Tampereen ammattikorkeakoulu

Linux-palvelimen tietoturva

Kuinka Linux-palvelin kovetetaan ohjelmallisesti

Ville Wallenius

OPINNÄYTETYÖ
Toukokuu 2019

Tieto- ja viestintätekniikan koulutus
Tietoliikennetekniikka ja tietoverkot

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tieto- ja viestintätekniikan koulutus
Tietoliikennetekniikka ja tietoverkot

WALLENIUS, VILLE:
Linux-palvelimen tietoturva
Kuinka Linux-palvelin kovetetaan ohjelmallisesti

Opinnäytetyö 33 sivua
Toukokuu 2019

Opinnäytetyössä selvitettiin, kuinka Linux-palvelimen tietoturvaa lisätään ohjelmallisesti. Työn päätavoitteena oli selvittää, mitkä ovat Linux-palvelimen keskeiset osa-alueet tietoturvan kannalta. Työn toisena tavoitteena oli selvittää, millä tavalla osa-alueiden turvallisuus toteutetaan käytännössä.

Linux-ytimeen perustuvat käyttöjärjestelmät ovat suosittuja palvelinkäytössä. Linux-palvelimet ovat hyvin turvallisia, kunhan ne on toteutettu parhaiden käytäntöjen mukaisesti. Työn yleisluontoisuuden takia käsitteisiin liittyviä komentoja ja konfigurointiparametreja ei alettu esittelemään, vaan turvallisuuden osa-alueita käsiteltiin käsitteellisellä tasolla. Työ tehtiin Linux-pohjaisten käyttöjärjestelmien turvallisuuden osa-alueita käsittelevän kirjallisen aineiston ja kirjoittajan ammatillisen kokemuksen pohjalta.

Opinnäytetyössä tarkasteltiin osa-alueita siitä näkökulmasta, että työ selvittäisi lukijalle kokonaiskuvaa turvallisen Linux-järjestelmän asentamiseen ja konfiguroimiseen. Lopputuloksena työstä tuli kokoelma Linux-palvelimen osa-alueita ja niiden turvallisuuden lisäämiseen opastavia neuvoja. Ohjeiden laadinnassa sovellettiin Suomen Puolustusministeriön julkaiseman Katakri-turvallisuusauditointikriteeristön vaatimuksia.

Asiasanat: linux, tietoturva, kyberturvallisuus, palvelin

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in ICT Engineering
Telecommunications and Networks

WALLENIUS, VILLE:
Information Security of a Linux Based Server
The Basics of Hardening a Linux Server

Bachelor's thesis 33 pages
May 2019

The main purpose of this thesis was to compose a comprehensive collection of essential factors on security of a Linux based server. The secondary objective of this thesis was to introduce ways to make the factors more secure.

Linux based servers are very popular in production use and have potential of obtaining outstanding information security. The most fundamental security factors on this thesis were composed of a combination of Linux and security related literature and professional experience of the writer.

The security guidelines in this thesis were adopted from the security audit criteria of Finnish Ministry of Defence. The main aspect in composing the guidelines was not to introduce commands and parameters but to facilitate understanding of the essential security factors. Most essential security factors and ways to increase their security were introduced sufficiently in this thesis. Further research and literature review are required to discuss each security factor in depth. A more extensive analysis could include the commands related to the security factors discussed in this thesis.

Key words: linux, information security, cyber security, server

SISÄLLYS

1 JOHDANTO

Linux-pohjaisia järjestelmiä käytetään laajalti yritysten tuotantoympäristöissä moniin tarkoituksiin. Avoin lähdekoodi, ohjelmiston toimintavarmuus ja asiantuntijoiden määrä lisäävät ilmaisen käyttöjärjestelmän ytimen käyttäjäkuntaa. Linux-pohjaiset järjestelmät ovat yleistymässä työpöytäkäytössä, mutta palvelinkäytössä ne ovat olleet isossa roolissa jo vuosia. Vuonna 2017 tehdyn raportin mukaan julkisen pilven palvelimista 90 prosenttia, sulautetuista järjestelmistä 62 prosenttia ja supertietokoneista 99 prosenttia käyttävät Linux-pohjaista käyttöjärjestelmää (Corbet, J. 2017, 1). Esimerkiksi Facebook käyttää lukuisilla palvelimillaan Linux-pohjaisia käyttöjärjestelmiä (Solarwinds Pingdom 2019).

Suuri osa Linux-asiantuntijoiden osaamisesta on usein peräisin avoimen lähdekoodin yhteisöjen tuottamasta materiaalista. Linux-jakeluiden dokumentaatioissa, keskustelupalstoilla ja aiheeseen liittyvissä blogeissa on yleensä ohjeet tehtävään kuin tehtävään. Kuitenkin käsitetasolla Linuxin tietoturvan kannalta oleelliset osa-alueet voivat olla ainakin aloittelevalle asiantuntijalle hieman vieraita, vaikka Linux-järjestelmistä olisikin käytännön kokemusta.

1.1 Järjestelmän kovettaminen

Järjestelmän kovettaminen tarkoittaa sitä, että lisätään turvallisuutta tekemällä järjestelmän asennus ja konfigurointi parhaiden käytäntöjen mukaisesti (Boelen, M. 2018). Aloittelevan Linux-asiantuntijan on vaikea löytää lähdettä, johon olisi koottu ja selitetty kokonaisvaltaisesti Linux-järjestelmän kovettamiseen liittyvät osa-alueet. Viralliset ohjeistukset saattavat olla niin abstraktilla tasolla, että aloittelijan on vaikea yhdistää turvallisuuden käsitteitä Linux-järjestelmään ja sen osa-alueisiin. Ammatillinen kokemus Linux-palvelinten asennuksesta ja kovettamisesta on luonut kirjoittajalle kokonaiskuvaa, mistä asioista palvelimen turvallisuudessa on kyse.

Tässä opinnäytetyössä käsitellään Linux-palvelimen tietoturvaa ja kyberturvallisuutta. Työn tavoitteena on luoda kokonaiskuva osa-alueista, joista Linux-palvelimen tietoturvan kovettaminen koostuu. Linuxin tietoturvallisuuden osa-alueet esitellään käsitteinä, minkä vuoksi työssä ei keskitytä konfiguraatioparametreihin ja komentorivillä suoritettaviin komentoihin. Lisätietoa käytettävistä komennoista löytyy työssä käytetyistä lähteistä. Keskeisimpien käsitteiden esittelyn ohella käydään läpi, mitä tulee huomioida tuotantoympäristöön turvallisen Linux-palvelimen asennuksessa, konfiguroinnissa ja järjestelmäylläpidossa.

Työssä käydään myös läpi mitä asioita Suomen puolustusministeriön Katakri-vaatimusten osalta tulisi huomioida palvelimen tietoturvan kovettamisessa. Aiheita käsitellään sillä oletuksella, että palvelimen käyttöjärjestelmä on virtualisoitu. Pyrkimyksenä on kuitenkin, että samat huomiot pätevät yhtä lailla palvelimeen, jonka käyttöjärjestelmä on asennettu laitteistotasolla.

Työn päätutkimuskysymyksenä selvitetään, mistä osa-alueista Linux-palvelimen tietoturva koostuu. Jatkokysymyksenä työssä selvitetään millä tavalla tietoturvallisuus toteutetaan käytännössä.

1.2 Ohjeistukset ja vaatimukset

Katakri, eli kansallinen turvallisuusauditointikriteeristö, on Suomen puolustusministeriön julkaisema työkalu, jonka avulla viranomaiset voivat auditoida kohdeorganisaation kykyä suojata viranomaisen salassa pidettävää tietoa (Puolustusministeriö 2015, 2). Tämä voi olla huomioitava tekijä julkiselle sektorille toteutettavassa tietojärjestelmässä. Katakrin vaatimusten lähteenä on käytetty Suomen lakia ja EU:n neuvoston turvallisuussääntöjä, jotka käsittelevät tiedon suojaamista.

Katakrin ensimmäinen versio valmistui vuonna 2009 osana hallituksen sisäisen turvallisuuden ohjelmaa. Kirjoitushetkellä uusin Katakri 2015 -versio on jaettu kolmeen pääosaan. Katakri määrittelee vaatimukset turvallisuusjohtamiseen, fyysiseen turvallisuuteen ja tekniseen tietoturvallisuuteen (Puolustusministeriö 2015, 3). Tässä työssä keskitytään Katakrin teknistä tietoturvallisuutta koskevaan

osaan. Teknisen tietoturvallisuuden osassa kuvataan sähköistä käyttöympäristöä, ja osan vaatimukset ovat jaettu tietoliikenne-, tietojärjestelmä-, tietoaineisto- ja käyttöturvallisuuden osa-alueisiin (Puolustusministeriö 2015, 29). Tässä työssä käsitellään pääsääntöisesti vain tietoliikenne- ja tietojärjestelmävaatimuksia.

Ohjeistusten ylläpito- ja hallinnointivastuu siirrettiin vuonna 2014 ulkoministeriössä toimivan Kansalliselle turvallisuusviranomaiselle NSA:lle (Puolustusministeriö 2015, 3). Valtaosa teknisen turvallisuuden lähteistä ovat muiden valtioiden ja vaikuttavien yksityisten yritysten tuottamista tutkimuksista. Katakri sisältää toteutusesimerkkejä vaatimuksista, mikä vähentää tulkinnanvaraa vaatimusten toteuttamiseen.

Katakri on taustansa ja laajan käytön ansiosta hyvä apuväline tietoturvallisuuteen, vaikka tarkoituksena ei olisikaan tehdä julkiselle sektorille tietojärjestelmiä. Jos auditointityökalu on riittävän hyvä Suomen puolustusvoimille, se täyttänee auditointikriteerit vähemmän kriittisillekin järjestelmille varsin hyvin. Katakri on hyvä viitekehys tietoturvallisen järjestelmän asentamiseen ja ylläpitämiseen, ja siksi tässä työssä käytetään Katakria lähteenä tietojärjestelmän turvallisuuden perusperiaatteisiin.

2 LINUX-JÄRJESTELMÄN ASENNUSVAIHE

Järjestelmän kovettamisessa on kyse siitä, että tehdään asiat niin kuin pitää. Kovettaminen tarkoittaa, että lopullinen tavoite on lisätä järjestelmän turvallisuutta kokonaisvaltaisesti. Kovettamisen keskeisimmät asiat ovat samoja monissa käyttöjärjestelmissä, eivätkä Linux-pohjaisiset käyttöjärjestelmät ole siinä poikkeus. (Boelen, M. 2018)

Saman turvallisuuden osa-alueen toteuttavia osia voi olla monia, ja kannattaakin olla. Vertauskuvallisesti ovessakin voi olla monta lukkoa. Erilaisilla lukoilla on oma käyttötarkoituksensa, mutta kaikkien lopullinen tehtävä on rajoittaa luvatonta pääsyä. Tietojärjestelmän tietoturvassa, toisin kuin ovessa, monitasoisuus ei vaikuta merkittävästi suorituskykyyn. Tietoturvaongelmat ja palvelunestohyökkäykset taas voivat vaikuttaa merkittävästi suorituskykyyn ja estää palvelimen toiminnan kokonaan (Binnie, C. 2016, 41). Katakrissa ohjeistetaankin suojaamaan erikseen jokainen osa-alue, vaikka jonkin toisen osa-alueen suojaus kattaisikin sen jo (Puolustusministeriö 2015, 33).

Kovettamisen kanssa tulee soveltaa vähimmäistoimintojen ja vähimpien oikeuksien periaatetta (Puolustusministeriö 2015, 42). Oletusarvoisesti estetään kaikki toiminta, mikä ei ole palvelimen kannalta välttämätöntä. Tämä filosofia on yksi tietoturvallisuuden perusta. Musta lista (*blacklist*) on virustorjunnassa käytetty malli, jossa haitalliset kohteet torjutaan ja muu sallitaan, johon verrattuna valkoinen lista (*whitelist*) on käänteinen malli (Jaakkola, J. 2013). Oletusarvoisesti estävän filosofian toteuttamiseen mustien listojen sijaan tulisikin käyttää valkosia listoja.

2.1 Linux-jakelu ja sen valinta

Linux-jakelut (*distribution*) ovat käyttöjärjestelmiä, jotka koostuvat useasta eri osasta. Käyttöjärjestelmän perustana on suomenruotsalaisen Linus Torvaldsin kehittämä Linux-ydin (*Linux kernel*). Ytimen lisäksi jakeluissa on työkalupaketti, joka sisältää kriittisiä sekä tavallisiin toimintoihin liittyviä ohjelmia. Linux-jakeluihin

vakiintuneena työkalupakettina käytetään GNU-työkaluja, mistä johtuu monen jakelun yhteydessä näkyvä GNU/Linux-nimitys. (Bresnahan, C. & Blum, R. 2015, 13)

Lopulliset sovellukset eivät ole juuri jakeluriippuvaisia, mutta osassa jakeluista on erikseen jakelulle räätälöityjä sovelluksia. Esimerkiksi Ubuntussa käytetään ufw-sovellusta palomuurin hallintaan, kun taas CentOS:ssa ja Red Hat Enterprise Linuxissa palomuurin hallintaan käytetään firewalld-sovellusta (Bresnahan, C. & Blum, R. 2015, 13). Valtaosa Linux-ohjelmista, kuten edellä mainitut, voidaan kuitenkin asentaa mille tahansa jakelulle. Jos valmista asennuspakettia ei ole olemassa, se voidaan asentaa suoraan lähdekoodista kääntämällä.

Eri Linux-jakelujen julkaisusyklit poikkeavat toisistaan. Osa jakeluversioista julkaistaan tietyin aikavälein, mutta osa käyttää samaa versiota jakelun julkaisuun, jolloin uusimmat ominaisuudet tulevat nopeasti käyttöön. Esimerkiksi Arch ja Gentoo ovat jakeluita, jotka julkaistaan edellä kuvatulla ”Rolling release” -tavalla (Bresnahan, C. & Blum, R. 2015, 17). Tässä työssä keskitytään kuitenkin yrityskäyttöön soveltuviin Linux-jakeluihin, jolloin uusimmat ominaisuudet eivät ole niin suuressa arvossa kuin vakaus ja tietoturva.

Yrityskäytössä tärkeää on ohjelmistopäivitysten ja tuen elinkaari, luotettavuus ja se, että palvelimella käytetyt ohjelmistot ovat huolellisesti testattu. Suosittuja yrityskäyttöön soveltuvia jakeluja ovat Red Hat Enterprise Linux, CentOS, Oracle Enterprise Linux, SUSE Linux Enterprise Server ja Ubuntu LTS (Van Vugt, S. 2016). Osaan Linux-jakeluista on saatavilla maksullista teknistä tukea ongelmatilanteiden varalta (Negus, C. 2015, 12).

CentOS on moneen tuotantoympäristöön soveltuva täysin ilmainen ja avoimen lähdekoodin versio Red Hat Enterprise Linuxista (Linux Databook, n.d.). Työssä kirjoitetut ohjeet ovat sovellettavissa suoraan CentOS-jakeluun. Jakelu valittiin taloudellisten kustannusten minimaalisuuden, ohjelmiston vakauden ja pitkän tukiaikansa vuoksi. Työssä käsitellyt aiheet pätevät samalla tavalla kuitenkin monissa muissakin jakeluissa, mutta pieniä eroja saattaa olla.

2.2 Linux-jakelun versio ja asennusmedia

Tuotantokäyttöön tulevan Linux-jakelun version tulee olla vakaa, ja tietoturvapäivityksien tuen täytyy ulottua mahdollisimman pitkälle. Pitkä tukiaika käyttöjärjestelmän julkaisuversiolle vähentää tarvetta suurille järjestelmäpäivityksille, sillä tuen loppuessa järjestelmä täytyy päivittää uudempaan versioon tai toiseen jakeluun.

Katakrissa ohjeistetaan, että ohjelmistot tulee asentaa tarpeen mukaan niin, että palvelimella on vain järjestelmän tarvitsemat ohjelmistokomponentit (Puolustusministeriö 2015, 42). Asennusmediaa valitessa on turvallisuuden ja suorituskyvyn kannalta syytä valita minimiasennus, jossa on mahdollisimman vähän ohjelmistopaketteja mukana. Minimiasennuksen kanssa asennetaan tarpeelliset ohjelmistot sen sijaan, että joudutaan tunnistamaan ja poistamaan tarpeettomat. Jos Linux-jakelusta ei ole minimiasennusta saatavilla, voi isommasta asennusmediasta asennuksen jälkeen poistaa käyttämättömiä ohjelmistoja.

2.3 Levyosiot ja niiden liittäminen

Levyosiointi tarkoittaa fyysisen tai loogisen kiintolevyn jakamista yhteen tai useampaan osaan. Analogiana voitaisiin ajatella, että pöytälaatikosto edustaa kiintolevyä ja pöytälaatikoston vetolaatikot kiintolevyn levyosioita (Bresnahan, C. & Blum, R. 2015, 82). Levyosiointia luodessa on hyvä huomioida, mitkä hakemistot ovat palvelimen luotettavan toiminnan kannalta kriittisiä. Juuri- (*/*), koti- (*/home*), käynnistys- (*/boot*), auditointi- (*/var/log/audit*), ja lokihakemistot (*/var/log*) kannattaa asentaa kriittisyytensä takia omille erillisille levyosioilleen. Tällöin jos esimerkiksi kotihakemisto täyttyy, lokitiedostot ja käyttäjien toiminta saadaan silti tallennettua (Negus, C 2015, 220). Hyökkäystilanteet kasvattavat usein merkittävästi auditointi- ja lokitiedostoja, joten niille on hyvä varata reilusti levytilaa.

Levyosioita liittäessä (*mount*) on hyvä huomioida, että kaikilla levyosioilla ei tarvitse olla suoritusoikeutta tai oikeutta muuttaa tiedostokohtaisia käyttöoikeuksia.

Tätä voidaan säätää liittämisen valitsimilla (*mount options*). Kannattaa myös tutustua käytetyn tiedostojärjestelmään muihin liittämisvalitsimiin, sillä niistä voi olla vaihtelevasti hyötyä käyttötarkoituksen mukaan.

Oletuksena useimmissa Linux-jakeluissa levyosioita ei salata, mutta siihen tarjotaan mahdollisuus Linux-ytimen LUKS-salausmenetelmän avulla. LUKS salaa datan kiintolevyn lohkotasolla niin, että levyn liittämiseen tarvitaan avain. Kannettavien tietokoneiden yhteydessä tämä on hyvin suositeltavaa, mutta palvelimen tapauksessa hyöty voi jäädä silti vähäiseksi. Käynnistyksen jälkeen salausavain täytyy syöttää LUKS:lle, että tiedostojärjestelmää voidaan lukea ja kirjoittaa. (Disk Encryption User Guide, n.d.)

Salauksen hyöty tietosuojassa tulee siitä, että se rajoittaa pääsyä dataan fyysisen murron tapahtuessa. Kuitenkin suurin hyöty salauksesta on tiedon eheyden takaaminen, sillä tiedostoja ei pysty peukaloimaan ilman salausavainta tai pääsyä käynnissä olevaan järjestelmään. Tiedon eheyden varmistaminen vähentää riskejä haittaohjelmien ja konfiguraatiotiedostojen luvattomaan muuttamiseen. Käynnissä olevan järjestelmän datan eheyden varmistamiseen voidaan käyttää luvattoman peukaloinnin varalta yksinkertaista eheyttä tarkistavaa tunkeilijanhavaitsemisjärjestelmää, esimerkiksi aide- tai tripwire-ohjelmaa (Binnie, C. 2016).

Tiedostojärjestelmä on käyttöjärjestelmän tapa tallentaa tietoa loogiselle tai fyysiselle levyosiolle. Analogiana pöytälaatikossa tiedostojärjestelmä edustaa rengaskansiota, jossa talletettu tieto lopulta sijaitsee. Tuotantoon soveltuvia tunnettuja tiedostojärjestelmiä ovat esimerkiksi XFS, Ext4, ZoL ja BTRFS. CentOS-jakelun asennuksessa on oletusarvoisesti käytössä XFS-tiedostojärjestelmä, mutta muiden käyttäminen on myös mahdollista levyosioita luodessa.

Edellä mainituista tiedostojärjestelmistä ZoL:n ja BTRFS:n käytön etuna tietoturvan kannalta on tiedostojärjestelmän datalohkojen tarkistesummien laskeminen. Tarkistesumman laskeminen tarkoittaa matemaattisen tiivistefunktion tuottamaa lyhyttä merkkijonoa, joka voidaan laskea uudestaan tiedon eheyden tarkistamiseksi. Tämä mahdollistaa jälleen uuden tason tiedon eheyden ylläpitämiseen järjestelmässä, sekä auttaa järjestelmänvalvojaa tunnistamaan levyjärjestelmän tasolla tapahtuvia virheitä.

Jos datassa on virheellisiä lohkoja, tiedostojärjestelmä palauttaa datan sijaan virheen, sekä osaa palautua virheestä, jos dataan on lisätty redundanssikerros esimerkiksi peilaavalla levyllä. Virtuaalikoneen tapauksessa kuitenkin lohkotason tarkistesummien hyöty saattaa olla triviaalia, sillä tiedon eheyden tulisi olla varmistettu virtuaalikonemonitorin käyttämässä levyjärjestelmässä. Kuitenkin esimerkiksi levyajureiden tuottaman virheen tapauksessa tarkistesummista voisi olla hyötyä virtuaalikoneessakin.

2.4 Varmuuskopiointi ja datan redundanssi

Varmuuskopiointi tarkoittaa toissijaista kopiota alkuperäisestä datasta siltä varalta, että alkuperäinen data muuttuu käyttökelvottomaksi (Evans, C. 2014). Säännölliset ja testatut varmuuskopiot ovat tuotantoympäristössä erittäin tärkeitä (Boelen, M. 2018). Varmuuskopiot auttavat palautumaan jo tapahtuneesta vahingosta, jonka on aiheuttanut haittaohjelma, käyttäjän virhe, ohjelmisto- tai laitteisto-ongelma tai mikä tahansa muu odottamaton ongelma. Tehtyjä varmuuskopioita ei tulisi olla mahdollista muuttaa tai poistaa varmuuskopioitavalta kohteelta kiristyshaittaohjelman tai vahingossa tehtyjen muutosten varalta.

Linux-järjestelmässä ajoittainen varmuuskopiointi voidaan toteuttaa yksinkertaisimmillaan siirtämällä tiedostojärjestelmän tärkeä tieto eheyden varmistavalla kopiointityökalulla etäpalvelimelle, esimerkiksi rsync-synkronointiohjelman, SSH-protokollan avulla tunneloidun salatun tiedonsiirron ja cron-ajastusohjelman avulla. Edistyneempiä ominaisuuksia, kuten tiedostojen deduplikaatio, pakkaus ja salaus, varten tarvitaan varmuuskopiointiin erityisesti suunniteltu ohjelmisto. Deduplikaatio tarkoittaa sitä, että samaa tiedostoa ei kopioida kahta kertaa, vaan toisesta esiintymisestä luodaan vain referenssi, jos se on jo kerran ilmentynyt varmuuskopiossa. Deduplikaatio ja tiedostojen pakkaus vähentävät varmuuskopioiden tarvitsemaa levytilaa. Katakrin mukaan varmuuskopiot tulisi salata vähintään samalla salaustasolla, kuin alkuperäinenkin data (Puolustusministeriö 2015, 65). Esimerkiksi amanda ja bacula ovat tuotantokäyttöön soveltuvia varmuuskopiointiohjelmistoja.

Redundanssi tarkoittaa sitä, että datan saatavuus on varmistettu levyongelman varalta. RAID-levypinta ja virtuaalikonemonitorin (*hypervisor*) tai tiedostojärjestelmän snapshot-ominaisuus ei ole varmuuskopiointia, vaan redundanssia. Sama koskee datan replikointia, eli datan monistamista. Kyse on kuitenkin samasta datasta, ei toissijaisesta kopiosta. Snapshot-ominaisuus ottaa mahdollistaa järjestelmän tilan palauttamisen ajassa taaksepäin (Evans, C. 2014). Levypeilausta tai pariteettidataa sisältävät RAID-tasot ehkäisevät levyongelmasta tapahtuvia vahinkoja (Negus, C. 2015, 222). Tämä on virtuaalikoneen tapauksessa virtuaalikonemonitorin tehtävä. Virtuaalikoneen omille virtuaalisille kiintolevyille ei siis ole välttämätöntä luoda erillistä redundanssikerrosta.

2.5 Käyttöjärjestelmän lataaja

BIOS, tai uudemmissa koneissa UEFI, on emolevyn muistista ladattava ohjelma, joka suoritetaan ensimmäisenä tietokoneen käynnistyessä (Negus, C. 2015, 554). Sen tehtävä on etsiä ja suorittaa tietokoneen käynnistyslevyksi konfiguroidulta kiintolevyltä käyttöjärjestelmän lataajan. Valtaosa Linux-jakeluista käyttää GRUB-lataajaa. GRUB:ista voidaan valita käynnistettävä käyttöjärjestelmä, käytettävä ydin sekä syöttää parametreja käynnistettävälle käyttöjärjestelmälle. GRUB lataa Linux-ytimen levykuvan muistiin ja käynnistää sen. Linux-ydin alustaa laitteet ja laiteohjaimet, jonka jälkeen se ottaa juuritiedostojärjestelmän käyttöön (How Linux Works, 81). Useimmiten GRUB on oletusarvoisesti konfiguroitu käynnistämään käyttöjärjestelmän uusin ydin automaattisesti muutaman sekunnin kuluessa käynnistyksestä.

BIOS:lle ja UEFI:lle voi asettaa salasanan, ja se voidaan konfiguroida etsimään käyttöjärjestelmää vain käyttöjärjestelmän sisältävältä levyltä. Tämä vaikeuttaa käynnistämistä haittaohjelman sisältämältä levykuvalta. GRUB:lle voi asettaa myös salasanan, joka auttaa rajoittamaan pääsyä konsolin kautta palvelimelle. GRUB voi myös laskea käyttöjärjestelmän ytimen tarkistesumman, ja verrata sitä tiedossa olevaan tarkistesummaan. Tämä lisää luottamusta järjestelmän eheyteen ja siihen, ettei sitä ole peukaloitu.

2.6 Ohjelmistot ja palvelut

Ohjelmistojen ja käytössä olevien palveluiden määrä Linux-järjestelmässä tulee pitää niin pienenä kuin mahdollista (Puolustusministeriö 2015, 42). Tämä tarkoittaa vain tarvittavien ohjelmien asentamista sekä käyttämättömien poistamista. Linux-jakelut yleensä sisältävät pakettienhallintaan tarkoitettuja ohjelmistoja, mitkä mahdollistavat tavan automatisoida ohjelmistopäivitysten asennuksen. Uusia ohjelmistoja asentaessa tulee käyttää vain vakaita ohjelmistoversioita sekä suosia pitkää tukiaikaa tietoturvapäivityksille.

Paketinhallintaohjelmisto mahdollistaa tavan asentaa tietokoneen arkkitehtuurille sopivia ohjelmistoja ilman tarvetta kääntää niitä erikseen lähdekoodista. Ohjelmistot ovat esikäännettyjä ja sisältävät tiedon riippuvuuksista, eli muista ohjelmista ja ohjelmakirjastoista, joista ohjelman toiminta on riippuvainen. Suoraan lähdekoodista käännetyt ohjelmistot eivät välttämättä sisällä minkäänlaista tapaa automatisoida ohjelmistopäivityksiä. Siksi ohjelmistot tulee asentaa mahdollisuuksien mukaan joko käyttöjärjestelmän sisäänrakennetuilla paketinhallintaohjelmistoilla, esimerkiksi yum:lla, apt:lla tai snapd:llä. Nämä mahdollistavat ohjelmistojen ja niiden riippuvuuksien automaattisen päivityksen.

Paketit ladataan usein internetistä Linux-jakelun omista virallisista pakettilähteistä (*repository*). Kuitenkin kolmannen osapuolen pakettilähteitä lisätessä tulee muistaa, että pakettilähteen kautta on mahdollista tulla myös haittaohjelma tai järjestelmän vakautta horjuttava huono ohjelmistopaketti. Pakettilähteiden määrä tulisi kuitenkin pitää mahdollisimman minimissä. Katakri ohjeistaa, että ulkoisia ohjelmistoja asentaessa ohjelmistotuottajalta voidaan edellyttää kehittäjien riittävää tietoturvatietoutta, ohjelmistokehityksen aikana suoritettua tietoturva-analyysia ja lähdekoodin katselmointia (Puolustusministeriö 2015, 51).

Docker-, LXC- ja OpenVZ -säiliöt (*container*) ovat erinomainen tapa eristää sovellus käyttöjärjestelmästä ja monesti järjestää myös automaattiset ohjelmistopäivitykset. Säiliöt ovat isäntäjärjestelmästä eristettyjä ohjelmistokokonaisuuksia, jotka käyttäytyvät hieman virtuaalikoneen tapaan (DeMuro, J. 2018). Säiliöiden kanssa, kuten muidenkin pakettien kanssa, tulee kiinnittää huomiota ohjelmiston tukiaikaan ja tietoturvapäivitysten julkaisuviiveeseen.

Paketinhallintaohjelmistot auttavat järjestelmänvalvojaa myös pysymään helpommin ajan tasalla siitä, mitä ohjelmistoja järjestelmässä on. Ohjelmistot voidaan listata helposti muutamalla komennolla sen sijaan, että järjestelmänvalvoja joutuisi erikseen etsimään ohjelmistobinäärejä tiedostojärjestelmästä. Jos ohjelmistoja jostain syystä joudutaan asentamaan suoraan lähdekoodista, on erityisen tärkeää dokumentoida se huolella, että järjestelmänvalvoja tietää miten sen kanssa toimitaan.

Käytetyt pakettienhallintaohjelmistot ja ohjelmistojen omat automaattiset päivitystoiminnot kannattaa konfiguroida asentamaan ajoittain tietoturvapäivitykset, mutta ei ominaisuuspäivityksiä. Automaattiset ominaisuuspäivitykset saattavat luoda järjestelmään tietoturva-aukkoja tai epävakautta, joista järjestelmänvalvoja ei ole tietoinen. Katakri ohjeistaa, että ohjelmistoja ja ohjelmistopäivityksiä asentaessa tulee olla tapa todentaa päivitysten eheys (Puolustusministeriö 2015, 44). Tämä voidaan toteuttaa esimerkiksi ohjelmistojen digitaalisen allekirjoituksen varmennuksella pakettienhallintaohjelmistossa. Kun ohjelmistopakettien eheys varmennetaan, on pienempi mahdollisuus siihen, että ohjelmistoa on muutettu haitallisesti esimerkiksi väärennetyn päivityspalvelimen avulla.

3 KÄYTTÖOIKEUDET LINUX-JÄRJESTELMÄSSÄ

Käyttöoikeuksia on syytä myöntää vain pienimmän mahdollisen oikeuden periaatteella. Linux-järjestelmässä käyttäjät voivat olla ihmisten lisäksi myös ohjelmia tai palveluita. Varsinkin ohjelmille on syytä myöntää vain tarvittavat oikeudet, mutta myös ihmiskäyttäjille. Kyse ei ole vain luottamuksesta, vaan pahimpaan varautumisesta. (Boelen, M. 2018)

Vanhoja käyttäjiä otettaessa pois käytöstä ne olisi hyvä jäädyttää koko käyttäjän poistamisen sijaan. Jäädyttämisen lisäksi olisi hyvä etsiä käyttäjän omistamat tiedostot ja joko poistaa käyttäjän tiedostot tai antaa niihin omistajuus toiselle käyttäjälle (Negus, C. 2015, 595). Järjestelmään automaattisesti luotujen käyttäjien käyttöoikeudet rajataan kaikin puolin minimiin, tai otetaan käyttäjät kokonaan pois käytöstä (Puolustusministeriö 2015, 42). Järjestelmää ylläpitäessä olemassa olevien käyttäjien oikeuksia tulisi myös ajoittain auditoida, ja miettiä, ovatko käyttäjien käyttöoikeudet tarpeellisia.

Järjestelmää kovettaessa minimoidaan myös ihmisen virheestä tapahtuvia tietoturvariskejä. Fyysiselle konsolille täytyy asettaa automaattinen lukitus, esimerkiksi salasanasuojattu näytönsäästäjä, joka aktivoituu 15 minuutin käyttämättömyyden jälkeen. Katakrissa vaaditaan aikakatkaisua etäyhteysistunnoissa ja muissa kirjautumista vaativissa hallintasovelluksissa (Puolustusministeriö 2015, 42).

3.1 Pääkäyttäjäoikeudet

Vain fyysiseltä konsolilta tulisi olla mahdollista kirjautua pääkäyttäjänä, tämäkin hätätapauksia varten. Pääkäyttäjän oikeuksia käytetään asennusvaiheen jälkeen vain sudo-komennon avulla. Tällöin pääkäyttäjänä ei tarvitse koskaan kirjautua, ja oikeuksia voidaan rajoittaa käyttäjän tarpeiden mukaan. Esimerkiksi WWW-palvelimen ylläpitäjällä on oikeus sammuttaa ja käynnistää vain WWW-palvelu, mutta ei järjestelmän muita palveluita. Sudo-komennon käyttämisestä jää lokiin merkintä, jolloin voidaan seurata, kuka on käyttänyt pääkäyttäjäoikeuksia ja milloin. (OS Protection, n.d.)

Sudo-komennolla voidaan pääkäyttäjäoikeuksilla suorittaa komentoja myös muinakin käyttäjinä, kuin pääkäyttäjänä. Oletusarvoisesti toisena käyttäjänä komentojen suorittaminen tulisi estää, vaikka se jättäisikin jäljen lokitiedostoon. Jäljitettävyyden selkeyden vuoksi sudo-komennolla tulisi voida suorittaa komentoja vain pääkäyttäjänä. Oletuksena sudo-komentoa suoritettaessa käyttäjän täytyy syöttää salasanansa ajaakseen komennon. Kuitenkin jos käyttäjällä ei ole salasanaa vaan jokin muu tapa kirjautua käyttäjälle, tällöin sudo-komentoa ei voida käyttää. Sudon konfiguraatiotiedostossa voidaan määritellä mahdollisuus suorittaa komentoja ilman tarvetta syöttää käyttäjän salasanaa.

3.2 Tiedostotason käyttöoikeudet ja roolikohtaiset oikeudet

Tiedostotason käyttöoikeudet, eli DAC, on jaoteltu kolmeen osaan: käyttäjä, ryhmä ja muut. Tiedosto-oikeuksia voi rajoittaa yksittäisen osan lisäksi myös pääsylistalla, eli ACL:llä, jos tiedostolle tai kansiolle on tarkoitus määrittää pääsyrajoitusta usealle eri käyttäjälle tai ryhmälle (Negus, C. 2015, 271). Tiedostotason käyttöoikeuksien myöntäminen järjestelmissä tulee tehdä huolellisesti, sillä liian avomieliset käyttöoikeudet johtavat helposti tietoturvariskeihin.

Tiedostojen oletusoikeudet asetetaan kansiokohtaisesti asettamalla oikeuksille umask-järjestelmäkutsulla (Ward, B. 2015, 35). Tällöin kaikkiin uusiin luotuihin tiedostoihin tulee asetetut oikeudet. Kriittisten tiedostojen, kuten ohjelmien konfiguraatiotiedostojen, SSH-avainten ja digitaalisten sertifikaattien hakemistot tulee suojata mahdollisimman vähäisillä tiedostotason käyttöoikeuksilla. Monissa tapauksissa vain pääkäyttäjä tarvitsee kirjoitusoikeuden tiedostoihin, mutta yksityisten avainten tapauksessa lukeminenkin tulisi estää kaikilta muilta, paitsi pääkäyttäjältä.

SELinux on Linux-ytimeen ohjelmoitu tapa asettaa ja hallita roolikohtaisia käyttöoikeuksia (Negus, C. 2015, 669). Roolikohtaiset käyttöoikeudet auttavat tehokkaasti rajoittamaan käyttöoikeuksia prosesseille, mutta sen käyttöönotto vaatii järjestelmänvalvojalta syvällistä perehtymistä sen toimintaan. Roolikohtaisilla käyttöoikeuksilla voidaan rajoittaa ohjelmiston pääsyä tiedostojärjestelmässä

vain tarvittaviin hakemistoihin. Luku, kirjoitus- ja suoritusoikeutta voidaan rajoittaa myös erikseen roolikohtaisesti. Sillä voidaan rajata tehokkaasti ohjelmistovirheiden mahdollistamista tietoturva-aukoista johtuvaa haittaa. Katakri-vaatimuksissa mainitaan, että järjestelmän lisäturvallisuusominaisuudet tulee ottaa käyttöön, mikä tarkoittaa muun muassa SELinuxin hyödyntämistä (Puolustusministeriö 2015, 42).

3.3 Tunnistautuminen

Käyttäjän tunnistautumisesta vastaa PAM-tunnistusjärjestelmä (*Pluggable Authentication Modules*). PAM on Sun Microsystemsin 1995 kehittämä Solaris-käyttöjärjestelmän osa, jonka pohjalta tehtiin Linuxin PAM-projekti, jota Linux-jakelut nykyään käyttävät tunnistautumiseen. Tunnistusta vaativa sovellus esittää käyttäjän tunnistuspyynnön PAM:ille, joka varmistaa käyttäjän salasanan tai sormenjäljen. (Negus, C. 2015, 648)

Käyttäjätiedot tallennetaan /etc/passwd-tiedostoon, jossa on tieto käyttäjutunnuksen nimestä, salasanasta, käyttäjätunnistenumerosta, ensisijaisen ryhmän tunnistenumerosta, käyttäjän oikeasta nimestä, kotihakemistosta ja oletuskomentotulkista (How Linux Works, 154). Salasana-kentässä on yleensä kuitenkin useimmiten selväkielisen salasanan sijaan vain x-kirjain kertomassa, että salasana löytyy sekoitettuna /etc/shadow-tiedostosta. (Shadow-tiedosto, 2015) Pääkäyttäjälle ei ole syytä asettaa salasanaa, sillä sille ei tulisi koskaan kirjautua konsolilta tai etäyhteydellä. Tavalliset käyttäjät voivat ajaa komentoja pääkäyttäjänä su- ja sudo-komentojen avulla. (Negus, C. 2015, 178)

Sekoitettu salasana (*hash*) tarkoittaa kryptografisen tiivistefunktion tuottamaa tiivistesummaa. Sitä ei voi siis muuttaa enää takaisin alkuperäiseen selväkieliseen muotoon. Tällä tavalla salasanoja voidaan säilöä niin, ettei selväkielistä salasanaa saada tietokannasta selville. Käyttäjän tunnistautuessa järjestelmään syötetyn salasanan tiivistesummaa verrataan säilöttyyn tiivistesummaan. Suosittuja tiivistefunktioita ovat esimerkiksi SHA256, SHA512, whirlpool, tiger, ripemd ja

SHA3 (Binnie, C. 2016, 102). CentOSin vuonna 2016 päivitetyssä dokumentaatiossa suositellaan salasanojen sekoitukseen käytettävän SHA512-algoritmia (OS Protection, n.d.).

Käyttäjillä on taipumus käyttää helposti muistettavia, ja usein myös helposti koneellisesti tai käsin arvattavia salasanoja. Koneellinen väsytyshyökkäys on yleinen menetelmä saada luvaton pääsy järjestelmään. Väsytyshyökkäys tarkoittaa salasana-autentikaation kontekstissa erilaisten salasanavariaatioiden kokeilemista, joskus älykkään algoritmin avustamana. Siksi on hyvä luoda /etc/login.defs -tiedostossa sääntöjä, jotka eivät salli heikkojen salasanojen käyttämistä. Väsytyshyökkäyksiä vastaan salasanan entropian, eli pituuden ja merkkien vaihtelevuuden, suuruus on tehokas puolustus. (Negus, C. 2015, 596)

Hyvän salasanan tulisi CentOS -Linux-jakelun dokumentaation mukaan sisältää pieniä ja suuria kirjaimia, erikoismerkkejä, numeroita, ja sen tulisi olla vähintään kahdeksan merkkiä pitkä (Centos OS Protection). Sertifioitu Linux-asiantuntija Cristopher Negus suosittelee kuitenkin vähintään 15 merkin pituista salasanaa (Negus, C. 2015, 597). Tietoturva-asiantuntija Matthew Monte huomauttaa, että käyttäjillä on luontainen taipumus keksiä laadultaan huonoja salasanoja määrätyistä salasanasäännöistä huolimatta (Network Attacks & Exploitation).

Yhdysvaltalainen teknologiastandardeja ylläpitävä viraston NIST:n tunnistautumista käsittelevässä julkaisussa mainitaan, että koska käyttäjät yrittävät keksiä helposti muistettavia salasanoja, niistä tulee sivutuotteena myös laadultaan huonoja. Salasanojen sijaan suositellaan käytettävän epäsymmetrisiä salausalgoritmeja ja ohjelmallisesti generoituja avaimia, jotka ovat turvallisempi tapa tunnistautua. Kryptografisten avainten entropia on luonnostaan huomattavasti suurempi kuin käyttäjän itse keksimän salasanan. (Fenton, J., Garcia M. & Grassi, P. 2017, 13)

4 LINUX-JÄRJESTELMÄ TIETOVERKOSSA

Edellisissä luvuissa käsitellyt aiheet ovat valmistelua siihen, että palvelin voidaan kytkeä tietoverkkoon. Vaikka palvelin olisi erillisen palomuurin takana, verkkoturvallisuus tulee huomioida käyttöjärjestelmässäkin tarkasti. Jos palvelimeen kytketään useita eri verkkoja, niistä tulee olla selvä dokumentaatio, että verkkojen luottamusalueet voidaan ottaa huomioon palvelinta kovettaessa. Tuntemattomia verkkoja ja internetiä kohdellaan aina vähimmän mahdollisen luottamuksen alueina.

4.1 Palomuuri

Palomuuri on kahden tai useamman pisteen välistä tietoliikennettä suodattava pääosin verkon kuljetuskerroksella toimiva komponentti. Linux-jakeluissa IP-liikennettä hallitaan netfilter-järjestelmän avulla avulla. Netfilterin sääntöjen hallintaan käytetään usein iptables-ohjelmaa. Palomuurisäännöt sallivat yleensä oletuksena kaiken verkkoliikenteen, ja joskus palomuuri saattaa olla oletuksena kokonaan pois käytöstä. Palomuuria on mahdotonta konfiguroida kaikkiin tarkoituksiin sopivaksi etukäteen, ennen kuin palvelimen verkkoliikenteen tarpeita tunnetaan, sillä palomuurisäännöt riippuvat täysin palvelimen käyttötarkoituksesta. Katakri-vaatimusten täyttämiseksi vain erikseen hyväksytty, toiminnalle välttämätön liikennöinti sallitaan, mutta oletusarvoisesti kaikki muu liikenne estetään (Puolustusministeriö 2015, 33).

Palomuurisääntöjen hallintaan kannattaa käyttää hallinnan selkeyden vuoksi esimerkiksi firewalld- tai ufw-ohjelmaa. Näissä ohjelmissa voidaan konfiguroida palomuurisääntöjä verkon erilaisille luottamusalueille. Esimerkiksi yrityksen sisäverkossa voi olla sallittua pääsy etähallintaan, mutta internetistä on pääsy vain webpalvelimen tarjoamalle HTTP-sisällölle.

Hyvä periaate on kieltää kaikki sisääntuleva liikenne ja sallia vain tarvittavat protokollat, joihin palvelimelle on tarkoitus tulla verkkoliikennettä kustakin verkosta ja verkkokortista. Esimerkiksi SSH-etähallinnan voi sallia vain rajatuista IP-osoitteista, mutta HTTP-protokolla halutaan monesti sallia kaikkialta.

Ulosmenevän liikenteen osalta palomuurisääntöjen luomiseen tulee selvittää, minkälaisia protokollia palvelut käyttävät. Ulosmenevää liikennettä voi rajoittaa samalla periaatteella kuin sisääntulevaa, eli kielletään oletusarvoisesti kaikki, ja sallitaan vain halutut protokollat haluttuihin verkkoihin. Ulosmenevän liikenteen rajoittaminen ei ole yhtä tärkeää kuin sisääntulevan, sillä palvelimen sisäinen verkkorajapinta mielletään usein korkeimman luottamuksen luottamusalueeksi. Ulosmenevän liikenteen rajoitus sisältyy Katakri-vaatimuksiin (Puolustusministeriö 2015, 42).

Jos kriittisiä verkkopalveluita halutaan tarjota heikomman luottamusalueen yli, voidaan harkita myös kahden pisteen tai verkon välistä salattua verkkotunnelointia, esimerkiksi IPsec tai tai OpenVPN -protokollia. Tällöin heikomman luottamusalueen verkkoliikenne saadaan minimoitua, eikä verkkosovellus ole avoimena hyökkäyksen kohteena heikommalla luottamusalueella.

4.2 Automatisoidut palomuurisäännöt

Automatisoidut dynaamiset palomuurisäännöt torjuvat tehokkaasti väsytyshyökkäyksiä. Verkkoliikenteen lokeja tarkkailevalle palvelulle voidaan määrittää, kuinka monta kertaa esimerkiksi SSH-etähallinnan kirjautumisyritys saa epäonnistua samasta IP-osoitteesta. Tällöin riittävä määrä lokitiedostoista luettuja epäonnistuneita kirjautumisyrityksiä määritetyssä aikaikkunassa luo palomuurisäännön, joka estää kyseisestä IP-osoitteesta kirjautumisen väliaikaisesti tai pysyvästi. Tällä voidaan rajoittaa väsytyshyökkäyksiä, ja vähentää niihin kuluvia palvelinresursseja. Automaattisia palomuurisääntöjä väsytyshyökkäyksiä vastaan voidaan konfiguroida esimerkiksi Linux-ytimen iptablesilla tai avoimen lähdekoodin fail2ban -ohjelmistolla.

Hyökkääjät voivat etsiä kohteita skannaamalla automatisoidusti verkossa avoimia portteja (Binnie, C. 2016, 69). Verkkopalvelun käyttämän portin muuttaminen epästandardiksi saattaa vähentää siihen kohdistuvia hyökkäysyrityksiä, mutta se ei silti piilota palvelua verkkoskannerilta. Sen sijaan palvelun voi piilottaa porttikoputuksen taakse. Tällöin palvelin ei kuuntele porttia, ennen kuin palvelimeen

”koputetaan”, eli yhdistetään lyhyesti, yhden tai useamman minkä tahansa portin yhdistelmällä. Lopullinen tietoliikenneyhteys kuitenkin avataan protokollan alkuperäiseen porttiin. Automaattiset palomuurisäännöt porttikoputusta varten voidaan luoda esimerkiksi Linux-ytimen iptablesilla, avoimen lähdekoodin fail2banilla tai avoimen lähdekoodin knockd:llä. Porttikoputus on hyödyllinen silloin, kun etähallinta halutaan sallia heikolle luottamusalueelle, esimerkiksi internetiin. Monien palveluiden kannalta porttikoputus heikentää palvelun käytettävyyttä, jolloin kannattaa harkita mieluummin verkkotunneloinnin käyttämistä.

4.3 Sovelluskerroksen tietoliikenteen säätely

TCP Wrapper on käyttöoikeuskirjasto Linuxin IP-protokollien liikennöinnin säätelyyn. Se on yksinkertainen ja nopea tapa rajoittaa palveluiden verkkoliikennettä verkon sovelluskerroksella. Sillä voidaan hallita palomuurin läpäisemää liikennettä palvelukohtaisesti (Pillai, S. 2013). Monitasoinen verkkosuojaus voi tuntua joissain tapauksissa tarpeettomalta, mutta etenkin silloin täytyy muistaa, että TCP Wrapper toimii eri tasolla kuin palomuuri.

TCP Wrapperin etu järjestelmänvalvojan kannalta on se, että sillä voidaan hallita useita eri sovelluksia samasta paikasta. TCP Wrapper koostuu kahdesta tiedostosta: /etc/hosts.allow ja /etc/hosts.deny, joihin määritellään sallittu ja kielletty verkkoliikenne palvelua kohden. Hyvä tapa on oletusarvoisesti kieltää kaikki, ja sallia tarpeen mukaan liikennettä palvelukohtaisesti. Esimerkiksi etähallinta, sähköposti ja tiedostojen yhteiskäyttöön liittyvät protokollat kannattaa sallia vain tietoverkoista, joista niitä käytetään.

4.4 Etähallinta

SSH on TCP/IP-protokolla, jota käytetään Linux-palvelimien etähallintaan. SSH-protokollan yli lähetetty liikenne on salattua, ja siksi OpenSSH on korvannut telnet-protokollan yleisimpänä etähallintaprotokollana. Tunnistautumisvaiheessa lähetettävä salasana ei ole verkossa nähtävissä selväkielisenä, vaikka verkkoliikennettä pääsisikin tarkkailemaan. (Securing OpenSSH, n.d.)

SSH-palvelun konfiguroimisessa on tärkeää estää pääkäyttäjän etähallinta. Jos kaikilla käyttäjillä ei ole tarvetta etähallintaan, etähallinta voidaan sallia vain rajatuille käyttäjille. SSH-protokollasta on kaksi versiota, joista ensimmäinen ei ole enää turvallinen, ja siksi kannattaa sallia vain uudemman protokollan käyttäminen. Kun SSH on konfiguroitu ja testattu, salasanakirjautuminen voidaan ottaa pois käytöstä, ja käyttää tunnistautumiseen pelkästään kryptografisia avaimia (Securing OpenSSH, n.d.). SSH-etähallinnan kanssa voidaan käyttää myös kaksivaiheista tunnistautumista. Tämä on hyvä lisä etenkin, jos SSH-portin täytyy olla avoin heikon luottamuksen luottamusalueeseen, kuten internetiin.

Etähallintayhteyksissä tulisi käyttää istuntojen aikakatkaisua, ja hallinta ei saa olla mahdollista ilman käyttäjän turvallista tunnistautumista ja todentamista (Puolustusministeriö 2015, 42). Etähallintayhteyksienkin suhteen tulee soveltaa pienimmän mahdollisen oikeuden periaatetta. Tämän takia SSH-palvelu konfiguroidaan whitelisting-mallin mukaan oletuksena estämään kirjautuminen kaikilta muilta, paitsi sallituilta käyttäjiltä.

4.5 Yleiset sovellukset ja verkkopalvelut

Sovellusten tarvitsemat porttiavaukset konfiguroidaan palomuuriin sovelluskohtaisesti. Palomuurissa määritellään pääsy kuhunkin verkkopalveluun kustakin luottamusalueesta. Verkkopalveluiden turvallisuus tulee huomioida myös sovellustasolla. Tämä voidaan toteuttaa TCP Wrapperin avulla, mutta myös sovelluksen omissa konfiguraatioissa.

Sovelluskohtaiset asetukset kannattaa konfiguroida vasta kun palomuuri ja TCP Wrapper on konfiguroitu sovellukselle. Sovelluksen tarjoamat mahdollisuudet verkkoliikenteen ja pääsyn rajoitukseen vaihtelevat sovellusta kohden, ja siksi niitä asentaessa ja konfiguroidessa tulee perehtyä sovelluksen dokumentaatioon ja parhaisiin käytäntöihin.

4.5.1 WWW-palvelin

WWW-palvelimen, eli web-palvelimen, tehtävä on kuunnella HTTP-protokollaa ja vastata dokumenttipyyntöihin TCP/IP-tasolla. Web-palvelimet konfiguroidaan usein kuuntelemaan verkkoa vähäisen luottamuksen alueella, ja siksi sen kovettamiseen kannattaa kiinnittää huomiota. WWW-palvelinta ei saa suorittaa koskaan pääkäyttäjänä, vaan WWW-palvelimella tulee olla oma käyttäjä vähimpien oikeuksien periaatteella (Negus, C. 2015, 319).

Riippumatta käytetystä palvelinohjelmistosta, samat perusperiaatteet pätevät. WWW-palvelimelle asennetaan vain tarvittavat ohjelmistomoduulit ja käyttämättömät poistetaan. Palvelimelle asennettavien web-sovellusten kanssa pätevät samat perusperiaatteet, kuin muidenkin ohjelmistojen kanssa. Kuitenkin käyttäjän turvallisuuden lisäämiseksi HTTP-liikenteen salaaminen HTTPS-protokollalla on suositeltavaa. SSL/TLS-protokolla mahdollistaa tietoliikenteen salaamisen lisäksi sen, että palvelimen identiteetti voidaan varmentaa digitaalisella varmenteella luotetulta kolmannelta osapuolelta (Negus, C. 2015, 465-466).

4.5.2 Tietokanta

Linuxissa yleisesti käytettyjä tietokantaohjelmistoja ovat MySQL, MariaDB ja PostgreSQL (Negus, C. 2015, 318). Tietokannat asetetaan mahdollisuuksien mukaan kuuntelemaan vain paikallista verkko-osoitetta (Boelen, M. 2018). Tällöin tietokannan verkkoturvallisuus ei ole riippuvainen pelkästä palomuurista ja TCP Wrapperista, vaan tietokanta ei hyväksy palvelimen ulkopuolelta tulevia yhteyksiä, vaikka niitä pääsisikin läpi. Joskus tietokantaan tulee olla pääsy myös verkosta, ja tällöin kannattaa tietokantapalvelun turvaamisessa kiinnittää huomiota erityisesti käyttäjien käyttöoikeuksien rajaamiseen ja väsytyshyökkäysten torjuntaan.

Tietokantaa käyttävien käyttäjien autentikaatio voidaan toteuttaa joko Linuxin PAM-moduulin kautta tai tietokantaohjelmistossa. Tietoturvan hallinnan kannalta PAM-moduulin kautta tapahtuva autentikointi vähentää erillisten riskitekijöiden lukumäärää. Tietokantojen kanssa on hyvä muistaa, että järjestelmänvalvojalla on

usein pääsy myös tietokantojen sisältöön, ja samoin muilla käyttäjillä, joille on annettu pääkäyttäjäoikeudet tietokannan hallintaan.

4.5.3 Tiedostojen yhteiskäyttö

Tiedostojen yhteiskäyttöön soveltuvia protokollia ovat esimerkiksi NFS, WebDAV ja SMB (Bresnahan, C. & Blum, R. 2015, 64). Niitä käytetään tiedostojen jakamiseen verkossa tiedostotasolla, mutta ei lohkotasolla, johon on omat protokollansa. Tiedostojen yhteiskäytön turvaamisessa käyttäjäoikeudet tulee rajata niin vähäisiksi kuin vain mahdollista. Vieraskäyttäjillä ei saisi olla koskaan kirjoitusoikeutta, ja vieraskäyttäjän olemassaoloa tulisi välttää muutenkin. Tiedostojen käyttöoikeutta voidaan rajata tiedostokohtaisissa käyttöoikeuslistoissa, mutta monessa tapauksessa sen lisäksi myös tiedostojakoon käytetyn ohjelmiston konfiguraatiotiedostossa.

Vuonna 2017 kiristyshaittaohjelma WannaCry hyödynsi Windows-käyttöjärjestelmässä ollutta haavoittuvuutta SMB-protokollassa, mikä mahdollisti verkkojakojen saastuttamisen ja tiedostojen salaamisen käyttökelvottomiksi (Virtanen, J. 2017). Haavoittuvuuden hyväksikäyttöä pystyi rajoittamaan rajaamalla hyökkäyspintaa estämällä SMB-pääsy internetistä (Lehto, T, 2017). Tämä ei kuitenkaan estänyt haittaohjelman leviämistä palomuurin luottamusalueen sisäpuolella. SMB-protokollaa käyttäessä kannattaakin käyttää protokollan uusinta mahdollista versiota.

Tiedostojen yhteiskäyttöön käytettävien protokollien kanssa tulee varmistaa, että autentikointi ja verkkoliikenne on salattu turvallisella menetelmällä. Vuonna 2014 julkaistut Heartbleed ja Shellshock -haavoittuvuudet koskivat SSL-salauksen versioita 2.0 ja 3.0, sekä joitakin TLS-salauksen variaatioita, joita on pidetty kryptografisten ominaisuuksiensa puolesta hyvin turvallisina (Rousku, K, 2014). Nämä ovat hyvä esimerkki siitä, että järjestelmänvalvojan tulee olla tietoinen järjestelmän käyttämistä protokollista ja niiden turvallisuudesta myös asennuksen jälkeen.

4.5.4 Muut sovellukset ja säiliöinti

Verkossa toimivien sovellusten kanssa kannattaa perehtyä sovelluksen dokumentaatioon ja parhaisiin käytäntöihin. Jos ohjelmisto on uusi tai sitä ei pidetä huolellisesti testattuna ja auditoituna, kannattaa kiinnittää erityistä huomiota Linux-ytimestä löytyviin työkaluihin, joilla voidaan rajoittaa sen käyttöoikeuksia ja verkkoliikennettä. Kannattaa perehtyä sovelluksen käyttämiin protokolliin ja ohjelmistoriippuvuuksiin. Sovellus itsessään voi olla turvallinen, mutta sen käyttämät ohjelmistoriippuvuudet voivat olla ikänsä tai testauksen puutteesta tietoturvariski.

Docker, LXC ja OpenVZ -säiliöt mahdollistavat sovellusten eristämisen myös verkkotasolla, sillä niille voidaan luoda oma muusta järjestelmästä eristetty verkkotopologia (DeMuro, J. 2018). Jos sovellus koostuu monesta komponentista, jokaiselle komponentille voidaan tehdä oma säiliö ja palomuurisäännöt. Varsinkin testausvaiheessa säiliöinti on hyödyllistä, kun sovelluksen ympäristö on eristetty omaan ympäristöönsä, jossa ohjelmistoriippuvuuksista voidaan asentaa eri versiot kuin isäntäjärjestelmässä. Lopullinen tuotantoympäristö voidaan asentaa helposti testialustan pohjalta omaksi säiliökseen.

Sovellusten säiliöinti tuo uuden turvallisuuskerroksen, kun ohjelmisto on eristetty käyttöjärjestelmän ytimestä ja verkosta (What is a Container, n.d.). Tällöin ilman roolikohtaisiakin käyttöoikeuksia voidaan rajata vaarantuneen sovelluksen hyökkäyspinta-alaa pienemmäksi. Esimerkiksi tietokantapalvelin voi olla erillään webpalvelimesta, jolloin onnistunut murtautuminen staattisia sivuja tarjoavalle webpalvelimelle näkyy parhaimmillaan kolahduksena järjestelmänvalvojan itsetunnossa. Pahimmillaan säiliöinti voi luoda valheellisen tunteen turvallisuudesta, mutta parhaimmillaan se voi helpottaa turvallisemman palvelinarkkitehtuurin ketterää suunnittelua ja verkkoturvallisuuteen liittyvien rajoitusten asettamista.

5 VALVONTA

Linux-järjestelmän pääsyn ja käyttöoikeuksien rajoittaminen lisää turvallisuutta, mutta ilman tietoa järjestelmän ajankohtaisesta tilasta järjestelmän vaarantumisesta saatetaan olla täydessä pimennossa (Boelen, M. 2018). Järjestelmän suorituskyvyn hyvinvoinnin valvonta on tärkeää, mutta vielä tärkeämpää on pyrkiä olemaan mahdollisimman tietoinen palvelimella tapahtuvasta luvallisesta ja luvattomasta toiminnasta. Tietoisuuden lisäämiseen käytetään valvontaa sekä palvelimella paikallisesti, että verkkopohjaisesti ulkoisella valvontapalvelimella. Katakri-vaatimuksiin sisältyy verkkoliikenteen valvonta siihen soveltuvalla sovelluksella (Puolustusministeriö 2015, 34).

Valvonta on yksi osa tietoturvallisuutta, sillä kovetettujakin järjestelmiä tulee käsitellä sillä ajatuksella, että niiden turvallisuus voi vaarantua milloin tahansa. Hyvin suunniteltu valvontajärjestelmä saattaa auttaa järjestelmänvalvojaa huomaamaan valvottavassa kohteessa epäkohdan, ennen kuin se aiheuttaa ongelman (Boelen, M. 2018).

5.1 Valvontaohjelmisto

Palvelimen järjestelmätietojen valvonta auttaa järjestelmänvalvojaa havaitsemaan tietomurtoja ja muita ongelmia. Ilman valvontajärjestelmää tietomurrot voivat jäädä järjestelmänvalvojalta täysin huomaamatta (Boelen, M. 2018). Valvonta voi tapahtua paikallisesti palvelimella, mutta useiden palvelimien valvonta kannattaa toteuttaa verkkopohjaisen valvontajärjestelmän kautta (Negus, C. 2015, 344). Tuotantoympäristöön voidaan käyttää esimerkiksi vapaan lähdekoodin Nagios- tai Zabbix -valvontaohjelmistoja.

Valvontaohjelmisto konfiguroidaan ilmoittamaan järjestelmänvalvojalle valvottavalla kohteella tapahtuvista poikkeamista järjestelmän tilassa. Valvonta voi olla joko aktiivista tai passiivista. Aktiivinen valvottava kohde lähettää itse tietoa valvontajärjestelmälle, kun taas passiiviselta kohteelta valvontajärjestelmä pyytää ajoittain tietoa.

Kohteesta voidaan valvoa esimerkiksi prosessien tilaa, verkkoliikenteen määrää ja laatua, levytilan käyttöastetta sekä kriittisten tiedostojen tarkistesummia. Palvelimen käyttötarkoituksen mukaan sovelletaan valvontaohjelmiston parhaita käytäntöjä kohteen tilan arvioimiseen. Esimerkiksi tietokantapalvelimen valvomiseen voidaan käyttää tietokantaohjelmiston omia suorituskyvyn ja käyttöasteen indikaattoreita, sekä käyttöjärjestelmän levyaktiviteettia ja käyttöviivettä mittaavia työkaluja.

5.2 Lokien kerääminen ja auditointi

Palvelimen tapahtumat ja käyttäjän tekemät muutokset tulee taltioida niin yksityiskohtaisesti kuin mahdollista. Muun muassa CentOS-jakelun palomuuri ei kerää oletuksena haitallisesta verkkoliikenteestä lokidataa juurikaan (OS Protection, n.d.). Lokien tallentaminen on erittäin tärkeää sekä mahdollisten vikatilanteiden selvittämiseen, mutta myös tietoturvan kannalta. Automaattiset palomuurisäännöt ja tunkeilijanhavaitsemisjärjestelmät toimivat lokitietojen pohjalta.

Käytön seuranta, eli auditointi, lisää informaation määrää potentiaalisissa ongelmatilanteissa. Linux-palvelimella onkin hyvä asettaa auditd-palvelu päälle, jolloin voidaan tarkkailla esimerkiksi suoritettuja bash-komentoja, luvattomia sudo-yrityksiä ja epäonnistuneita kirjautumisyrityksiä (UTD CSG, 2018). Tämä on etenkin hyödyllistä ongelmatilanteiden selvittämisessä, mutta myös helpottaa järjestelmänvalvojaa pysymään ajan tasalla palvelimen tilasta.

Lokien keskitetty kerääminen tarjoaa tavan tarkkailla useiden palvelimien kuntoa yhdestä paikasta. Tällöin palvelimet lähettävät lokinsa keskitetylle lokipalvelimelle, jolla voidaan tarkkailla usean palvelimen tilannetta yhdestä portaalista (Negus, C. 2015, 334). Tästä on etua myös siinä tapauksessa, että palvelin on mennyt sellaiseen kuntoon, että lokitietoja ei päästä lukemaan (Negus, C. 2015, 335). Näin voi käydä vikatilanteen tai tietomurron yhteydessä. Levytilan säästämiseen tarkoitettu lokitiedostojen rotaatio tarkoittaa vanhojen lokimerkintöjen poistamista uusien tieltä. Tämä antaa hyökkääjälle mahdollisuuden pyyhkiä jälkensä teke-

mällä riittävästi lokimerkinnän aiheuttavia tapahtumia. Lokien tulisi olla tallennettuna johonkin aina sillä tavalla, että vanhojakin lokimerkintöjä päästään vielä tarkastelemaan.

Tapahtumien jäljitettävyys vähintään kuuden kuukauden ajalta täyttää Katakri-vaatimukset, mutta sen lisäksi lokien käsittelyyn tulee olla jaettuna kirjallinen dokumentaatio (Puolustusministeriö 2015, 46). Katakri-vaatimuksissa riittävään jäljitettävyyden toteuttamiseen ehdotetaan keskitetylle lokien keräämiselle vaihtoehdoksi hyppykone-käytäntöä, jossa hallintatoimet tehdään ja kirjataan lokeihin hyppykoneen kautta (Puolustusministeriö 2015, 36). Hyppykone tarkoittaa erillistä palvelinta, jonka kautta etäyhteys otetaan hallittavaan palvelimeen.

5.3 Tunkeilijoiden ja haavoittuvuuksien havaitseminen

Tunkeilijanhavaitsemisjärjestelmän tehtävä on tarkkailla järjestelmää ja tunnistaa haitallinen toiminta. Tämä voi olla esimerkiksi palvelunestohyökkäys, porttiskannaus tai verkon yli tapahtuva tietomurtoyritys. Tunkeilijanhavaitsemisjärjestelmä tarkkailee reaaliaikaisesti lokitiedostoista ja muista mahdollisista tallenteista, ja varoittaa jos se havaitsee luvattoman käyttöyrityksen tai poikkeuksellisia yhteyksiä. Tunkeilijanhavaitsemisjärjestelmä auttaa täyttämään Katakri-vaatimuksen yleisiin verkkohyökkäyksiin varautumisesta (Puolustusministeriö 2015, 33).

Yksinkertaisimmillaan automattisia palomuurisääntöjä hallitseva fail2ban voi toimia tunkeilijanhavaitsemisjärjestelmänä, mutta tätä varten on hyvin monipuolisiakin työkaluja. Esimerkiksi tripwire ja aide ovat ohjelmistoja tunkeilijoiden havaitsemiseen ja monenlaisten hyökkäysten estämiseen. Näiden ohjelmistojen avulla voidaan luoda tarkistesummat tietokantaan järjestelmälle tärkeistä tiedostoista, jolloin tärkeiden konfiguraatiotiedostojen eheydestä ollaan tietoisia. Tunkeilijanhavaitsemisjärjestelmän voi konfiguroida lähettämään sähköpostin, kun tiedostojen eheyden tarkistuksen yhteydessä löydetään muuttuneita tarkistesummia. Tällöin muutoksia tärkeisiin järjestelmätiedostoihin ei siis pitäisi pystyä tekemään ilman, että järjestelmänvalvoja saa siitä tiedon.

6 JOHTOPÄÄTÖKSET JA POHDINTA

Keskeisintä Linux-palvelimen kovettamisessa on hahmottaa tietoturvan perusperiaatteita ja tietoturvassa huomioitavia osa-alueita. Päätutkimuskysymykseen työssä vastattiin selvittämällä keskeiset osa-alueet Linux-palvelimen tietoturvassa. Turvallisuuden osa-alueet koostuvat pääpiirteittäin asennukseen ja ohjelmistoihin liittyvistä huomioista, käyttöoikeuksista, tunnistautumisesta, verkkoturvallisuudesta ja valvonnasta. Järjestelmän turvallisuutta voidaan lisätä, kun kaikkiin osa-alueisiin sovelletaan vähimpien oikeuksien periaatetta ja pyritään pitämään riskitekijöiden määrä minimissä.

Pienimpien mahdollisten oikeuksien periaatteen lisäksi osa-alueisiin sovelletaan niihin liittyviä parhaita käytäntöjä. Yhtä osa-aluetta voi suojata useilla eri tavoilla, mutta varmempaa on suojata kaikilla mahdollisilla tavoilla. Hyvin suojattuakin järjestelmää tulee kohdella sillä ajatuksella, että tietoturva voi vaarantua milloin tahansa. Ongelmatilanteisiin varaudutaan muun muassa varmuuskopioiden, valvontajärjestelmän ja tunkeilijanhavaitsemisjärjestelmän avulla.

Turvallisuuteen liittyviä käsitteitä tarkasteltiin pitkälti yleisellä tasolla, eli lukijalle haluttiin luoda käsitystä Linux-palvelimen tietoturvaan liittyvien asioiden kokonaisuudesta. Toivon mukaan työstä voisi olla hyötyä aloittelevalle Linux-asiantuntijalle, jolla on jo kokemusta Linux-palvelimen ylläpidosta, mutta tietoturvan kokonaisuuteen voisi tuoda lisäarvoa. Työssä valittiin turvallisuuden osa-alueiden käsittelyyn hyvin yleinen taso. Työssä ei käsitelty asioiden yksityiskohtia, kuten komentoja, niin kuin ei ollut tarkoituskaan. Kokonaisuuden hahmottaminen Linux-järjestelmille vieraalle ihmiselle ei työn avulla välttämättä onnistu, sillä kovin yleismaailmalliselle tasolle tietoturvallisuuden osa-alueita työssä ei avattu.

Ennen kuin aloitin kirjoittamaan tätä opinnäytetyötä, koin olevani jossain määrin kokenut Linux-asiantuntija alan työkokemuksella. Opinnäytetyön kirjoittamisen aikana syvällinen perehtyminen Linux-järjestelmien toimintaa käsittelevään aineistoon opetti kuitenkin hahmottamaan enemmän järjestelmien turvallisuuteen liittyvien asioiden kokonaisuutta. Kokemus lisää itsevarmuutta, mutta hyvät käy-

tännöt mahdollisimman monessa osa-alueessa lisäävät turvallisuutta. Kokonaisuuden hahmottaminen on tärkeää, että on mahdollisuus ymmärtää, mistä asiasta täytyy ottaa vielä selvää.

Mielestäni työ onnistui keskeisten turvallisuuden osa-alueiden selvittämisessä, mutta kuten aina turvallisuuden kanssa, järjestelmä ei ole koskaan murtovarma, jos sen turvallisuutta ei aktiivisesti auditoida ja kehitetä. Koska työssä laadittiin yleisluontoinen katsaus Linux-palvelimen tietoturvan osa-alueisiin, työtä olisi loogista jatkaa syventymällä kuhunkin osa-alueeseen. Tämän voisi toteuttaa esimerkiksi esittelemällä aiheisiin liittyviä komentoja ja komentoparametreja. Linuxin tietoturvan käsittely sellaisessa laajuudessa olisi jo väitöskirjan aihe.

LÄHTEET

Kirjalliset lähteet:

Binnie, C. 2016. Linux Server Security: Hack and Defend. Indianapolis, IN: Wiley.

Bresnahan, C. & Blum, R. 2015. Linux Essentials. 2. painos. Indianapolis, IN: Sybex.

Monte, M. 2015. Network Attacks and Exploitation: A Framework. Indianapolis, IN: Wiley.

Negus, C. 2015. Linux Bible. 9. painos. Indianapolis, IN: Wiley.

Ward, B. 2015. How Linux Works: What Every Superuser Should Know. 2. painos. San Fransisco, CA: No starch press.

Sähköiset lähteet:

Boelen, M. 2018. Linux hardening steps for starters. Luettu 10.3.2019. https://linux-audit.com/linux-server-hardening-most-important-steps-to-secure-systems/

CentOS Wiki. OS Protection. Luettu 17.3.2019. https://wiki.centos.org/HowTos/OS_Protection

CentOS Wiki. Securing SSH. Luettu 31.3.2019 https://wiki.centos.org/HowTos/Network/SecuringSSH

Corbet, J. & Kroah-Hartman, G. 2017. Linux Kernel Development Report. The Linux Foundation. Luettu 3.5.2019. https://www.linuxfoundation.org/2017-linux-kernel-report-landing-page/

DeMuro, J. 2018. What is container technology. Techradar. Julkaistu 9.8.2018. Luettu 5.5.2019. https://www.techradar.com/news/what-is-container-technology

Docker Inc. What is a Container. n.d. Luettu 6.5.2019. https://www.docker.com/resources/what-container

Evans, C. 2014. Backup vs replication, snapshots, CDP in data protection strategy. Luettu 24.3.2019. https://www.computerweekly.com/feature/Backup-vs-replication-snapshots-CDP-in-data-protection-strategy

Fedora wiki. Disk Encryption User Guide. Luettu 14.4.2019. https://fedoraproject.org/wiki/Disk_Encryption_User_Guide

Fenton, J., Garcia M. & Grassi, P. 2017. Digital Identity Guidelines. NIST Special publication 800-63-3. Luettu 24.4.2019.
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf

Jaakkola, J. 2013. Yrityksen tietoturvaa – Application Whitelisting. Luettu 3.5.2019.
https://wiki.tut.fi/Tietoturva/Tutkielmat/ApplicationWhitelisting

Lehto, T. Tekniseltä tasoltaan kuin vuoden 2003 mato – WannaCry hyödyntää Windows-päivitysten puuttumista. Tekniikka ja talous. Lehtiartikkeli. Julkaistu 15.5.2017. Luettu 21.4.2019.
https://www.tekniikkatalous.fi/ttpaiva/tekniselta-tasoltaan-kuin-vuoden-2003-mato-wannacry-hyodyntaa-windows-paivitysten-puuttumista-6649306

Linux Databook. n.d. Linux Standards and Best Practices. Luettu 13.4.2019.
http://www.linux-databook.info/?page_id=2022

Linux Security Crash Course. 2018. Video. UTD CSG. Katsottu 2.5.2019
https://www.youtube.com/watch?v=i-noRUNh9Vk

Linux Wiki. Shadow-tiedosto. Julkaistu 6.3.2015. Luettu 5.3.2019.
https://www.linux.fi/wiki/Shadow-tiedosto

Pillai, S. 2013. Linux access control using TCP Wrappers. Luettu 5.5.2019
https://www.slashroot.in/linux-access-control-using-tcp-wrappers

Puolustusministeriö. 2015. Katakri – Tietoturvallisuuden auditointityökalu viranomaisille. Viitattu 4.4.2019.
https://www.defmin.fi/files/3165/Katakri_2015_Tietoturvallisuuden_auditointity-okalu_viranomaisille.pdf

Rousku, K. 2014. Miksi Shellshock on pahempi haavoittuvuus kuin Heartbleed. Tivi. Lehtiartikkeli. Julkaistu 29.9.2014. Luettu 25.4.2019.
https://www.tivi.fi/blogit/miksi-shellshock-on-pahempi-haavoittuvuus-kuin-heart-bleed/24018cb2-6795-36ac-ba06-d541597718ad

Solarwinds Pingdom. Exploring the Software Behind Facebook, the World’s Largest Social Media Site. Luettu 4.5.2019.
https://royal.pingdom.com/the-software-behind-facebook/

Van Vugt, S. 2016. Compare the best Linux distros for enterprise servers. Luettu 23.4.2019.
https://searchdatacenter.techtarget.com/feature/Compare-popular-Linux-distributions-for-servers

Virtanen, J. 2017. Näin toimii WannaCry-haittaohjelma – ”Uudet hyökkäykset ovat väistämättömiä". Tivi. Lehtiartikkeli. Julkaistu 15.5.2017. Luettu 21.4.2019.
https://www.tivi.fi/uutiset/nain-toimii-wannacry-haittaohjelma-uudet-hyokkaykset-ovat-vaistamattomia/9000bc54-1498-3ea1-9096-b219ba0a3de9