksissa. Monilla organisaatiolla on nykypäivänä oma tietoturvapoikkeamien hoitamisesta vastaava ryhmä eli SIRT-ryhmä (Security incident response team), joka vastaa tietoturvapoikkeamiin reagoimisesta sekä toimenpiteistä tietoturvapoikkeamien vastaamiseen laaditun suunnitelman eli IRP:in (Incident response plan) mukaan. (Rouse 2019b.)

Wazuh tarjoaa valmiita toimenpide-ehdotuksia tietoturvapoikkeaman tapahtuessa. Esimerkiksi ulkoapäin tulevassa skannauksessa Wazuh voi ehdottaa pudottamaan sisäänpäin tulevan liikenteen haitalliseksi toteamastaan liikenteestä. (Incident Response n.d.)

3.11 Tietoturvasäännöstön noudattaminen

Wazuh tarjoaa joitain tarvittavia tietoturvavalvontatoimenpiteitä alan standardien ja -määräysten noudattamiseksi. Nämä ominaisuudet auttavat organisaatioita täyttämään tekniset tietoturvavaatimukset. (Regular Compliance n.d.a.)

Maksunkäsittely-yritykset ja finanssilaitokset käyttävät Wazuhia laajalti PCI DSS (Payment Card Industry Data Security Standard) -vaatimusten täyttämiseen. Wazuhin verkkokäyttöliittymä tarjoaa raportteja ja näkymiä eli dashboardeja, jotka voivat auttaa esimerkiksi GPG13- tai GDPR- säädöksiä noudatettaessa. (Regular Compliance n.d.b.)

3.12 Pilviturvallisuuden seuranta

Wazuh auttaa tarkkailemaan pilvi-infrastruktuuria esimerkiksi verkkosovellusten välisten rajapintojen tasolla (API -tasolla) käyttämällä integrointimoduuleja, jotka pystyvät hakemaan turvallisuustietoja tunnetuilta pilvipalvelujen tarjoajilta. Tällaisia pilvipalvelujen tarjoajia ovat muun muassa Amazon AWS, Azure tai Google Cloud. Lisäksi Wazuh tarjoaa säännöt pilviympäristön kokoonpanon arvioimiseksi ja havaitsee mahdolliset heikkoudet. Lyhyesti sanottuna Wazuhin kevyitä agentteja käytetään siis yleisesti pilviympäristöjen seuraamiseen (Cloud Security Monitoring n.d).

3.13 Konttien turvallisuuden seuranta

Wazuh tarjoaa tietoturvanäkyvyyden myös Docker- kontteihin. Kontit toimivat yleensä yhteisen alustan päällä. Kontilla tarkoitetaan keinoa paketoida ja ajaa sovelluksia samalla eristäen sovellukset toisistaan (Kotilainen 2017). Wazuh seuraa konttien toimintaa, havaitsee uhkia, haavoittuvuuksia ja poikkeavuuksia konteissa. Wazuh-agentti on integroitu alkuperäiseen osaan Docker-moottoria, jonka avulla käyttäjät voivat seurata esimerkiksi levykuvia, verkkoasetuksia ja käynnissä olevia kontteja. (Containers Security n.d.a.)

Wazuh kerää ja analysoi jatkuvasti yksityiskohtaisia tietoja konttien toiminnasta. Wazuh varoittaa esimerkiksi priviledged modessa eli suurimman prioriteetin toimintatilassa toimivista konteista, haavoittuvista sovelluksista, kontissa käynnissä olevasta komentokehotteesta, sekä muista mahdollisista uhista. (Containers Security n.d.b.)

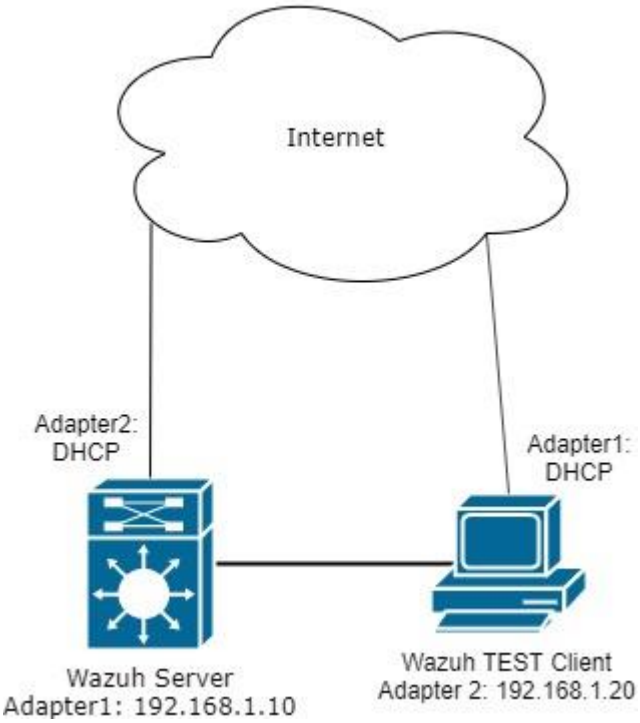
4 Tekninen toteutus

4.1 Wazuh manager -asennus

4.1.1 Yleistä

Tässä luvussa käydään läpi Wazuh managerin asennuksen vaiheet. Opinnäytetyössä käytettiin valmiiksi asennettua CentOS Linux -versiota, johon ei ollut tehty mitään muita asennuksia. Asennukset tehtiin root- eli järjestelmäkäyttäjänä, jotta vältyttiin mahdollisilta käyttäjäoikeusongelmilta.

Opinnäytetyön ympäristöä on käytetty vain Wazuhin toiminnan testaamiseen ja siitä tietojen keräämiseen. Ympäristö on verrattain yksinkertainen kuten topologiakuvio-sta 3 voidaan nähdä.



Kuvio 3. Ympäristön topologia

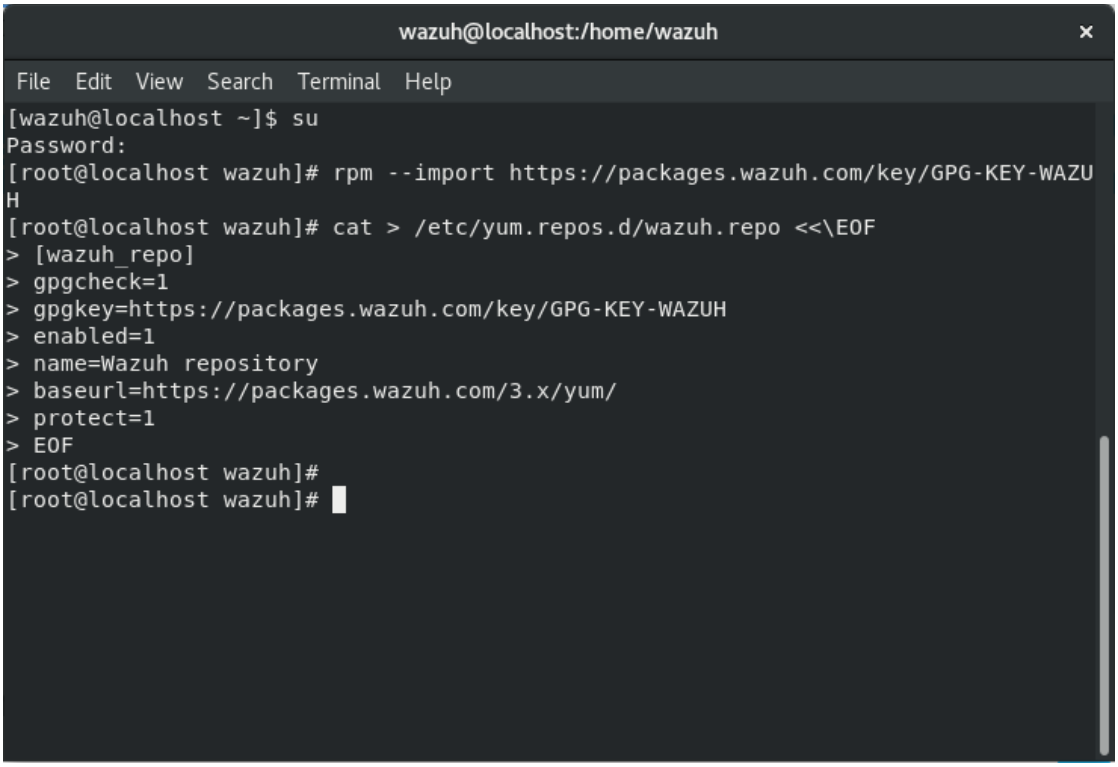
Kummassakin Virtual box -virtualisointialustaan asennetussa virtuaalikoneessa on kaksi verkkokorttia, joista toinen on osoitettu sisäverkkoon ja toinen osoittaa internetiin. Tämä siksi, että internetistä saatiin ladattua asennuspaketit Wazuhin asennusta varten. Toiset verkkokortit osoittavat sisäverkkoon, jotta kyseiset työasemat pystyvät keskustelemaan ja välittämään lokitietoja.

Wazuh Serverissä olevista verkkokorteista toiseen on asetettu edellä mainittu kiinteä IP-osoite 192.168.1.10 ja aliverkon maski on 255.255.255.0. Toinen verkkokorteista osoittaa internetiin käyttäen DHCP:tä, jolloin laite saa automaattisesti verkon IP -osoitteen.

Wazuh test Clientissä olevista verkkokorteista toisen IP-osoite on 192.168.1.20 aliverkon maskilla 255.255.255.0. Toinen verkkokorteista osoittaa internettiin samoin kuin Wazuh serverinkin toinen verkkokortti eli käyttäen DHCP:tä.

4.1.2 Hakemistojen lataus

Wazuh-asennus lähti liikkeelle asennukselle välttämättömien hakemistojen lataamisesta osoitteesta: https://packages.wazuh.com/key/GPG-KEY-WAZUH. Komento hakemistojen lataamiseen oli: ”rpm –import https://packages.wazuh.com/key/GPG-KEY-WAZUH” kuvion 4 mukaisesti.



Kuvio 4. Wazuh hakemistojen lataaminen

Kuten kuviosta 4 nähdään, hakemistojen lataamiskomennon syöttämisen jälkeen CentOS:in konsoli ajoi automaattisesti komennon: ”cat > /etc/yum.repos.d/wazuh.repo <<\EOF”. Kyseisellä komennolla lisättiin tiedosto ”wazuh.repo”.

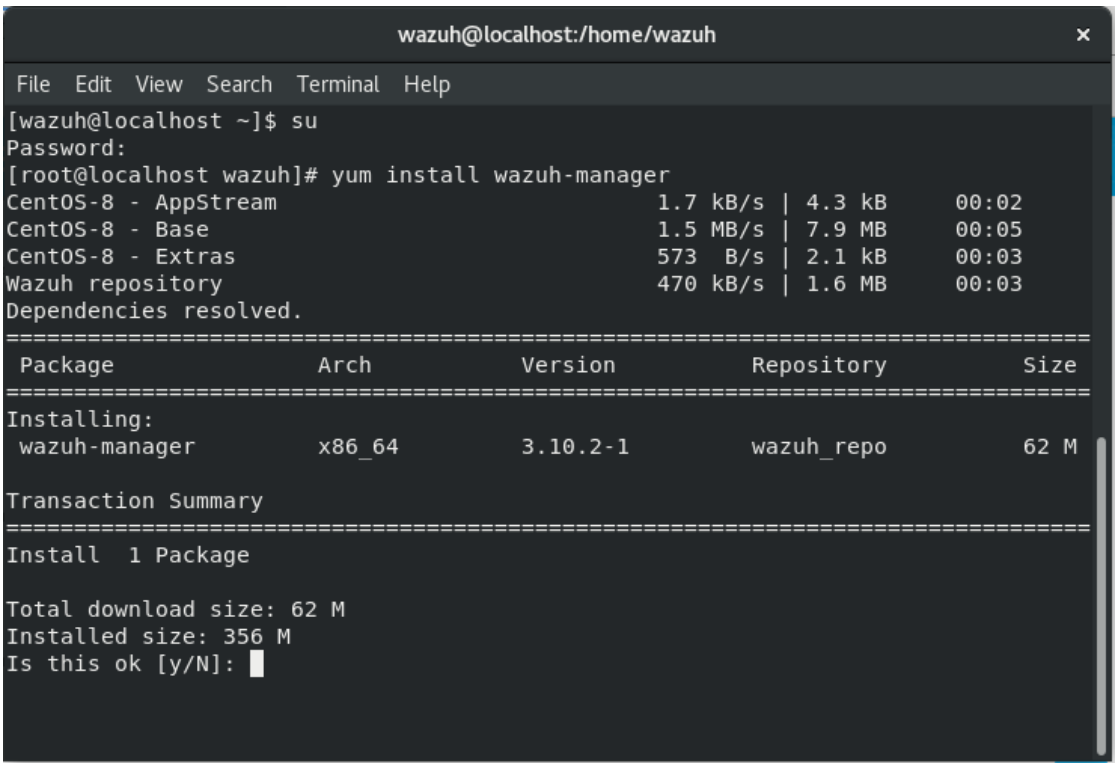
Kun hakemistojen lataus oli valmis, käytiin varmistamassa, että hakemistot tuotiin oikein oikeaan kansioon, kuten kuviosta 5 voidaan nähdä. Tämä tehtiin siksi, että ha-

kemistojen tuonnissa oli aluksi ongelma ja hakemistot eivät todellisuudessa siirtyneetkään oikeaan kansioon, vaikka komento saatiinkin suoritettua onnistuneesti. Verkkokorttien uudelleenkäynnistämisen jälkeen hakemistot saatiin onnistuneesti ladattua ja niiden tuominen kansioon: yum.repos.d onnistui.

Kuvio 5. Hakemistojen tuonnin tarkistaminen

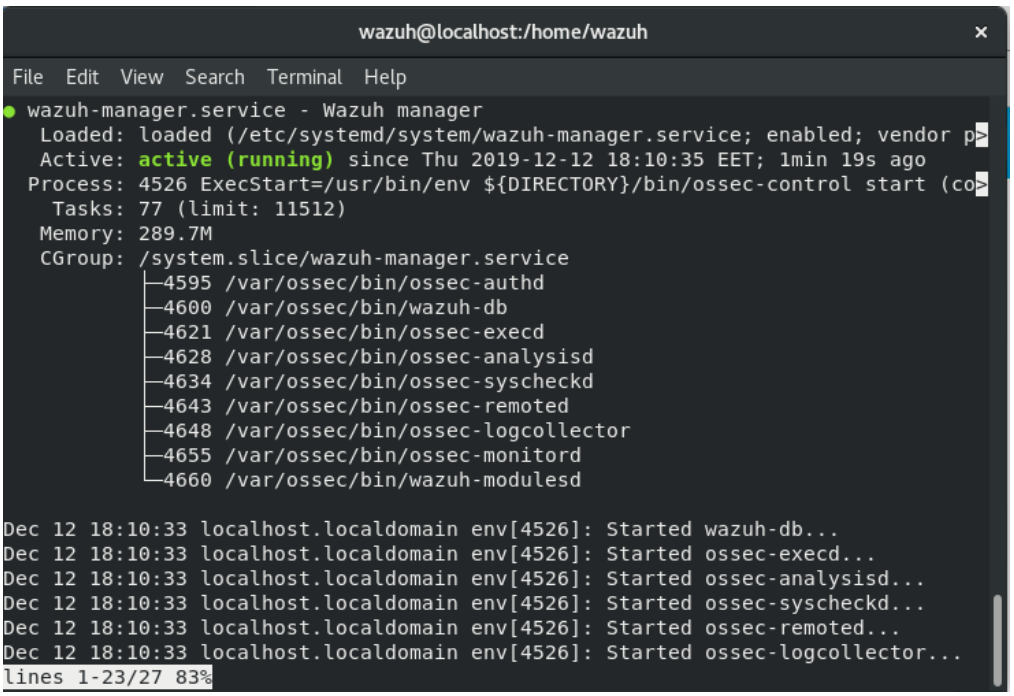
4.1.3 Wazuh Manager -asennus

Hakemistojen onnistuneen tuonnin jälkeen voitiin siirtyä itse Wazuh-manager 3.10.2-1 -version asentamiseen. Wazuh managerin asentaminen tapahtui komennolla: ”yum install wazuh-manager”. Koska todellinen asennuspaketin koko on kohtuullisen suuri, kannattaa Wazuh server -koneelle varata noin 20 gigan suuruinen kovalevy, sillä itse CentOS-käyttöjärjestelmä itsessään vie levyltä jo melko suuren osan. Wazuh manager -asennuksen lopullinen koko oli 356 MB, kuten kuviosta 6 voidaan nähdä.

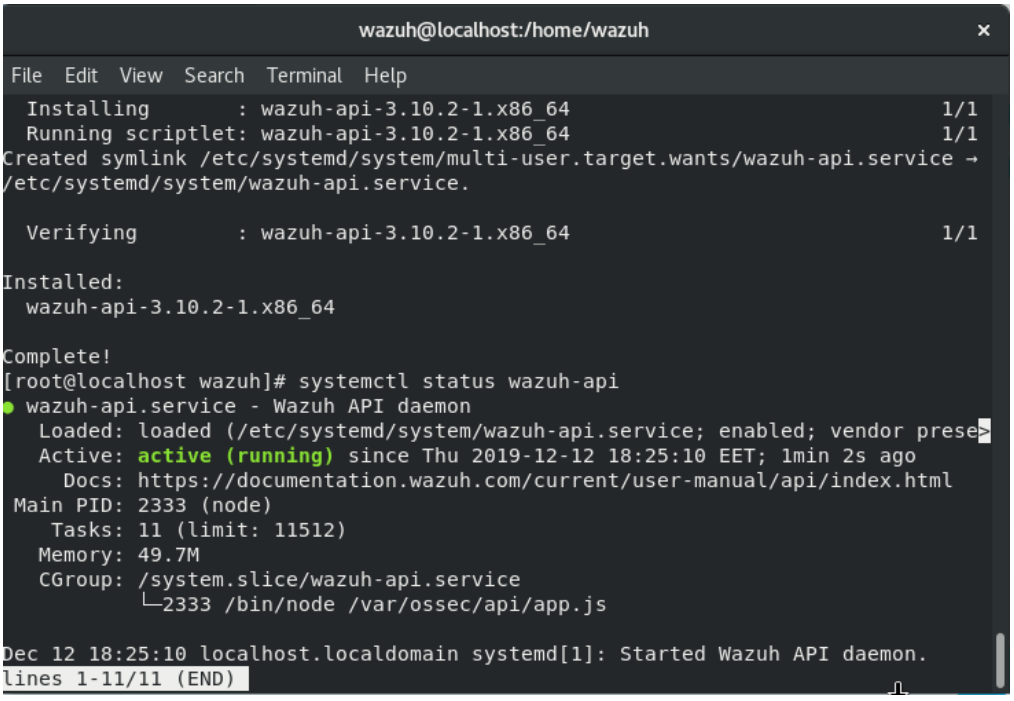


Kuvio 6. Wazuh manager -asennus

Asennuksen onnistuttua testattiin, että Wazuh managerin prosessit käynnistyivät onnistuneesti. Kuvioista 7 ja 8 nähdään, että systemd sekä SysV Init -prosessit käynnistyivät onnistuneesti. Kyseisten prosessien tarkistaminen tapahtui komennoin: ” systemctl status wazuh-manager” sekä ”service wazuh-manager status”.



Kuvio 7. Wazuh Manager Systemd -prosessin toiminnan todennus

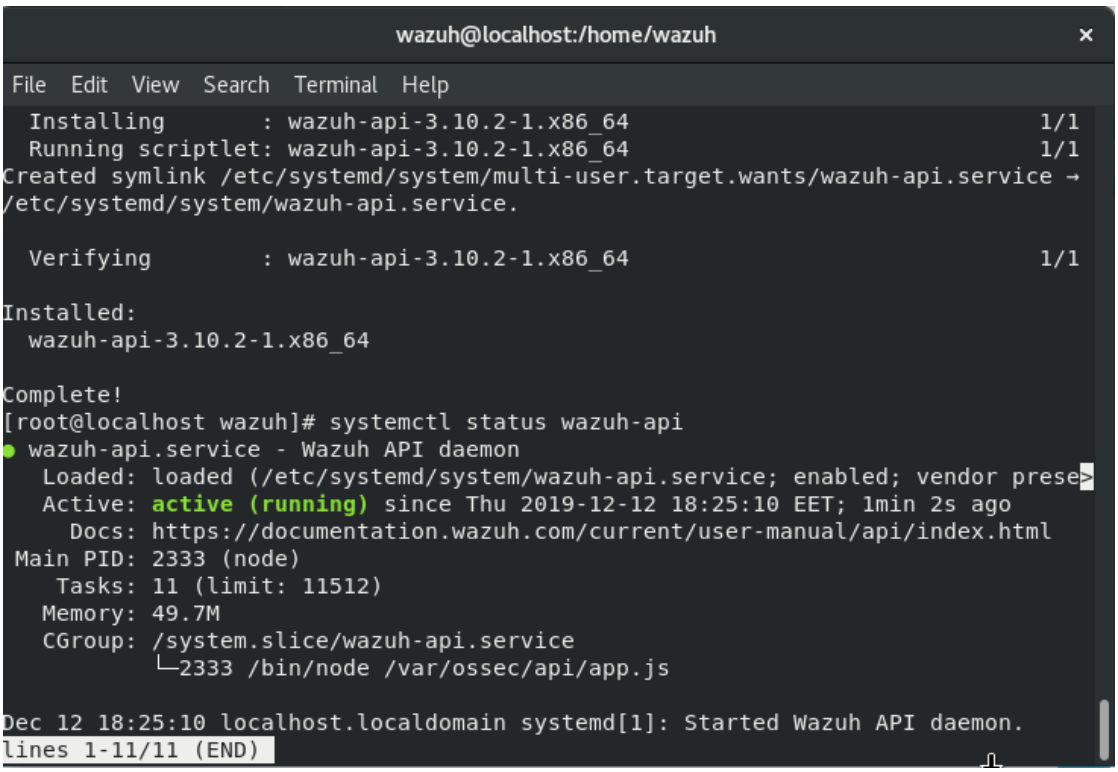


Kuvio 8. Wazuh Manager SysV Init -prosessin toiminnan todennus

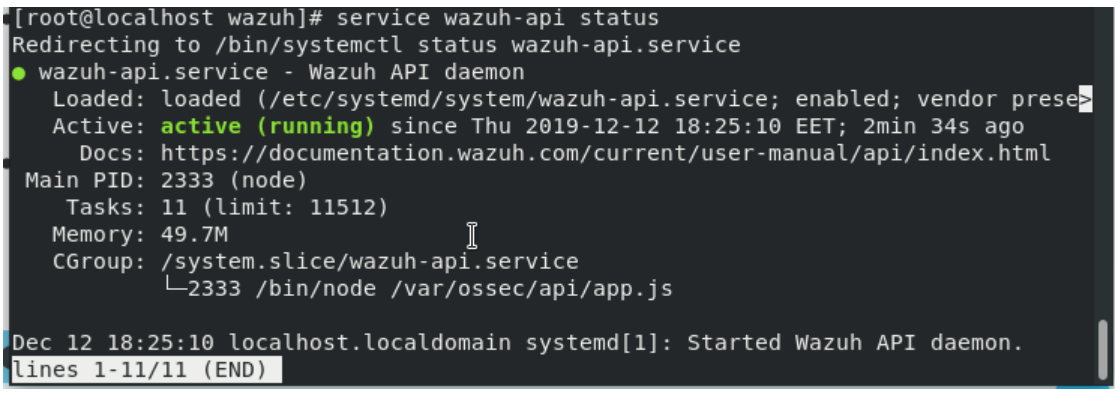
4.1.4 Wazuh API -asennus

Kun Wazuh managerin prosessit saatiin käynnistymään onnistuneesti, voitiin seuraavana aloittaa Wazuh API:n asennus. Wazuh API vaatii toimiakseen avoimen lähdekoodin alustariippumattoman JavaScript Run-Time -ympäristön, NodeJS:n, joka on versioltaan 4.6.1 tai sitä uudempi. Koska NodeJS ei ollut asennettuna, voitiin asennuspaketti ladata komennolla: ”curl --silent --location ahttps://rpm.nodesource.com/setup_8.x | bash –”. Kun asennuspaketti oli onnistuneesti ladattu, voitiin NodeJS asentaa komennolla: ”yum install nodejs”. (Installing the Wazuh API n.d.)

NodeJS:n asennuksen onnistuttua voitiin asentaa itse Wazuh API. Wazuh API:n asennus tapahtui komennolla: ” yum install wazuh-api”. Asennuksen onnistuttua todennettiin vielä Wazuh API:n prosessien toimivuus lähes samalla tavoin kuin Wazuh Managerin asennuksessa. Komennot prosessien toiminnan tarkistamiseen olivat: ”systemctl status wazuh-api” sekä ”service wazuh-api status”. Kuten kuvioista 9 ja 10 voidaan nähdä, prosessit käynnistyivät onnistuneesti.



Kuvio 9. Wazuh API Systemd -prosessin toiminnan todennus



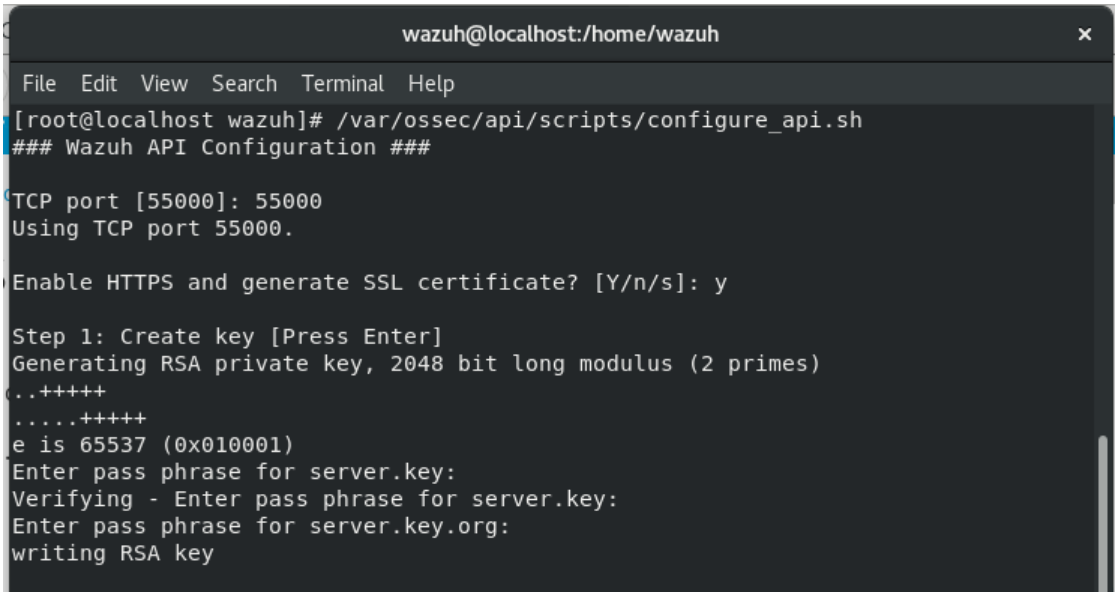
Kuvio 10. Wazuh API SysV Init -prosessin toiminnan todennus

4.1.5 Wazuh API turvallisuustason nostaminen

Tämä vaihe ei Wazuhin toiminnan kannalta ollut välttämättä tarvittava, mutta Wazuhissa oli olemassa riski sille, että tapahtuu tarkoituksetta tapahtuvia päivityksiä, jotka voivat pahimmillaan aiheuttaa järjestelmän toiminnan lakkautumisen tai osittain toimimisen. Tämä voitiin kuitenkin välttää poistamalla käytöstä Wazuh -hakemisto, josta päivityksiä haetaan. Tämä tapahtui komennolla: ”sed -i "s/^enabled=1/enabled=0/" /etc/yum.repos.d/wazuh.repo”. (Securing the Wazuh API n.d.a.)

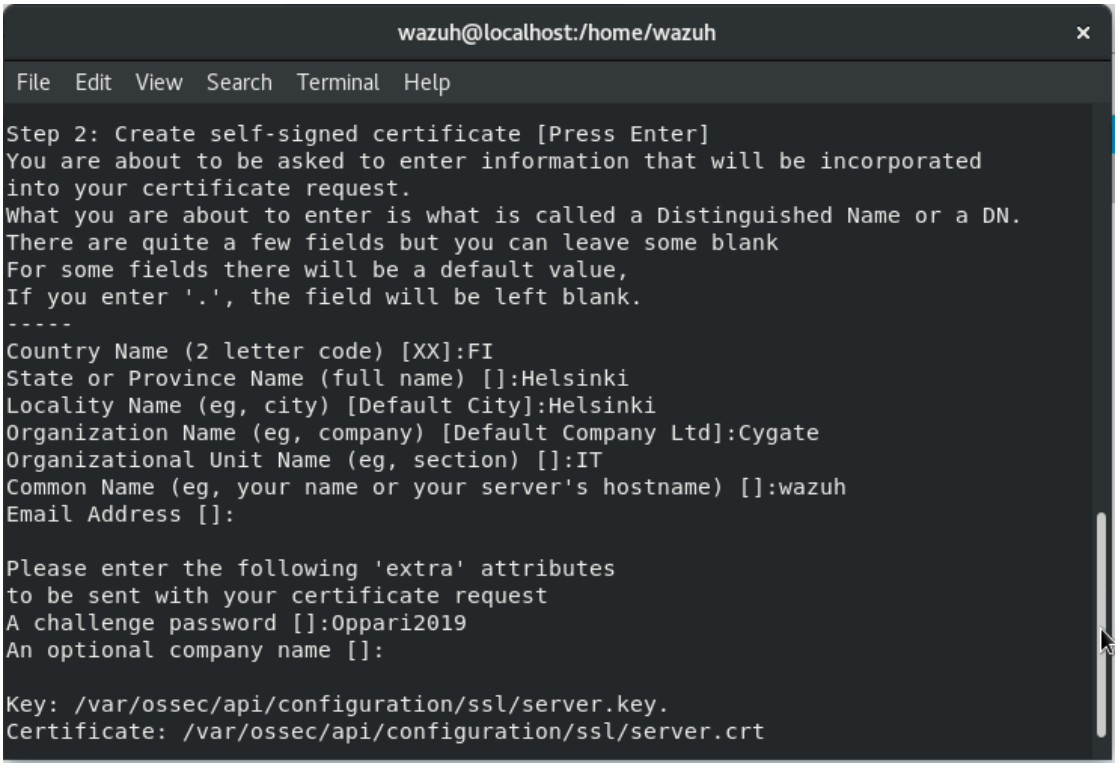
Oletuksena yhteys Wazuh Kibanan sekä Wazuh API:n välillä ei ollut salattu. Tietoturvatason nostamiseksi otettiin käyttöön HTTPS -protokolla HTTP-protokollan sijaan, jotta liikenne kulki salattuna. HTTPS-protokollan käyttöönottaminen vaatii oman sertifikaatin luomista. Tämä voitiin generoida automaattisesti käyttäen olemassa olevaa skriptiä: ”/var/ossec/api/scripts/configure_api.sh” (Securing the Wazuh API n.d.b.)

Kuviossa 11 nähdään configure_api.sh -skriptin ajo, jossa automaattisesti luodaan sertifikaatti HTTPS -liikennettä varten.



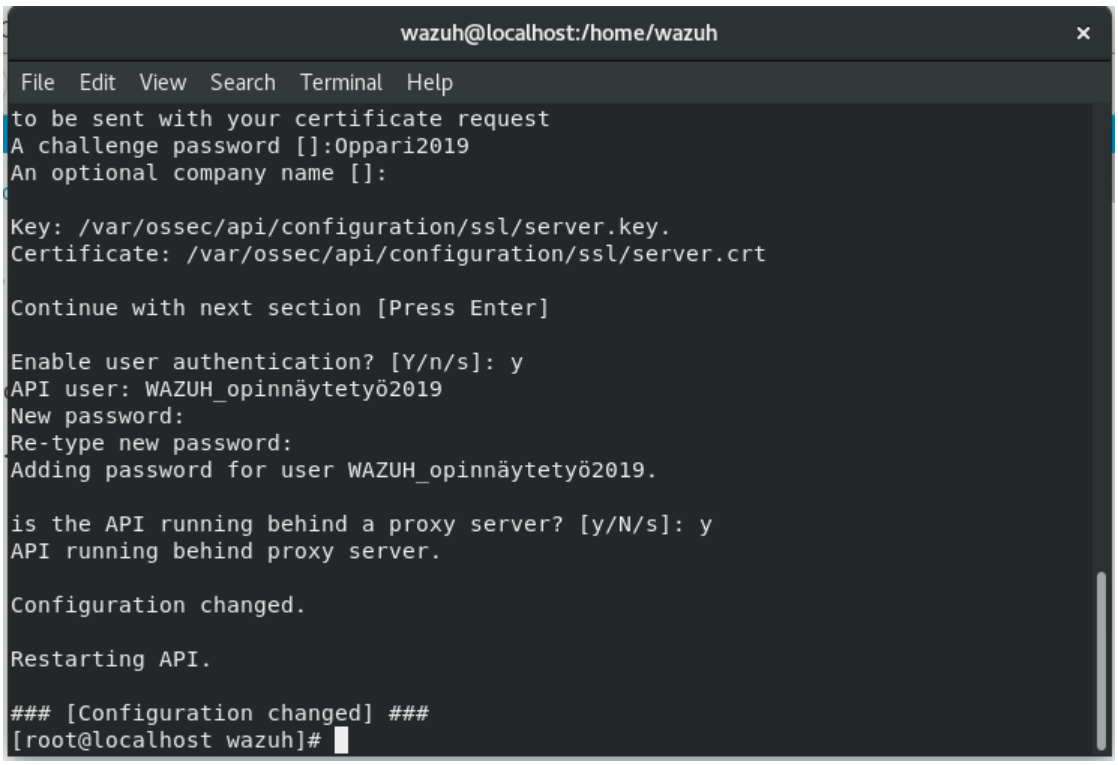
Kuvio 11. HTTPS sertifikaatin luonti configure_api.sh scriptillä

Scripti kysyy esimerkiksi maatiedot, kaupungintiedot ja salasanan, kuten kuviosta 12 voidaan nähdä.



Kuvio 12. Tietojen täydennys luotavaan sertifikaattiin

Tietojen lisäyksen jälkeen otettiin käyttöön käyttäjän autentikointi. Käyttäjäksi valittiin koneen paikallinen käyttäjä: ”WAZUH_opinnäytetyö2019” ja salasanaksi: ”Oppari2019”, kuten kuviosta 13 voidaan nähdä.



Kuvio 13. Sertifikaattiin käyttäjätietojen lisäys

Jotta muutokset saatiin käyttöön, täytyi wazuh-api sekä wazuh-manager -prosessit käynnistää uudelleen.

Jotta vältyttiin odottamattomilta päivityksiltä, poistettiin wazuh.repo käytöstä. Tämä tapahtui komennolla: ” sed -i "s/^enabled=1/enabled=0/" /etc/yum.repos.d/wazuh.repo”

4.1.6 Filebeatin asennus

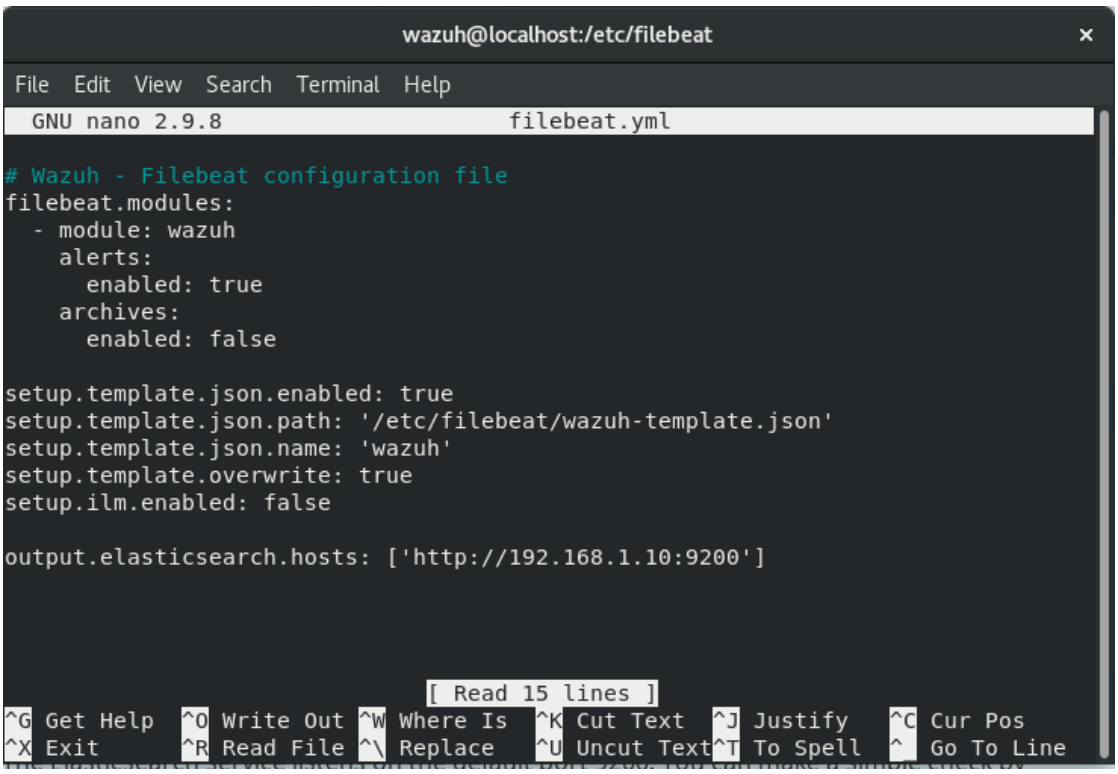
Filebeatin asennus aloitettiin Elasticsearch -hakemistojen lisäämisellä. Tämä tapahtui komennoilla: ”rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch”, ”cat > /etc/yum.repos.d/elastic.repo << EOF”.

Kun hakemistot oli onnistuneesti lisätty, voitiin siirtyä itse Filebeatin asennukseen. Filebeat asennettiin komennolla: ”yum install filebeat-7.4.2”.

Filebeat-asennuksen onnistuttua ladattiin Filebeat konfiguraatio -tiedosto, josta löytyi valmiiksi tehtyjä herätteitä Elasticsearchia varten. Filebeat konfiguraatio -tiedosto ladattiin Wazuhin hakemistosta komennoin: ”curl -so /etc/filebeat/filebeat.yml https://raw.githubusercontent.com/wazuh/wazuh/v3.10.2/extensions/filebeat/7.x/filebeat.yml”, ”chmod go+r /etc/filebeat/filebeat.yml”. Filebeat konfiguraatio -tiedoston lisäksi täytyi ladata myös herätekirjasto (alerts template) Elasticsearchia varten. Tähän komennot olivat: ”curl -so /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/v3.10.2/extensions/elasticsearch/7.x/wazuh-template.json”, “chmod go+r /etc/filebeat/wazuh-template.json”.

Jotta Filebeat saatiin toimimaan Wazuhin dokumentaation mukaan, ladattiin Filebeatiin Wazuh -moduuli komennolla: ” curl -s https://packages.wazuh.com/3.x/filebeat/wazuh-filebeat-0.1.tar.gz | sudo tar -xvz -C /usr/share/filebeat/module”.

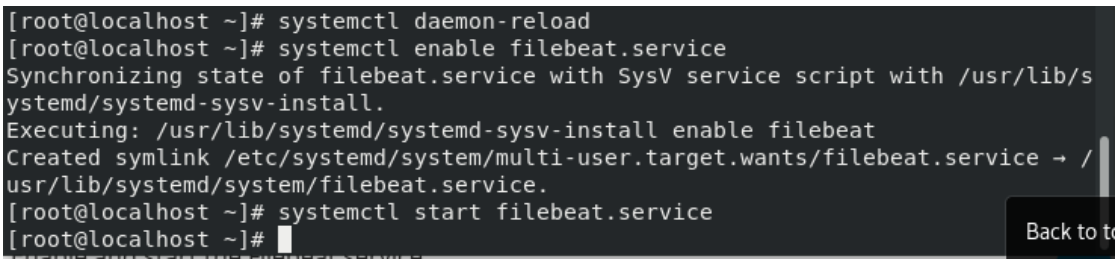
Tiedosto “/etc/filebeat/filebeat.yml” oli muutoin valmiiksi konfiguroitu, mutta Elasticsearch -serverin IP-osoite täytyi muuttaa vastaamaan tässä opinnäytetyössä käytettyä serverin IP-osoitetta. Kuviosta 14 nähdään, että tiedoston riviä: ” output.elasticsearch.hosts: ['http://YOUR_ELASTIC_SERVER_IP:9200']” muutettiin siten, että riville laitettiin tässä opinnäytetyössä käytetyn serverin osoite 192.168.1.10.



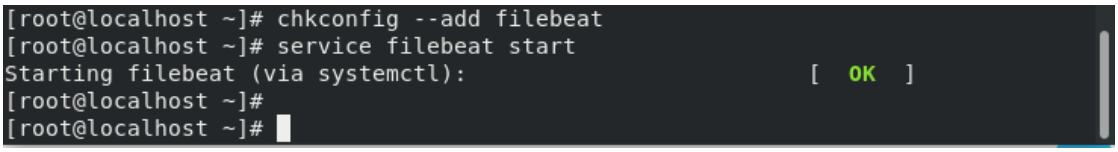
Kuvio 14. Filebeat.yml -tiedostoon serverin osoitteen lisäys.

Tiedostoon IP-osoitteen lisäyksen jälkeen Filebeatin asennus oli käytännössä valmis ja Filebeat voitiin ottaa käyttöön ja käynnistää. Tämä tapahtui Systemdille komennoin: ”systemctl daemon-reload”, ”systemctl enable filebeat.service”, ”systemctl start filebeat.service” ja SysV Initiä varten komennoin: ”chkconfig --add filebeat”, ”service filebeat start”.

Kuvioista 15 ja 16 nähdään, että virheilmoitusta ei tullut ja prosessit käynnistyivät ongelmitta.



Kuvio 15. Filebeat Systemd -prosessien käynnistys



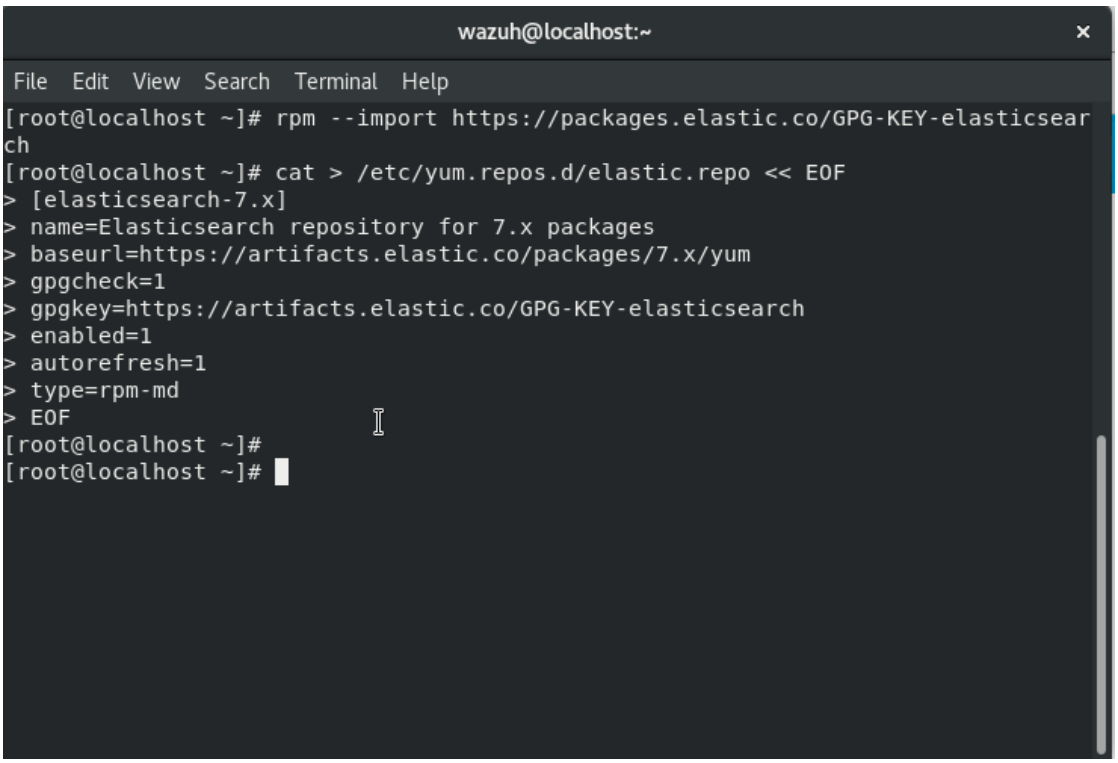
Kuvio 16. Filebeat SysV Init -prosessien käynnistys

Jotta myös Elasticin ei-toivotuilta päivityksiltä vältyttiin, otettiin myös elastic.repo hakemisto pois käytöstä komennolla: ” sed -i "s/^enabled=1/enabled=0/" /etc/yum.repos.d/elastic.repo”.

4.1.7 Elasticsearchin asennus

Elasticsearch on indeksointityökalu, jota Wazuh käyttää Filebeatilta saatavien lokien indeksoimiseen.

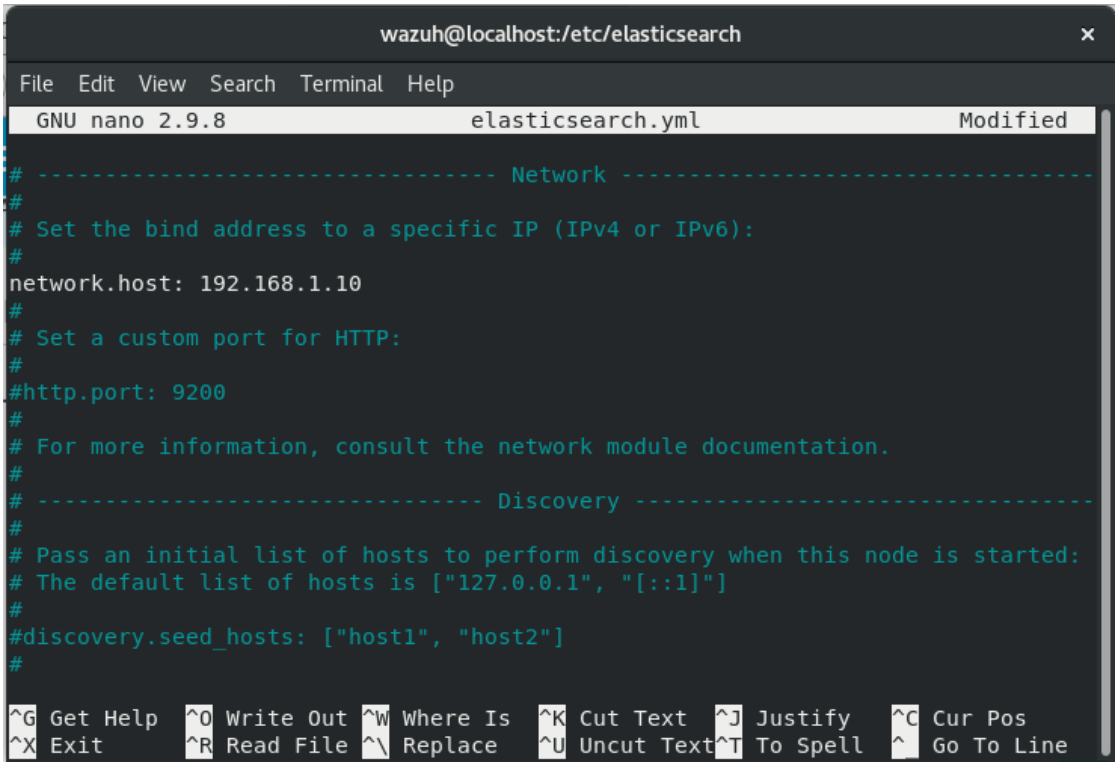
Elasticsearchin asennus aloitettiin lataamalla Elastic -hakemisto ja sen GPG -avain komennoilla: ”rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch”, ”cat > /etc/yum.repos.d/elastic.repo << EOF”. Kuviosta 17 nähdään, että hakemistot saatiin onnistuneesti lisättyä.



Kuvio 17. Elasticsearch -hakemistojen lisäys

Hakemistojen lisäyksen jälkeen voitiin aloittaa itse Elasticsearchin asennus. Tämä Tapahtui komennolla: ”yum install elasticsearch-7.4.2”.

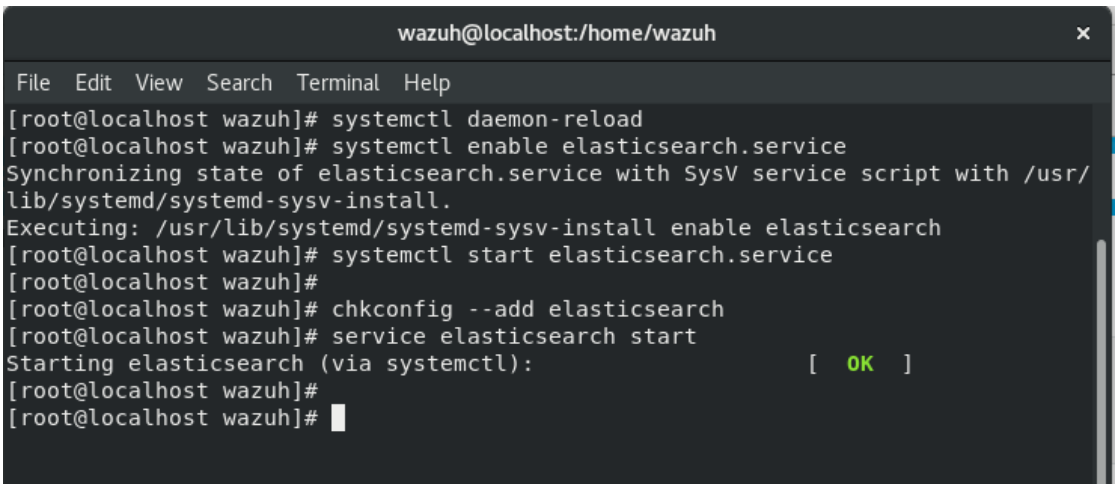
Elasticsearch kuunteli oletuksena vain loopback interfacea eli itseään, joten se muutettiin kuuntelemaan omaa Elasticsearch-serverin osoitetta. Muutokset kohdistuivat riveille: ” network.host: <elasticsearch_ip>” ja ”node.name: <node_name>, ”cluster.initial_master_nodes: ["<node_name>"]”. Kuviosta 18 nähdään IP-osoitteen muutos omaa serverin IP-osoitetta vastaavaksi.



Kuvio 18. elasticsearch.yml -tiedoston muuttaminen ympäristöä vastaavaksi

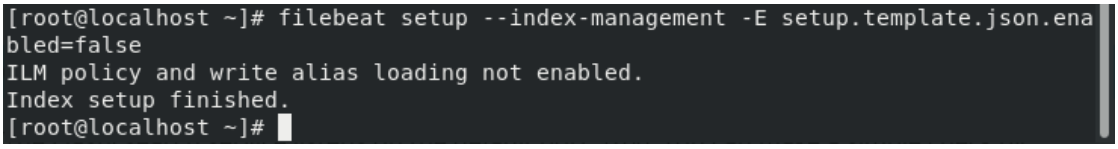
Muutokset tehtiin tiedostoon: ” /etc/elasticsearch/elasticsearch.yml”.

Tiedoston muutoksien jälkeen elasticsearchin asennus oli valmis ja elasticin prosessit voitiin käynnistää kuvion 19 mukaisesti komennoin: ”systemctl daemon-reload”, ”systemctl enable elasticsearch.service”, ”systemctl start elasticsearch.service” sekä ”chkconfig --add elasticsearch”, ”service elasticsearch start”.



Kuvio 19. Elasticsearch -prosessien käyttöönotto ja käynnistys

Kun Elasticsearch saatiin käyntiin onnistuneesti, ladattiin Filebeat -pohja (template) komennolla: ” filebeat setup --index-management -E setup.template.json.enabled=false”, sekä asennettiin kuvion 20 mukaisesti.



Kuvio 20. Filebeat -pohja asennus

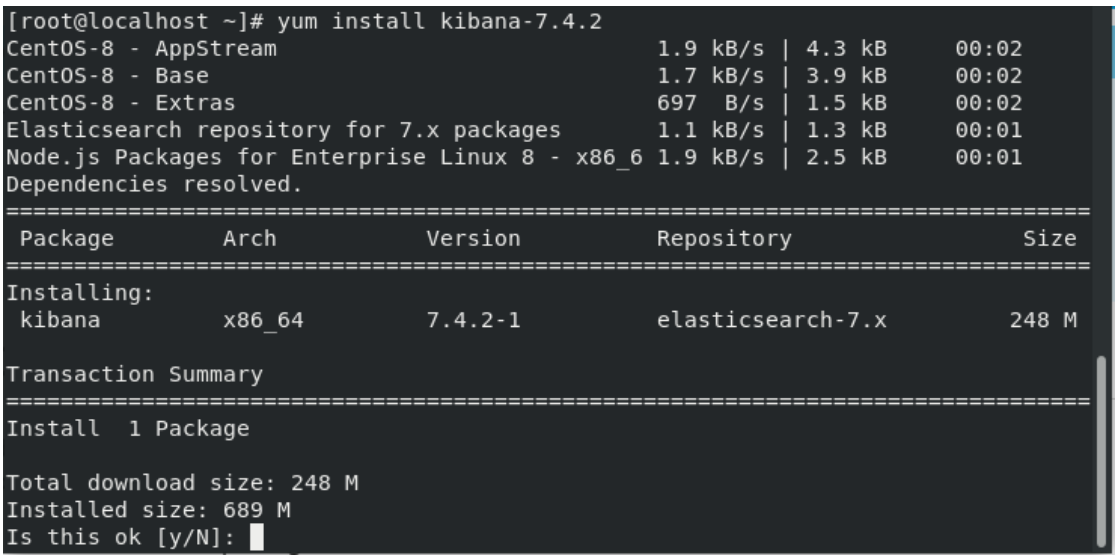
Lähtökohtaisesti Elasticsearch -prosessi kuunteli oletuksena porttia: 9200. Tämä kuitenkin todennettiin vielä komennolla: ”curl http://192.168.1.10:9200”. Komennon ajattua nähtiin, että Elasticsearch pystyi kuuntelemaan porttia 9200, ja se palautti kuvion 21 vastauksen.

Kuvio 21. Tarkistetaan, että Elasticsearch kuuntelee porttia 9200

4.1.8 Kibana -asennus

Kibana on visualisointityökalu, jonka web-käyttöliittymässä on visualisoituna ja tallennettuna Elasticsearchin indeksoimia lokeja analysointia varten.

Kibanan asennus tapahtui komennolla: ” yum install kibana-7.4.2”, kuvion 22 mukaisesti.



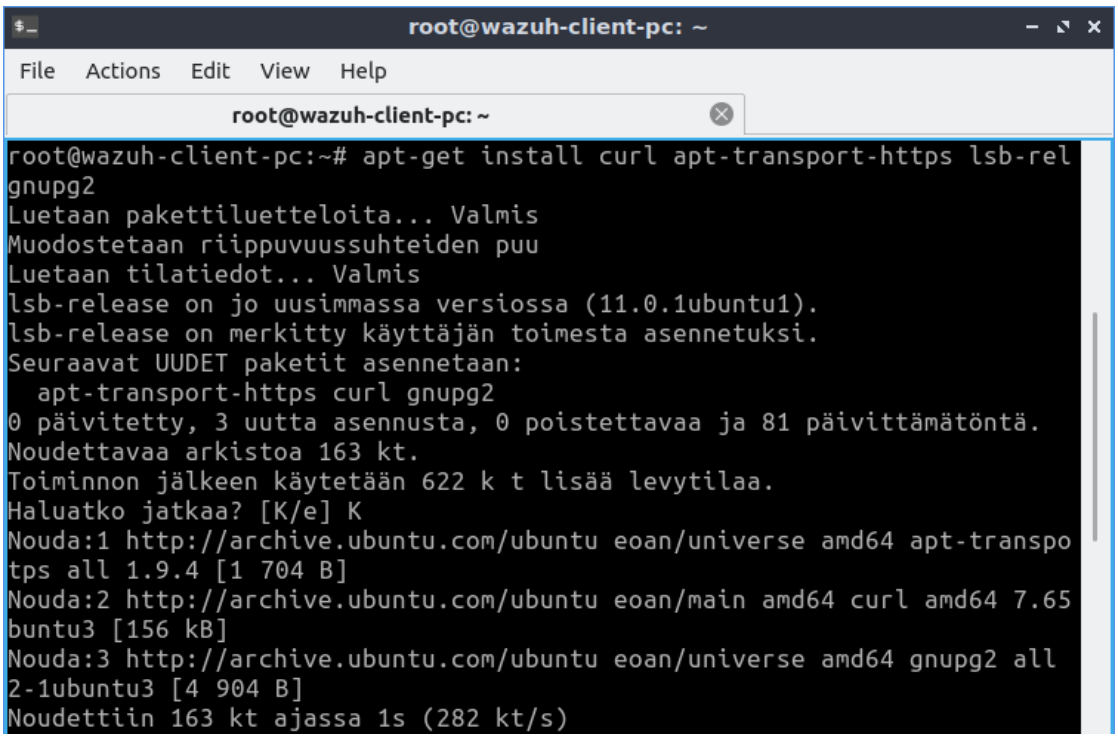
Kuvio 22. Kibana -asennus

Kibanan asennus oli viimeinen vaihe Wazuh managerin asennuksessa ja Kibanan asennuksen jälkeen Wazuh manager-asennus testattiin onnistuneesti toimivaksi. Wazuh managerin asennuksen jälkeen voitiinkin aloittaa Wazuh agentin asennus monitoroidulle testi työasemalle.

4.2 Wazuh agent -asennus

Tässä luvussa käydään läpi Wazuh agentin asennuksen vaiheet. Opinnäytetyössä seurattavana testikoneena käytettiin kevyttä Ubuntu Linux-versiota.

Wazuh agent oli pitkälti valmiiksi konfiguroitu jo Wazuhissa ja näin ollen asennus sujui kohtuullisen vaivattomasti. Ensimmäinen vaihe Wazuhin asennuksessa oli repositorioiden tuominen työasemalle. Ennen repositorioiden tuontia tarkistettiin varmuudenvuoksi vielä verkkoasetukset työasemalta oikeiksi siten, että työasema oli varmasti samassa lähiverkossa Wazuh-palvelimen kanssa ja yhteys näiden koneiden välillä toimi. Repositorioiden haku tapahtui kuvion 23 mukaisesti komennolla: ”apt-get install curl apt-transport-https lsb-rel gnupg2”.



Kuvio 23. Wazuh client -repositorioiden haku

Jotta yhteys toimi varmennetusti Client -agentin ja Wazuh -serverin välillä, tuotiin myös valmiiksi tehty PGP -avain Client -koneelle kuvion 24 mukaisesti komennolla: “curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | apt-key add”.

Kuvio 24. Wazuh client aget PGP -avaimen tuonti

Ennen varsinaista Wazuh clientin asennusta haettiin vielä loput Wazuh agentiin kuuluvat repositoriot komennolla: “echo “echo "deb https://packages.wazuh.com/3.x/apt/ stable main" | tee /etc/apt/sources.list.d/wazuh.list”. Tämän lisäksi ajettiin myös komento: “apt-get update”, jolla pakettidata päivitettiin Wazuh-agentin asennusta varten.

Kun edellä mainitut kohdat käytiin läpi, voitiin siirtyä itse Wazuh-agentin asennukseen. Asennus tapahtui komennolla: ”apt-get install wazuh-agent”. Wazuh-agentin asennuksen valmistuttua, täytyi Wazuh-agent myös rekisteröidä, jotta yhteys Wazuh-managerin ja Wazuh-agentin välillä toimi. Rekisteröinti toteutettiin käyttäen yksinkertaista rekisteröintimenetelmää. Yksikertaisella rekisteröintimenetelmällä rekisteröintiin käytettiin agent-auth ohjelmaa. Rekisteröinti tapahtui komennolla: ”/var/ossec/bin/agent-auth -m <MANAGER_IP_ADDRESS>”, jossa ”<MANAGER_IP_ADDRESS>” korvattiin Wazuh managerin omalla IP-osoitteella.

5 Wazuh -arviointi

5.1 Käyttöönotto

Wazuhin käyttöönotto oli itsessään kohtalaisen yksinkertainen ja selkeä kattavan dokumentaation takia. Wazuhin asennuksen aikana kuitenkin voitiin huomata, että tarvittava palvelinkapasiteetti olisi jatkossa hyvä laskea ja suunnitella tarkemmin jo ennen Wazuhin asennusta, jotta voitaisiin välttyä resursseihin kohdistuvilta ongelmilta. Riippuen ympäristön ja valvottavien laitteiden koosta ja määrästä, Wazuh voi vaatia huomattaviakin määriä kovalevytilaa, prosessorisuorituskykyä sekä RAM -muistia, esimerkiksi suuren lokimäärän indeksoimiseen, parsimiseen sekä tiedon tallentamiseen tarvittavan pitkäksi aikaa.

Tässä opinnäytetyössä kuvatussa ympäristössä Wazuh toimi erittäin pienellä palvelinkapasiteetilla ympäristöä ajettaessa henkilökohtaisella tietokoneella, jossa teho- ja

tallennus kapasiteetti oli huomattavasti tuotantoympäristöä heikompaa. Tässä opinnäytetyössä käytetyn tietokoneen tarjoama palvelinkapasiteetti rajoittikin huomattavasti ympäristöä niin sen koon ja ominaisuuksien kuin toiminnankin puolesta.

5.2 Ylläpito

Wazuhin voidaan sanoa olevan niin sanotusti huoltovapaa, sillä se sisältää itsessään kattavasti valmiita sääntöjä herätteiden nostamiseksi, eikä tästä syystä vaadi välttämättä yhtä paljon ylläpitotoimia kuin perinteinen SIEM-ratkaisu, johon voidaan joutua lisäämään sääntöjä merkittävissä määrin pitkin sen elinkaarta. Wazuh kuitenkin tarjoaa tarvittaessa mahdollisuuden myös tähän. Toki hyvällä verkon ja ympäristön perusrakenteen selvityksellä voidaan välttää suurimpia ylläpitotöitä jatkossa.

Wazuh on kuitenkin helposti konfiguroitavissa jälkikäteen ja se sopii tästä syystä myös teknisesti haastavampiinkin ympäristöihin. Tämä siitä syystä, että Wazuhiin on tarvittaessa mahdollista määrittää myös uusia sääntöjä, mikäli ympäristössä on joitain toiminteita, mitä Wazuh ei automaattisesti osaa nostaa herätteeksi. Tällainen hälytys voisi esimerkiksi liittyä tuotantoympäristön käyttäjään, jolla yrittäessä kirjautumista Wazuh hälyttäisi siitä.

Yksinkertaisemmissa ympäristöissä, joissa ei ole merkittävissä määrin erityisiä laitteita, jotka vaativat omat säännöstönsä toimii Wazuh hyvin sen omilla herätekirjastoillaankin. Tästä syystä Wazuhin voidaankin sanoa tiettyyn pisteeseen asti olevan lähes huoltovapaa ja edullinen vaihtoehto. Esimerkiksi kun huomioidaan se työaika, joka useiden SIEM -ratkaisujen asennukseen, sääntöjen luontiin ja ylläpitoon kuluu.

5.3 Luotettavuus

Opinnäytetyötä tehdessä ja Wazuhia omassa ympäristössä ajettaessa voitiin huomata, että Wazuhin luotettavuus nojaa pitkälti siihen, kuinka korkean kapasiteetin ympäristön päälle se on asennettu. Itse Wazuh toimi opinnäytetyön ympäristössä olosuhteisiin nähden vakaasti, tosin suuremmalla palvelinkapasiteetilla olisi voitu

luotettavuutta vielä parantaa. Lisäksi opinnäytetyössä Wazuhin luotettavuuden todentaminen tapahtui vain todella pienessä ja rajatussa ympäristössä, joten suuremmissa ympäristöissä voi esiintyä poikkeavuuksia eikä näin ollen laajempia johtopäätöksiä tämän opinnäytetyön pohjalta voida tehdä. Wazuhin luotettavuutta heikentävänä asiana voidaankin näiden testien pohjalta sanoa vain ympäristöön itseensä liittyvät heikkoudet eli esimerkiksi liian vähäinen prosessorikapasiteetti, vähäinen RAM-muistin määrä tai vähäinen kovalevytila suhteessa valvottavien laitteiden määrään. Tulevaisuudessa tarvittavaa palvelinkapasiteetin määrää tosin voitaisiin tuotantokäyttöön vietäessä vähentää sillä, että jätettäisiin nostamatta sellaisia herätteitä, jotka eivät tuo lisäarvoa yleiselle toiminnalle.

5.4 Skaalautuvuus

Wazuh on skaalattavissa suhteellisen helposti, kun Wazuh Server on asennettu ja toiminnassa. Tämä johtuu siitä, että Wazuh tarjoaa valmiin asennuspaketin myös kaikille uusille ympäristön laitteille ja siten hoitaa kaiken tarvittavan lokin lähetyksen saman ympäristön Wazuh Serverille. Uusien ympäristön laitteiden asennuksia pystytään myös automatiosoimaan, jotta kaikkia asennuksen vaiheita ei tarvitse manuaalisesti tehdä. Tällä voidaan säästää Wazuhin käyttöönottoon liittyvissä kustannuksissa.

Wazuh mahdollistaa myös laajemman ympäristön seurannan, mikä kattaa Applen työasemat, Windows-työasemien seurannan sekä eri Linux-jakeluilla toimivien työasemien seurannan. Tässä opinnäytetyössä perehdyttiin juuri Linuxin seurantaan ja sitä kautta näkyvyyden lisäämiseen Linux-käyttöjärjestelmiin.

5.5 Loppupäätelmät

Wazuhiin on saatavana kattava dokumentaatio lähes kaikesta, mitä sillä on mahdollista tehdä, sekä siitä, kuinka se konfiguroidaan ja asennetaan. Tämä tekeekin Wazuhista verrattain yksinkertaisen ja edullisen vaihtoehdon esimerkiksi Linux -työasemien valvontaan. On kuitenkin hyvä muistaa, että kyseessä on vapaaseen lähdekoodiin pohjautuva ohjelmisto, joten sen vieminen tuotantoympäristöön on aina tietoi-

nen riski. Tämä johtuu siitä, että takuita dokumentaation ajantasaisuudesta tai päivityksistä jatkossa ei voida varmasti odottaa saatavan. Wazuhissa dokumentaation päivityksien loppumisesta ei kuitenkaan ole toistaiseksi viitteitä ja näin ollen asennukseen liittyvät tulevaisuuden riskit ovat kohtalaisen pienet. Etenkin, kun kyseessä on ilmainen järjestelmä, jossa ainoat kulut käytännössä syntyvät tarvittavan palvelinkapasiteetin hankinnasta sekä työtunneista, joita asennus ja käyttöönotto vaatii.

Wazuh kehitettiin kilpailemaan OSSECin kanssa tarjoten päivitettyjä ominaisuuksia ja tuoden mukaan myös uusia. OSSEC on yleisesti tunnettu avoimen lähdekoodin päätelaitepohjainen IDS/IPS tietoturva-alusta. Wazuhin kehittäjät näkivät, että OSSEC ei enää vastaa nykypäivän tarpeita niin hyvin kuin se voisi, etenkin, kun sen kehittäminenkin on jäänyt niin heikolle tasolle, että nykyään käytännössä vain käyttäjien ilmoittamia ongelmia pyritään enää korjaamaan. Tästä syystä vuonna 2015 Wazuh-tiimi päättikin haastaa OSSEC-projektin tuomalla tarjolle jotain uutta ja innovatiivista. Tuloksena Wazuhin sivuston mukaan saatiinkin rakennettua paljon kattavampi, helpokäyttöisempi, luotettavampi ja skaalautuvampi tietoturva-alusta-ratkaisu. Wazuhilla on kattava käyttäjäverkosta, joten apua ongelmatilanteisiin sekä ylläpidollisiin kysymyksiin on ainakin näillä näkymin saatavissa vielä tulevaisuudessakin. Tästä syystä Wazuh onkin nostanut arvoansa ja on näin nopeasti vallannut laajasti alaa myös osana yritysympäristöjä. (Migrating from OSSEC n.d.)

Kun Wazuhia tarkastellaan kokonaisuutena, voisi sen hyvinkin nähdä yhtenä elementtinä tukemassa SOC-toimintaa niin asiakasympäristöihin, kuin Telia Cygate Oy:n omassa ympäristössäkin. Etenkin Linux-työasemien seuraamiseen Wazuh toimii hyvin.

6 Pohdinta

Opinnäytetyön tavoitteena oli arvioida Wazuhia sen lisäarvoa tuovien ominaisuuksien pohjalta. Wazuhista saatiin lopulta hyvä tuntuma ja voisin sen hyvin nähdä yhtenä osana Telia Cygate Oy:n SOC-toimintaa.

Wazuhin asennus itsessään on kohtuullisen yksinkertainen hyvän dokumentaation ansiosta, mutta ongelmia asennukseen liittyen kuitenkin tuli vastaan. Alussa aliarvioin ympäristön vaatimukset ja Wazuhin asennuksen aikana jouduinkin Wazuh-manager koneelle lisäämään erillisen virtuaalikovalevyn ja konfiguroimaan sen siten, että levy uusi levy yhdistyi aiempaan ja tila vain kasvoi. Muutoin levyn näkyessä omanaan ei Wazuhin palveluiden asennus välttämättä olisi onnistunut, sillä valmiiksi määritellyt tiedosto polut olisivat voineet vääristyä.

Kovalevytilan lisäyksen jälkeen Wazuh-asennus onnistui melko kivuttomasti ja kaikki prosessit käynnistyivät onnistuneesti. Agent koneeksi olin ajatellut käyttäväni alun perin Lubuntu-käyttöjärjestelmää käyttävää työasemaa, sillä se on yksi tällä hetkellä kevyimmistä käyttöjärjestelmistä ja näin myös suorituskykyä olisi säästetty Wazuh-manageria varten. Wazuh-agentin asennus onnistui Lubuntu koneelle, mutta Lubuntu koneen rekisteröiminen Wazuh-managerille ei onnistunut. Tähän en netistä tietoa etsimälläkään löytänyt apua, jolla ongelman olisi korjannut ja näin ongelma jäi ratkaisematta. Tästä syystä jouduin vastoin aiempaa suunnitelmaa siirtymään käyttämään normaalia Ubuntu Linux-jakelua testi työasemallani. Opinnäytetyössä esitellyn ympäristön ajoon käytettiin henkilökohtaista läppäriä, jossa suorituskyky oli erittäin rajallinen.

Tästä syystä Wazuhin testaaminen jäi paljon alkuperäistä suunnitelmaa vaatimattomammaksi. Esimerkiksi lyhyen ja erittäin pienenkin brute-force-hyökkäyksen kohdistaminen testi työasemalle riitti Wazuh-managerin kaatumiseen. Asiaa tutkittuani ja ylimääräisiä prosesseja ajettuani alas en siltikään saavuttanut tarvittavaa tehokapasiteettia järjestelmän toimimiseksi testauksen aikana.

Vaikka Wazuh-managerin herätteiden nostokykyä ei tämän opinnäytetyön teon aikana voitukaan järkevästi toteuttaa tai arvioida, saatiin Wazuhista kuitenkin kattavasti kokemusta ja ymmärrystä. Esimerkiksi siitä, miten paljon Wazuhin asennukseen tarvitaan kutakuinkin aikaa ja resursseja. Wazuhin asennus perusasetuksillaan voitaisiin saada mielestäni riittävällä palvelinkapasiteetilla pystyyn jo muutamassa päivässä. Asennuksen jälkeenkin Wazuhiin on kohtuullisen

yksinkertaista lisätä herätteitä ja valvottavia kohteita sekä muuttaa niitä. Yhtenä lisättävänä kohteena voitaisiin esimerkiksi työasemien audit-lokeja alkaa seurata.

Tulevaisuudessa mikäli Wazuh asennettaisiin Telia Cygaten tuotantoympäristöön tai asiakasympäristöihin, pitäisi aikaa kuitenkin varata Wazuhin base-linen määrittämiseksi. Wazuhissa on nimittäin niin sanotusti tehdasasetuksissaankin kohtuullisen kattava herätelista, joka sisältää monia myös mahdollisesti turhia hälytyksiä. Wazuh voisi täydessä toiminnassaan jopa haitata SOC-analyytikoiden työtä, sillä turhat hälytykset jouduttaisin kuitenkin tarkistamaan ja sulkemaan. Tämä veisi työaikaa muiden työtehtävien suorittamiseen käytettävästä ajasta. Tästä syystä Wazuhia olisikin hyvä ensin testata kattavasti esimerkiksi labra-ympäristössä ennen tuotantoon vientiä. Tällä voitaisiin myös minimoida palvelinkapasiteetin tarve.

Loppujen lopuksi opinnäytetyössä esiintyneiden ongelmien jälkeenkin koen, että opinnäytetyö on ajankohtainen ja tarpeellinen Telia Cygatelle. Opinnäytetyön teon jälkeen koen, että Wazuh voisi hyvinkin olla vastaus Linux näkyvyyden parantamiseksi kustannustehokkaasti. Wazuh on avoimenlähdekoodin järjestelmä, eikä sen käyttöönottoon siksi liity kuin tarvittavan palvelikapasiteetin hankkimisen kustannukset sekä yhden tai useamman ihmisen varaamisesta aiheutuvat kustannukset Wazuhin asentamiseksi. Opinnäytetyötä tehdessäni sain hyvää kokemusta Wazuhista ja se oli täysin uusi järjestelmä itselleni. Lisäksi opinnäytetyön aihe oli kiinnostava, joten sen tekeminen sujui hyvin pienistä haasteista huolimatta.

Lähteet

Combitech Finland. 2017a. Combitech Finlandin kirjoittama artikkeli aiheesta ”Mikä on SOC ja miksi sellainen tarvitaan?” medium.fi-sivustolla. Viitattu 11.12.2019. https://medium.com/@combitech/mik%C3%A4-on-soc-ja-miksi-sellainen-tarvitaan-a0df93609118

Combitech Finland. 18.4.2017b. Combitech Finland:in kirjoittama artikkeli aiheesta ”Mikä on SOC ja miksi sellainen tarvitaan?” medium.fi-sivustolla2. Viitattu 11.12.2019. https://medium.com/@combitech/mik%C3%A4-on-soc-ja-miksi-sellainen-tarvitaan-a0df93609118

Configuration Assessment. N.d. Tietosivu kokoonpanon arvionnista sivustolla documentation.wazuh.com. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Cloud Security Monitoring. N.d. Tietosivu pilviturvallisuuden seurannasta wazuh.com-sivustolla. Viitattu 13.1.2020. https://wazuh.com/product/#usecases

Containers Security. N.d.a. Tietosivu konttien turvallisuuden seurannasta wazuh.com-sivustolla. Viitattu 13.1.2020. https:// wazuh.com/product/#usecases

Containers Security. N.d.b. Tietosivu konttien turvallisuuden seurannasta wazuh.com-sivustolla2. Viitattu 13.1.2020. https:// wazuh.com/product/#usecases

File Integrity Monitoring. N.d. Tietosivu tiedostojen eheyden seurannasta documentation.wazuh.com-sivustolla. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Filebeat overview. N.d. Tietosivu Filebeatista yleisesti elastic.co-sivustolla. Viitattu 12.1.2020. https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-overview.html

How does Elasticsearch work?. N.d.a. Tietosivu Elasticsearchin toiminnasta elastic.co-sivustolla. Viitattu 13.1.2020. https://www.elastic.co/what-is/elasticsearch

How does Elasticsearch work?. N.d.b. Tietosivu Elasticsearchin toiminnasta elastic.co-sivustolla2. Viitattu 13.1.2020. https://www.elastic.co/what-is/elasticsearch

Incident Response. N.d. Tietosivu tietoturvapoikkeamiin vastaamisesta sivustolla documentation.wazuh.com. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Installation guide. N.d. Wazuh asennus tietosivu dokumentation.wazuh.com-sivustolla. Viitattu 4.12.2019. https://documentation.wazuh.com/3.11/installation-guide/index.html

Installing the Wazuh API. N.d. Tietosivu Wazuh API:n asennuksesta documentation.wazuh.com-sivustolla. Viitattu 13.12.2019. https://documentation.wazuh.com/3.10/installation-guide/installing-wazuh-manager/linux/centos/wazuh_server_packages_centos.html#wazuh-server-packages-centos

Indexing. N.d. Tietosivu indeksoinnista elastic.co-sivustolla. Viitattu 12.1.2020. https://www.elastic.co/guide/en/elasticsearch/client/net-api/7.x/indexing.html

Introduction. N.d.a. Tietosivu Kibanasta yleisesti elastic.co-sivustolla. Viitattu 13.1.2020. https://www.elastic.co/guide/en/kibana/current/introduction.html

Introduction. N.d.b. Tietosivu Kibanasta yleisesti elastic.co-sivustolla2. Viitattu 13.1.2020. https://www.elastic.co/guide/en/kibana/current/introduction.html

Intrusion Detection. N.d.a. Tietosivu Wazuhissa olevasta Intrusion Detection ominaisuudesta sivustolla documentation.wazuh.com. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Intrusion Detection. N.d.b. Tietosivu Wazuhissa olevasta Intrusion Detection ominaisuudesta sivustolla documentation.wazuh.com2. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Kotilainen, S. 2017. Koodi sujahtaa konttiin – sovellusten kehittäminen mullistuu uutinen tivi.fi-sivustolla. Viitattu 14.1.2020. https://www.tivi.fi/uutiset/koodi-sujahtaa-konttiin-sovellusten-kehittaminen-mullistuu/7931ccac-1cd7-3c40-b338-be25469cf1dd

Log Data Analysis. N.d. Tietosivu lokien analysoinnista documentation.wazuh.com-sivustolla. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Migrating from OSSEC. N.d. Artikkeli Wazuhin synnystä wazuh.com-sivustolla. Viitattu 13.1.2020. https://wazuh.com/migrating-from-ossec

Regular Compliance. N.d.a. Tietosivu tietoturvasäännöstön noudattamisesta sivustolla-documentation.wazuh.com. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Regular Compliance. N.d.b. Tietosivu tietoturvasäännöstön noudattamisesta sivustolla-documentation.wazuh.com2. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Rouse, M. 2019a. zero-day (computer) Artikkeli searchsecurity.techtarget.com-sivustolla. Viitattu 14.12.2019. https://searchsecurity.techtarget.com/definition/zero-day-vulnerability

Rouse, M. 2019b. Definition incident response Artikkeli searchsecurity.techtarget.com-sivustolla. Viitattu 14.12.2019. https://searchsecurity.techtarget.com/definition/incident-response

Security Analytics. N.d. Wazuh security analytics esittely dokumentation.wazuh.com-sivustolla. Viitattu 3.12.2019. https://documentation.wazuh.com/3.10/index.html

Securing the Wazuh API. N.d.a. Tietosivu Wazuhin turvallisuustason nostamisesta documentation.wazuh.com-sivustolla. Viitattu 13.12.2019. https://documentation.wazuh.com/3.10/installation-guide/securing_api.html#securing-api)

Securing the Wazuh API. N.d.b. Tietosivu Wazuhin turvallisuustason nostamisesta documentation.wazuh.com-sivustolla2. Viitattu 13.12.2019. https://documentation.wazuh.com/3.10/installation-guide/securing_api.html#securing-api)

Telia Cygate Oy lyhyesti. N.d. Telia Cygate Oy:n esittely lyhyesti Telia Cygate Oy:n www-sivuilla. Viitattu 27.12.2019
https://www.teliacygate.fi/fi/lyhyesti.

Vulnerability Detection. N.d. Tietosivu haavoittuvuuksien havaitsemisesta documentation.wazuh.com-sivustolla. Viitattu 13.1.2020. https://documentation.wazuh.com/3.10/index.html

Welcome to Wazuh. N.d. Wazuhin lyhyt esittely dokumentation.wazuh.com-sivustolla. Viitattu 3.12.2019. https://documentation.wazuh.com/3.10/index.html

What is an Intrusion Detection System?. N.d. Artikkeli paloaltonetworks.com-sivustolla. Viitattu 12.12.2019. https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-detection-system-ids