Toni Hämäläinen

IoT-verkon tietoturvauhat ja niiden estäminen

Tietotekniikan kandidaatintutkielma

10. toukokuuta 2020

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Toni Hämäläinen

Yhteystiedot: tomahama@student.jyu.fi

Ohjaaja: Timo Tiihonen

Työn nimi: IoT-verkon tietoturvauhat ja niiden estäminen

Title in English: Security threats of IoT-network and how to prevent them

Työ: Kandidaatintutkielma

Opintosuunta: Tietotekniikka

Sivumäärä: 29+0

Tiivistelmä: Tässä tutkielmassa tutustutaan IoT-verkkoon kohdistuviin hyökkäyksiin, sekä tapoihin joilla suojautua niiltä. Tutkielman tavoitteena on kartoittaa tapoja joilla IoT-verkko voidaan toteuttaa turvallisesti tietoturvan ja yksityisyyden suojan kannalta. Aihetta tarkastellaan IoT:n kerrosarkkitehtuurin näkökulmasta. Tutkielmassa kerrotaan tiettyihin kerrosarkkitehtuurin kerroksiin kohdistuvista hyökkäyksista ja suojastavoista. Tämän lisäksi tutkielmassa pohditaan, kuinka IoT-verkko voidaan toteuttaa mahdollisimman turvalliseksi tietoturvan kannalta. Tämä vaatii toimia niin laitevalmistajalta, kuin asiakkaalta. Näitä toimia ovat esimerkiksi ohjelmistopäivistysten ajankohtainen asentaminen, sekä erillisten turvallisuusmoduulien käyttäminen verkossa.

Avainsanat: kandidaatintutkielma, IoT, Tietoturva, Yksityisyyden suoja, Kerrosarkkitehtuuri, IoT-elementit, IoT-hyökkäykset, Esineiden internet, Internet of Things, Esineiden internetin suojaaminen

Abstract: This thesis focuses on researching different kind of security threats on internet of things. The goal of this research is to find ways to prevent these threats in order to create a safe IoT-network from the perspective of security and privacy of the user. The subject is viewed from the perspective of layered architecture of IoT. This thesis brings out the specific attacks and defense methods that target these specific layers in layered architecture of IoT. Besides this, the thesis discusses how IoT-network can be implemented safely from

the perspective of security. This requires actions from the product manufacturer and from the client. These actions are quickly installing software updates and using separate safetymodule in the IoT-network.

Keywords: IoT, security, privacy policy, layered architecture, IoT-elements, IoT-attacks, Internet of Things, security of IoT

Termiluettelo

DoS-hyökkäys	palvelunestohyökkäys, jolla estetään jonkin verkkosivun tai laitteen toiminta syöttämällä sille paljon palvelupyyntöjä.
SCA-hyökkäys	side channel attack, hyökkäys jossa laitteeseen tehdään fyysisiä muutoksia, tai mitataan sen fyysisiä ominaisuuksia ja päätellään datan sisältöä niiden avulla.
Hunajapurkki	ansa, jonka tarkoituksena on havaita tietojen käyttäminen ilman käyttöoikeutta tai autorisaatiota.

Kuviot

Kuvio 1. IoT-verkon yleisin rakenne tutkijoiden Han, Jeon ja Kim (2015) mukaan 4
Kuvio 2. IoT-Elementit (Burhan ym. 2018) 5
Kuvio 3. Tutkijoiden Sethi ja Sarangi (2017) luoma kerrosarkkitehtuuri ja perinteinen kolmen kerroksen arkkitehtuuri 8

Sisältö

1 JOHDANTO 1

2 TIETOTURVA ESINEIDEN INTERNETISSÄ 3

3 IOT-VERKON TEKNINEN JA LOOGINEN ARKKITEHTUURI 4
3.1 IoT-elementit 4
3.2 IoT:n kerrosarkkitehtuuri 6
3.2.1 Kolmen kerroksen arkkitehtuuri 6
3.2.2 Neljän kerroksen arkkitehtuuri 7
3.2.3 Viiden kerroksen arkkitehtuuri 7

4 IOT-VERKKOIHIN KOHDISTUVAT HYÖKKÄYKSET 9
4.1 Yleisimmät hyökkäykset IoT-verkkoihin 9
4.2 Havainnointikerrokseen kohdistuvat hyökkäykset 9
4.3 Kuljetuskerrokseen kohdistuvat hyökkäykset 10
4.4 Sovelluskerrokseen kohdistuvat hyökkäykset 11
4.5 Liiketoimintakerrokseen kohdistuvat hyökkäykset 11
4.6 Side channel attack (SCA) 11

5 IOT-VERKON SUOJAAMINEN 14
5.1 Turvallisuusmoduulien lisääminen IoT-verkkoon 14
5.2 Verkossa kulkevan datan salaaminen 16
5.3 Salausavaimen sopiminen IoT-ympäristössä 16

6 IOT-VERKON TURVALLINEN TOTEUTTAMINEN 18

LÄHTEET 20

1 Johdanto

Termillä esineiden internet (Internet of Things, IoT) on useita määritelmiä. Yleisellä tasolla IoT:llä tarkoitetaan erilaisten sensoreiden, laitteiden ja aktuaattoreiden, kuten moottoreiden yhdistämistä internettiin. IoT-laitteet ovat vuorovaikutuksessa fyysisen maailman kanssa, ne keräävät siitä tietoa ja lähettävät tämän datan toisille laitteille tai käsittelevät tiedon itse. Esineiden internettiä käytetään jo laajasti muunmuassa teollisuudessa, kaupungeissa, kuluttajatuotteissa, sekä maataloudessa. Teollisuudessa sitä hyödynnetään esimerkiksi logistiikassa, maataloudessa sen avulla valvotaan maaperän laatua ja kotitalouksissa IoT-teknologiaa löytyy niin älytelevisioista kuin automatisoidusta ilmastoinnista. IoT:n vahvuus on sen itsenäisyys, järjestelmä ei tarvitse ihmistä hallinnoimaan toimintaa lainkaan.

IoT on kasvava teknologia ja sen käytön arvioidaan laajenevan jatkossakin. Tutkimusyritys IDC (2019) on ennustanut IoT:n tuottaman datan määrän kasvavan 79,4 ZB:iin (tsettatavuun) ja verkkoon yhdistettyjen laitteiden määrän 41,6 miljardiin vuoteen 2025 mennessä. Tulevaisuudessa voidaan todennäköisesti puhua älykaupungeista, kun viitataan kaupunkeihin, jotka hyödyntävät IoT-sensoreita (Rhee 2016). IoT-laitteiden yleistyessä on tärkeää huomioida tietoturva, yksityisyyden suoja, sekä IoT-teknologian tuomat haasteet. Huonosti toteutettuna heikko tietoturva voi johtaa yksityisyyden suojan vaarantumiseen. Tämä tarkoittaa, että ulkopuolisilla ihmisillä on mahdollisuus päästä käsiksi käyttäjän henkilökohtaisiin tietoihin. Mikäli IoT-verkon tietoihin päästään käsiksi, ulkopuoliset henkilöt voivat lukea ja muunnella IoT-laitteiden lähettämää dataa (Pammu ym. 2016). Hyvä tietoturva ja yksityisyyden suoja ovat siis erityisen tärkeitä, sillä IoT kerää jatkuvasti tietoa tuotetta käyttävästä kuluttajasta, sekä IoT-laitteen ympäristöstä (Peng, Pal ja Huang 2019). Terveydenhuollossa yksityisyyden suoja korostuu, IoT-järjestelmän tulee varmistaa, että vain tietyillä henkilöillä on pääsy tiettyihin tietoihin joita henkilöstä kerätään (Sfar ym. 2018).

Tässä tutkielmassa tehdään kirjallisuuskatsaus IoT-laitteiden tietoturvaan ja sen toteuttamisen haasteisiin. Haasteet joihin tässä tutkielmassa tutustutaan ovat tietoturvaongelmat, kuten hyökkäykset IoT-laitteisiin ja IoT-verkkoon. Tutkielma toteutetaan tekemällä kirjallisuuskartoitus. Tämän avulla selvitetään, millaisia tietoturvariskejä tutkijat ovat löytäneet IoT-laitteista ja IoT-verkosta, sekä millaisia ratkaisuja näihin ongelmiin on keksitty.

Seuraavassa luvussa käydään läpi käsitteet tietoturva ja yksityisyyden suoja, sekä perustellaan, miksi ne ovat tutkielman kannalta keskeisiä aiheita. Luvussa IoT-verkon tekninen ja looginen arkkitehtuuri, kuvataan IoT-verkon rakennetta, mistä osa-alueista verkko koostuu, sekä millaisia teknologioita IoT-arkkitehtuurin eri osissa hyödynnetään. Lisäksi luvussa havainnollistetaan erilaisten arkkitehtuurien tuomia etuja verrattuna aikaisempiin kerrosarkkitehtuureihin. Luvussa neljä kerrotaan IoT-verkon eri osa-alueisiin kohdistuvista hyökkäyksistä, sekä niiden toteutustavoista ja päämääristä. Tämän jälkeen luvussa viisi pyritään vastaamaan näihin hyökkäystapoihin erilaisilla suojausmekanismeilla. Nämä suojausmekanismit jaotellaan osa-alueisiin sen mukaan, mitä kerrosarkkitehtuurin osa-aluetta niillä yritetään suojella. Kuudennessa luvussa tehdään yhteenveto siitä, kuinka IoT-verkko voitaisiin toteuttaa turvallisesti niiden uhkien osalta, jotka tässä tutkimuksessa on esitetty.

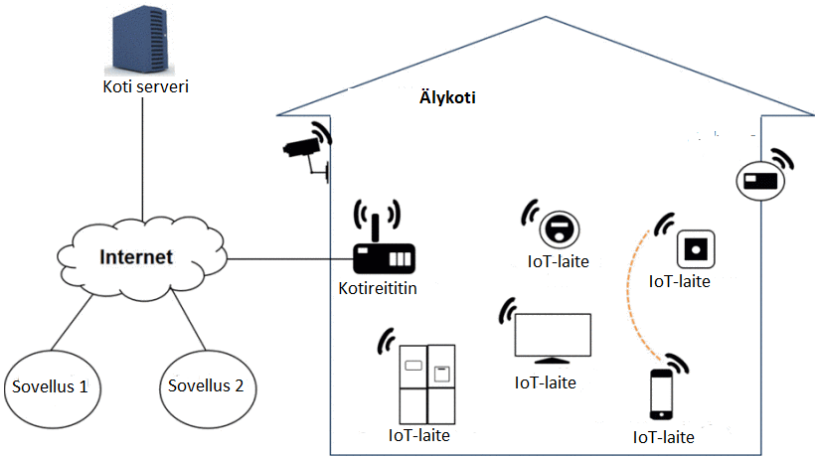
2 Tietoturva esineiden internetissä

Tietoturva ja yksityisyyden suoja ovat IoT:n kannalta oleellisia asioita siitä syystä, että kyseistä teknologiaa käytetään ihmisten päivittäisessä arjessa. Erityisesti terveydenhuollossa käytettävissä teknisissä ratkaisuissa yksityisyyden suoja ja tietoturva korostuvat, sillä potilaasta kerätään jatkuvasti dataa. Tietoturva koostuu luottamuksellisuudesta, eheydestä ja käytettävyydestä. Eheys tarkoittaa sitä, että tietoja ei voida muuttaa ulkopuolisen toimesta. Käytettävyys tarkoittaa sitä, että tietoja on mahdollista hyödyntää, jos käyttäjällä on niihin käyttöoikeus. Luottamuksellisuus tarkoittaa sitä, että tiedot ovat vain niiden henkilöiden käytettävissä, joilla on kyseiseen tietoon oikeus päästä käsiksi. (Kyberturvallisuuskeskus, 2019)

Yksityisyyden suoja määritellään Suomen perustuslaissa niin, että ihmisen yksityiselämää koskevaa tietoa ei saa saattaa lukuisten ihmisten saataville, jos tiedosta voi aiheutua henkilölle vahinkoa. Tietoturva ja yksityisyyden suoja ovat keskeisiä aiheita kun suunnitellaan turvallista IoT-verkkoa, sillä IoT-laitteet keräävät paljon tietoa käyttäjästä ja ympäristöstä. Mikäli näitä tietoja ei suojata riittävän hyvin, on ulkopuolisten uhkatekijöiden mahdollista päästä käsiksi suureen määrään dataa.

3 IoT-verkon tekninen ja looginen arkkitehtuuri

IoT-verkko koostuu usein IoT-laitteista, kuten sensoreista tai kodinkoneista, sensorin ja pilven välisestä portista (engl. gateway), sekä IoT-palveluntarjoajan serveristä tai pilvipalvelusta. Tämän lisäksi palveluntarjoaja usein tarjoaa käyttöliittymän tai sovelluksen, jolla IoT-verkkoa voidaan ohjata ja valvoa pilven välityksellä. Kaikissa verkoissa ei kuitenkaan ole välttämätöntä olla porttia, tai reititintä pilven tai palvelimen ja IoT-laitteen välillä. Han, Jeon ja Kim (2015) kuvaavat IoT-verkon kotitalouksissa seuraavalla tavalla:



Kuvio 1. IoT-verkon yleisin rakenne tutkijoiden Han, Jeon ja Kim (2015) mukaan

IoT-verkon perusrakenne on teollisuudessa sama, mutta sensoreita ja muita IoT-laitteita on enemmän. IoT-verkkoon kohdistuvat hyökkäykset voidaan kohdistaa IoT-laitteeseen, porttiin, serveriin, tai edellä mainittujen laitteiden väliseen yhteyteen. Hyökkäys voidaan kohdistaa myös IoT-laitteen palveluntarjoajan luomaan sovellukseen, mutta näitä hyökkäyksiä ei tutkita tässä tutkielmassa.

3.1 IoT-elementit

Burhan ym. (2018) rakentavat kokonaiskuvan IoT-verkon osa-alueista määrittelemällä IoT-elementit. Nämä elementit antavat kattavan kokonaiskuvan IoT-verkon osa-alueista ja niissä

käytetyistä teknologioista. Burhan ym. (2018) esittelevät IoT-elementit ja jakavat ne kuuteen osaan: Identifiointi, mittaus, kommunikointi, laskenta, palvelut, semantiikka.

Kuvio 2. IoT-Elementit (Burhan ym. 2018)

Identifioinnilla tarkoitetaan laitteen identiteettiä verkon sisällä. Identifioinnilla on kaksi osaa, nimeäminen ja osoitteen antaminen. Nimeämisellä tarkoitetaan yksinkertaisesti sitä, että laitteelle annetaan jokin nimi. Osoitteen antaminen on yksilöllinen jokaiselle laitteelle. Kahdella laitteella on mahdollista olla sama nimi, mutta osoitteet ovat aina erilaiset. Nimeämisessä käytetään esimerkiksi tuotenumeroa ja osoitteenmäärittelyssä Ipv6 teknologiaa.

Mittauksella tarkoitetaan informaation keräämistä erilaisilla sensoreilla. Tämä kerätty data lähetetään eteenpäin.

Kommunikoinnilla on keskeinen rooli IoT-teknologiassa, sillä laitteiden tulee pystyä kommunikoimaan keskenään ja lähettämään, sekä vastaanottamaan erilaista dataa. Kommunikoiminen on myös tietoturvan kannalta keskeisin elementti. Kommunikoiminen vastaa tiedon salaamisesta, kuljetusprotokollasta, sekä kommunikoimisen turvallisuudesta. Myös suurin osa hyökkäyksistä kohdistuu tähän elementtiin. IoT-laitteet tyypillisesti kommunikoivat langattomasti, mihin on tarjolla useita eri tapoja, kuten RFID (Radio Frequency Identification), NFC (Near Field Communication), Bluetooth, ja WiFi. IoT-verkot voivat myös kommunikoida muilla teknologioilla, mutta tässä kappaleessa mainitut teknologiat ovat yleisimpiä.

Laskennassa käsitellään sensoreiden keräämää dataa. Sensorit eivät itse välttämättä laske datasta mitään, vaan tässä voidaan hyödyntää esimerkiksi Rasperry Pi:tä tai Arduinoa. Laskennassa on mahdollista myös käyttää laitteiston lisäksi ohjelmistoa, esimerkiksi Androidia, tai muuta käyttöjärjestelmää.

Palveluita on neljä erilaista tyyppiä: identifiointi, tiedon kerääminen ja käsittely, toiminnan aktivointi kerätyn datan perusteella, sekä saatavuus. Saatavuudella tarkoitetaan että laitteelle joka haluaa kommunikoida, vastataan välittömästi riippumatta ajankohdasta.

Semantiikan tarkoitus on helpottaa käyttäjän elämää suorittamalla IoT-laitteistolle tarkoitettuja tehtäviä. Semantiikka kerää kaiken informaation, tekee päätöksiä kerätyn informaation perusteella ja vastaa verkon laitteille. Semantiikassa voidaan hyödyntää RDF:ää (Resource Description Framework) tai WOL:ia (Web Ontology Language). (Burhan ym. 2018)

3.2 IoT:n kerrosarkkitehtuuri

Tutkijat ovat jäsentäneet IoT-järjestelmiä erilaisten arkkitehtuurien avulla, joista yleisimpiä ovat kolmen kerroksen, neljän kerroksen, ja viiden kerroksen arkkitehtuuri. Neljän ja viiden kerroksen arkkitehtuuri sisältävät samat kerrokset kuin kolmen kerroksen arkkitehtuuri. Näihin useamman kerroksen arkkitehtuureihin on lisätty kerroksia IoT:n kehittyessä ja kasvaessa, jotta arkkitehtuuri voisi vastata kaikkiin vaadittuihin turvallisuustoimenpiteisiin. (Burhan ym. 2018)

3.2.1 Kolmen kerroksen arkkitehtuuri

Kolmen kerroksen arkkitehtuuri rakentuu havainnointikerroksesta, kuljetuskerroksesta ja sovelluskerroksesta. Havainnointikerros koostuu sensoreista ja laitteista. Se vastaa myös laitteiden identifioinnista. Kuljetuskerros siirtää kerätyn datan ja sisältää ohjelmistoja ja laitteistoja. Sovelluskerros määrittelee kaikki käyttötavat johon IoT teknologiaa käytetään. Se tarjoaa käyttökohteeseen sopivat palvelut. (Said ja Masud 2013)

Havainnointikerrokseen kohdistuvia hyökkäyksiä ovat tutkijoiden Burhan ym. (2018) mukaan laitteen salakuuntelu (engl. eavesdropping), laitteen kaappaaminen, oman laitteen li-

sääminen verkkoon, sekä ajoitushyökkäys. Salakuuntelu tarkoittaa sitä, että ulkopuolinen henkilö kuuntelee laitteen kommunikointia ja yrittää varastaa tietoa jota kuljetetaan verkossa. Salakuuntelu hyödyntää huonosti suojattua datan siirtoa. Kuljetuskerrokseen kohdistuvia hyökkäyksiä ovat palvelunestohyökkäys, man-in-the-middle hyökkäys, hyökkäys tietovarastoon, järjestelmän heikkouksia hyödyntävä hyökkäys (engl. exploit attack) (Burhan ym. 2018). Näistä hyökkäyksistä kerrotaan lisää myöhemmissä kappaleissa. Sovelluskerrokseen voidaan hyökätä injektiohyökkäyksellä, haitallisella koodilla, tai massadatan syöttämisellä laitteelle. Kun laitteelle syötetään liikaa dataa, se ei kykene käsittelemään kaikkea informaatiota ja täten häiritsee verkkoa, sekä hukkaa dataa (Burhan ym. 2018).

3.2.2 Neljän kerroksen arkkitehtuuri

IoT-teknologia on kasvanut ja laajentunut suuresti viimeisen vuosikymmenen aikana, joten kolmen kerroksen arkkitehtuuri ei kykene enää vastaamaan kaikkiin tietoturvauhkiin jotka kohdistuvat IoT-verkkoon. Tästä syystä tutkijat ovat kehittäneet uudempia arkkitehtuureja, joiden uudet kerrokset pyrkivät korjaamaan näitä ongelmia. Darwish (2015) esittelee arkkitehtuurin, jossa sovellus- ja kuljetuskerroksen väliin luotaisiin palvelu- ja sovellustukikerros. Muut kerrokset toimivat samalla tavalla kuin edellä esitetyssä arkkitehtuurissa, mutta tukikerros vastaa IoT:n turvallisuudesta (Burhan ym. 2018). Tässä arkkitehtuurissa sensoreiden keräämä data lähetetään tukikerrokselle, tukikerros varmistaa että informaatio on tullut autentikoidulta käyttäjältä ja että se on suojattu uhilta. Tiedon oikeellisuuden ja käyttäjän autentikoimiseksi on monia keinoja ja näihin tutustutaan myöhemmissä kappaleissa. Tämän lisäksi tukikerros lähettää dataa kuljetuskerrokselle. Tukikerrokseen kohdistuvia hyökkäyksiä ovat palvelunestohyökkäykset (Burhan ym. 2018).

3.2.3 Viiden kerroksen arkkitehtuuri

Sethi ja Sarangi (2017) ovat ehdottaneet viisikerroksista arkkitehtuuria, jossa havainnointi- ja kuljetuskerroksen jälkeen on prosessointikerros, sovelluskerros ja liiketoimintakerros.
Sethi ja Sarangi (2017) kuvaavat prosessointikerroksen, tai välilaitekerroksen (engl. middleware layer) varastoivan, analysoivan ja prosessoivan kuljetuskerrokselta tulevan datan. Se tarjoaa ja hallitsee palveluita alemmille kerroksille. Se voi hyödyntää tietokantoja, pilvilas-

Kuvio 3. Tutkijoiden Sethi ja Sarangi (2017) luoma kerrosarkkitehtuuri ja perinteinen kolmen kerroksen arkkitehtuuri

kentaa ja massadatan prosessointimoduuleja. Liiketoimintakerros hallitsee koko IoT-järjestelmää, sisältäen sovelluskohteet, liiketoimintamallin ja käyttäjän yksityisyyden. Prosessointikerrokseen on mahdollista hyökätä haittaohjelmilla jotka vaarantavat käyttäjän tietojen luottamuksellisuuden (Burhan ym. 2018). Liiketoimintakerroksessa esiintyviä vikoja ovat ohjelmoijan tekemät virheet. Näitä ovat heikko salasanan palautusjärjestelmä ja web-sovelluksen ohjelmoinnissa tehty salaustekniikan virheellinen toteutus, sekä syötetyn tiedon huono tarkistamisen. Näiden hyökkäyksien avulla on mahdollista varastaa dataa ja murtaa verkon salaus. (Rouse, 2020).

4 IoT-verkkoihin kohdistuvat hyökkäykset

IoT-laitteisiin ja verkkoihin kohdistuu suuri määrä erilaisia hyökkäyksiä. Hyökkäykset kohdistuvat myös verkon eri osiin. Tässä tutkielmassa esitetty kerrosarkkitehtuuri on hyvä tapa havainnollistaa hyökkäysten jakautuminen verkon eri osa-alueisiin. Tästä syystä hyökkäykset on jaoteltu niiden kohdekerroksen mukaan alaluvuiksi. Tukikerrokseen ei syvennytä, sillä tukikerros pyrkii vastaamaan samoihin tietoturvauhkiin kuin havainnointi- ja kuljetuskerros.

4.1 Yleisimmät hyökkäykset IoT-verkkoihin

F-Secure (2019) julkaisi raportin jossa yhtiö kertoi IoT-laitteisiin kohdistuneiden hyökkäysten kasvaneen rajusti ensimmäisen vuosipuoliskon aikana. F-Secure on tarkoituksellisesti luonut niin sanottuja hunajapurkkeja (engl. honeypot), joiden tarkoituksena on kerätä tietoa kyberhyökkääjien hyökkäystavoista ja kohteenvalintaprosessista. Vuonna 2019 tammikuun ja kesäkuun välisenä aikana IoT-laitteisiin kohdistuneita hyökkäyksiä oli kertynyt 2.9 miljardia. Vuonna 2018 hyökkäyksiä oli kertynyt kaiken kaikkiaan hieman yli yksi miljardi. Hyökkäysten määrä on siis selvästi noussut hyvin lyhyessä ajassa. Jos katsotaan F-Securen asiakkaisiin kohdistuvia hyökkäyksiä, yleisiä haittaohjelmia ovat kiristysohjelmat, troijalaiset ja kryptomainaajat. F-Secure ei ole ainoa yritys, joka on havainnut tämän kaltaisia tuloksia. Kaspersky (2019), joka myös toimii tietoverkkoturvallisuuden alalla, raportoi 105 miljoonasta hyökkäyksestä heidän noin viiteenkymmeneen hunajapurkkiin ensimmäisen vuosipuoliskon aikana vuonna 2019.

4.2 Havainnointikerrokseen kohdistuvat hyökkäykset

Burhan ym. (2018) mainitsevat havainnointikerrokseen kohdistuvia hyökkäyksiä olevan muun muassa laitteen salakuuntelu, laitteen kaappaaminen, sekä valelaitteen lisääminen verkkoon. Tämän lisäksi laitteisiin kohdistuvia hyökkäyksiä ovat sivukanavan hyökkäykset (engl. Side channel attack). Salakuuntelussa laitteen kommunikointia seuraa ulkopuolinen tekijä, joka haluaa varastaa verkossa kulkevaa informaatiota (Burhan ym. 2018). Salakuuntelu kohdistuu niin havainnointi- kuin kuljetuskerrokseen.

Laitteen kaappaamisessa puhutaan hyökkääjän täydestä laitteen hallinnasta. Tässä tapauksessa laitteen kaappaajan on mahdollista suorittaa verkossa erilaisia toimintoja ja hän kykenee vaarantamaan koko verkon (Bharathi ym. 2012). Valelaitteen lisääminen verkkoon mahdollistaa valheellisen datan syöttämisen. Hyökkäyksen tarkoituksena on pysäyttää oikean datan kulku kuluttamalla verkon tiedonkäsittelykapasiteettia. (Burhan ym. 2018)

4.3 Kuljetuskerrokseen kohdistuvat hyökkäykset

Kuljetuskerroksen tehtävänä on toimittaa havainnointikerroksen lähettämä data sovelluskerrokselle. Kuljetusteknologia voi olla langallinen, tai langaton. Kuljetuskerros vastaa myös laitteiden yhdistämisestä toisiinsa. Tästä johtuen kuljetuskerros on herkkä hyökkäyksille. Yleisiä hyökkäyksiä tähän kerrokseen ovat palvelunestohyökkäykset, man in the middle-hyökkäykset, sekä järjestelmän heikkouksia hyödyntävät hyökkäykset (engl. exploit attack). (Burhan ym. 2018) Palvelunestohyökkäyksen tarkoituksena on estää oikeita käyttäjiä käyttämästä laitetta, tai sen resursseja. Tämä tapahtuu usein tukehduttamalla laite liian suurella määrällä palvelupyyntöjä.

Man in the middle-hyökkäyksissä hyökkääjä keskeyttää kommunikoinnin ja muuntaa dataa joka liikkuu lähettäjän ja vastaanottajan välillä. Laitteilla ei ole tästä muuntelusta tietoa ja hyökkääjä voi muuttaa dataa tahtonsa mukaan. Tämä synnyttää uhkan turvallisuudelle, sillä datan oikeellisuudesta ei ole mitään varmuutta. (Conti, Dragoni ja Lesyk 2016) Järjestelmän heikkouksia hyödyntävät hyökkäykset koostuvat turvallisuuspuutteisiin iskevistä hyökkäyksistä. Näitä turvallisuuspuutteita voi löytyä käyttöjärjestelmästä, käyttäjälle tarkoitetusta sovelluksesta, tai kirjastoista joita ohjelmistossa käytetään. (Rouse)

Raportissaan F-Secure (2019) kertoo valtaosan hyökkäyksistä kohdistuneen TCP-portteihin. IoT-laitteet käyttävät yhä kuljetuskerroksessa Telnet protokollaa. Telnet protokollasta on siirrytty muissa laitteissa pois, sillä se ei ole turvallinen.

4.4 Sovelluskerrokseen kohdistuvat hyökkäykset

Burhan ym. (2018) mainitsevat injektiohyökkäykset, sekä vaarallisen koodin syöttämisen. Tämän lisäksi on mahdollista hyödyntää järjestelmän rajallista kykyä käsitellä suuria määriä dataa. Kun verkkoa rasitetaan ylimääräisellä liikenteellä, on mahdollista aiheuttaa häiriöitä verkossa, sekä datan häviämistä (Burhan ym. 2018). F-Securen (2019) raportin mukaan hunajapurkeista löydetyistä haittaohjelmista Mirai oli yleisin. Tämä on erikoista, sillä Mirai on ollut olemassa jo vuodesta 2016. Mirai-haittaohjelma hyökkää IoT-laitteisiin ja pakottaa nämä osaksi DDoS-hyökkäystä. Sovelluskerroksen suojaamiseen on ehdotettu useita protokollia, näihin tutustutaan tulevissa kappaleissa.

4.5 Liiketoimintakerrokseen kohdistuvat hyökkäykset

Rouse kuvailee liiketoimintakerrokseen kohdistuvien hyökkäyksien hyödyntävän ohjelmointivirheitä. Tämän lisäksi hyökkäyksessä voidaan hyödyntää heikkoa salasanan palautusjärjestelmää tai syöttötiedon heikkoa tarkistamista. Esimerkkinä tälläisesta virheestä voidaan käyttää Heartbleed turvallisuusbugia. Tämän lisäksi hyökkääjä voi tutkia ohjelmaa ja löytää uusia tietoturva-aukkoja joita ohjelman kehittäjä ei ole huomannut (Bilge ja Dumitraundefined 2012).

4.6 Side channel attack (SCA)

Side channel attack tarkoittaa fyysistä hyökkäystä laitetta kohtaan. Laite voidaan fyysisesti kaapata ja siihen voidaan tehdä muutoksia, joiden avulla laitteen tuottamaa dataa voidaan tutkia. Hyökkäyksen kohteena voi olla IoT-verkon sensori, tai reititin. Nämä hyökkäykset keskittyvät monesti mittaamaan laitteen virrankulutusta (Kocher ym. 2011), mutta hyökkäykset voivat myös mitata laitteen elektromagneettista säteilyä (Hori ym. 2012), tai laitteen kellon ajoitusta (Coppens ym. 2009). Laitteen virrankulutusta mittaavat hyökkäykset vaativat pääsyn laitteeseen, sekä muutosten tekemistä laitteeseen jotta virrankulutusta voidaan luotettavasti mitata (Pammu ym. 2016). Tämä voi vaikeuttaa hyökkäyksen implementointia. SCA-hyökkäyksiin on keksitty vastatoimia, jotka kuitenkin vaativat laitetason muutoksia, tai ohjelmisto muutoksia (Brumley ja Boneh 2003).

CEMA (Correlation electromagnetic analysis) on yleinen elektromagneettisen säteilyn mittaamiseen perustuva hyökkäystapa. Koska se mittaa laitteen tuottamaa elektromagneettista säteilyä, laitteeseen ei tarvitse fyysisesti koskea, tai tehdä muutoksia, toisin kuin mitattaessa virrankulutusta (Balasch ym. 2015). CEMA:n tarkoituksena on paljastaa salausavain joka on tuotettu AES-suojauksella.(Pammu ym. 2016) Tutkijat ovat onnistuneet purkamaan AES-suojauksen suorittamalla 20 000 elektromagneettista mittausta laitteen staattisesta RAM-muistista (Pammu ym. 2016).
Vasteaikahyökkäys (engl. timing) tarkkailee järjestelmän viivettä kun järjestelmälle lähettää palvelupyyntöjä tai muita kyselyitä. Vasteaikahyökkäykset kohdistetaan usein heikon laskentatehon laitteisiin, joihin IoT-laitteet myös kuuluvat. (Brumley ja Boneh 2003). Analysoimalla laitteen vastausviivettä, on mahdollista saada laitteen käsittelemästä datasta osia, tai jopa salausavain haltuun. Näin kävi esimerkiksi tutkimuksessa, jossa vasteajan avulla purettiin RSA-salaus (Kocher ym. 2011).

RSA-salaus on julkisen avaimen salausalgoritmi, joka perustuu julkiseen ja yksityiseen avaimeen. Yksityistä avainta ei voida nykytekniikalla johtaa julkisesta avaimesta. Julkisen avaimen avulla luodaan salattuja viestejä jotka voidaan lukea yksityisen avaimen avulla. Salaus toimii myös toisinpäin, yksityisellä avaimella voidaan luoda viestejä, jotka voidaan avata ainoastaan julkisella avaimella. Tässä tapauksessa viesti voidaan "allekirjoittaa"lähettäjän omalla yksityisellä avaimella. Tämä varmistaa että viesti on peräisin alkuperäiseltä lähettäjältä. (Wikipedia, 2020). Vaikka RSA-salausalgoritmin purkaminen laskemalla ei nykyteknologialla ole mahdollista, avain itsessään on mahdollista saada haltuun laitteen vasteaikaa kuuntelemalla (Kocher ym. 2011). Vasteaikahyökkäys on kuitenkin mahdollista estää suorittamalla salauksen aikana satunnaisia laskutoimituksia, jotka muuttavat tietokoneen vaatimaa aikaa laskennalle jokaisella salausoperaatiolla ja salauksen purkamisella. (Arjunan, Narayanan ja Ramu 2016) IoT-laitteissa tämä kuitenkin tarkoittaa ylimääräisten laskutoimien suorittamista, joka ei välttämättä ole sopiva ratkaisu laitteen alhaisesta laskentatehosta ja pienestä muistista johtuen.
Kuten aiemmin todettiin, virtapiirin virrankulutusta seuraamalla on mahdollista saada haltuun virtapiirin käsittelemää dataa. DPA (engl. Differential power analysis) keskittyy purkamaan algoritmin tuottaman salauksen laitteen virrankulutuksen kautta. Tämä tarkoittaa sitä, että vaikka algoritmi ja suojausprotokolla ovat vahvoja, ne voidaan purkaa tutkimalla laitteen

fyysisiä ominaisuuksia.

5 IoT-verkon suojaaminen

IoT-verkon suojaamiseen on useita tapoja. Verkon jokaisen laitteen tietoturvaa voidaan parantaa kehittämällä salausalgoritmeja, vahvoja tunnistautumisprotokollia, sekä toteuttamalla laitteisiin mekanismi, joka varoittaa jos laitteeseen tehdään fyysisiä muutoksia tai jos laitteeseen kohdistuu hyökkäyksiä. Tämän lisäksi on mahdollista parantaa IoT-verkon tietoturvaa lisäämällä siihen erikseen erilaisia turvallisuusmoduuleja. Näitä moduuleja voidaan sijoittaa reitittimeen, sensoreihin, tai näiden välille.

5.1 Turvallisuusmoduulien lisääminen IoT-verkkoon

Tutkijat Simpson, Roesner ja Kohno (2017) ehdottavat tutkimuksessaan välilaitteen lisäämistä reitittimen ja sensoreiden välille niin, että kaikki tietoliikenne kulkee tämän turvallisuusmoduulin läpi. Tämä moduuli mahdollistaisi ohjelmistopäivityksien turvallisen asentamisen, tietoliikenteen suodattamisen, sekä vahvemman tunnistautumisen niin alkuperäisille verkon laitteille, kuin uusille laitteille, jotka voidaan lisätä verkkoon myöhemmin.

Toinen lähestymistapa aiheeseen on tutkijoiden Yu ym. (2015) kehittämä välimoduulimalli, jossa yhden keskitetyn turvallisuusmoduulin sijaan käytetään moduulia jokaisen sensorin ja reitittimen välillä. Nämä yksittäiset välimoduulit voitaisiin kustomoida jokaisen verkon laitteen tarpeiden mukaiseksi. Useiden välimoduulien avulla on helppoa paikata yksittäisten laitteiden tietoturvaongelmia. Tästä esimerkkinä toimii tutkijoiden Yu ym. (2015) toteuttama prototyyppi valvontakamerasta, serveristä ja välimoduulista. Valvontakameran käyttöliittymä ei tarjonnut mahdollisuutta muuttaa laitteen kirjautumistietoja, joka tarkoittaa, että laitteeseen pääsee käsiksi oletussalasanalla. Hyökkääjän on helppo päästä kameraan käsiksi, sillä oletussalasanoja käyttävät hyökkäykset ovat yleisiä (F-Secure 2019). Välimoduuli mahdollistaa kuitenkin uuden salasanan luomisen. Tämän jälkeen oletussalasanan tietäminen ei enää riitä, sillä kameran hallintajärjestelmä on turvallisuusmoduulin salasanan takana.

Yleisen turvallisuusmoduulin tarkoituksena on pyrkiä estämään erilaisia haavoittuvuuksia. Näitä ovat ohjelmistopäivityksien tarkistaminen, lataaminen ja asentaminen, sekä tietoliikenteen valvominen ja suodattaminen. Tämän lisäksi moduulin on mahdollista estää MitM

(Man in the middle) hyökkäykset ja havaita laitteen vaarantuminen. Ohjelmistopäivityksien välitön lataaminen on tärkeää, sillä se minimoi hyökkääjän aikaikkunan toteuttaa hyökkäys päivittämättömään tietoturva-aukkoon. Turvallisuusmoduuli myös valvoo, milloin päivitys asennetaan, esimerkiksi television ohjelmistopäivitys toteutettaisiin katseluajan ulkopuolella.
Tietoliikenteen suodattamisella estetään vaarallisen liikenteen tapahtuminen laitteelle. Tämä estää matojen lataamisen ja järjestelmän heikkouksia hyödyntävien hyökkäysten tapahtumisen. MitM-hyökkäys estetään niin, että moduuli toteuttaa yhteyden ja varmistaa että sitä ei ole keskeytetty. Tämän lisäksi moduuli vahvistaa serverin sertifikaatit avainten vaihdon yhteydessä. Tämä mahdollistaa turvallisen kommunikoinnin. Turvallisuusmoduulin olisi myös mahdollista havaita laitteen vaarantuminen, ilman että laitteeseen tehdään mekanismia, joka ilmoittaa tällaisesta hyökkäyksestä. Tämä suojaus perustuu koneoppimiseen ja laitteen tilan vertaamiseen sen aikaisempaan tilaan. (Simpson, Roesner ja Kohno 2017) Viimeisenä ominaisuutena Simpson, Roesner ja Kohno (2017) ehdottavat tietoliikenteen nopeuden rajoittamista. Jos jokin verkon laite syöttää dataa liian nopeasti, moduuli estää tämän laitteen. Erityisesti palvelunestohyökkäyksissä laitteen tietoliikenteen määrä kasvaa suuresti kun laite alistetaan osaksi zombiverkkoa. Moduuli tunnistaa tilanteen ja reagoi siihen. Laitteen vaarantuessa on myös tärkeää estää sen pääsy muihin verkon laitteisiin, jotta muu verkko ei vaarannu.

Molemmissa ratkaisuissa on vahvuutensa, tutkijoiden Yu ym. (2015) Ratkaisu tarjoaa yksittäisille laitteille parempaa tietoturvaa, kun taas tutkijoiden Simpson, Roesner ja Kohno (2017) ratkaisu parantaa koko verkon yleistä turvallisuutta. Simpson, Roesner ja Kohno (2017) korostavat kuitenkin, että keskitetyn turvallisuusmoduulin suojaaminen itsessään on kriittistä, sillä se vastaa muiden laitteiden turvallisuudesta ja on tästä syystä hyvä kohde myös hyökkääjille. Kummallekin ratkaisulle on ominaista uuden laitteen lisääminen verkkoon vastaamaan tietoturvasta. Vaikuttaakin siltä, että on helpompaa lähteä vahvistamaan verkon turvallisuutta lisäämällä verkkoon tähän tarkoitukseen tarkoitettua turvallisuusmoduulia, kuin toteuttaa riittävä tietoturva verkon jokaiseen laitteeseen. Tätä johtopäätöstä tukee tutkijoiden Rahmani ym. (2015) tekemä tutkimus, jossa he ehdottavat palvelunestohyökkäyksen estämiseksi DTLS-kättelyä (Datagram Transport Layer Security Protocol), jolla reititin ja sensori todentavat toisensa. Tutkimus keskittyy turvallisen kommunikoimisen varmistamiseen

terveydenhuollossa, jossa yksityisyyden suoja ja tietoturva ovat erityisen tärkeässä roolissa. Tutkijat perustelevat turvallisuusmoduulin lisäämistä sillä, että moduuli keventää sensoreiden ja muiden verkon solmujen kuormitusta. Tämä on järkevää, sillä se mahdollistaa laskentatehon ja muistin käyttämisen sensoreissa ja päätelaitteissa tiedon salaamiseen.

5.2 Verkossa kulkevan datan salaaminen

Vaikka aiemmin esitetyillä tietoturvaa parantavilla toimilla voidaan estää datan joutuminen vääriin käsiin, on kuitenkin viisasta varautua myös tähän mahdollisuuteen. Tästä syystä kuljetettava data tulee salata niin, että ulkopuolinen tekijä ei saa alkuperäistä viestiä selville salatusta datasta. Salausalgoritmeja on useita, yleisin on varmasti AES (Advanced Encryption Standard), joka kehitettiin 2001. AES salaus on turvallinen, kun salauksen pituus on vähintään 128-bittiä. Salausta ei ole saatu murrettua esimerkiksi raa'an voiman menetelmillä, mutta SCA-hyökkäyksellä alkuperäinen salausavain on saatu haltuun, jolloin salaus raukeaa. 128-bittisenä AES salaus voi olla liian raskas IoT-laitteille. Tästä syystä tutkijat Lin, Tsai ja Kao (2017) parantelivat Googlen RAPPOR (Randomized Aggregatable Privary-Preserving Ordinal Response) salausalgoritmia tehden siitä toimintavarmemman ja laskuteholtaan kevyemmän. Tutkijoiden Lin, Tsai ja Kao (2017) kehittämä salausalgoritmi TBDR (Time Based Dynamic Response) tarjoaa 256-bittisen suojauksen jonka suoritusnopeus on kymmenes perinteisestä AES-suojauksesta.

5.3 Salausavaimen sopiminen IoT-ympäristössä

Tiedon salaamisen lisäksi on sovittava salausavaimesta jolla salaus voidaan purkaa. Roman ym. tutkivat julkisen avaimen kryptaamiseen ja ennalta määriteltyjen avainten liittyviä protokollia IoT-kontekstissa. Tällaisia turvallisuusprotokollia on useita, kuten TLS, jossa laitteet neuvottelevat yhteisen avaimen, joka varmistaa turvallisen lähetyksen kuljetuskerroksella. Niin TLS, kuin verkkokerroksen protokollat tarjoavat mekanismin jolla voidaan neuvotella yhteinen salainen avain kahden laitteen välillä. Useimmat näistä mekanismeista perustuvat julkisen avaimen kryptaamiseen (engl. public key cryptography) tai ennalta määritellyn avaimen jakamiseen. Julkisen avaimen kryptaaminen laitteissa vaatii laskutehoa, tai erillisen

kryptaamispiirin. Tästä johtuen tämä tapa ei sovi pienille IoT-laitteille joissa on rajallinen muisti ja laskuteho. Tämän sijasta voidaan käyttää ennalta määriteltyä jaettua avainta. (Roman ym. 2011) Salausavaimia voi olla yksi tai useampi ja useassa protokollassa ne vaihtuvat joka istunnon jälkeen.

Shah ja Venkatesan (2018) ehdottavat uutta menetelmää IoT-laitteiden yhteyden turvaamiseen. Sen sijaan että laitteet käyttäisivät vain yhtä salausavainta, ne käyttävätkin salausavainten sarjaa, joka on jaettu molemmille osapuolille, IoT-laitteelle ja serverille. Salausavaimet sisältävää tietokantaa kutsutaan holviksi ja alkuperäinen holvi on varastoitu IoT-laitteeseen. Tämä holvi jaetaan serverin kanssa, johon IoT-laite yhdistetään. Holvien sisältämät salasanat muuttuvat istunnon aikana vaihdetun datan perusteella. Tästä johtuen SCA-hyökkäykset (engl. side channel attack) eivät ole tehokkaita protokollaa vastaan. Jos salasana saataisiin haltuun SCA-hyökkäyksen avulla, siitä ei olisi hyötyä, sillä salasanoja on useampi ja ne vaihtuvat jatkuvasti. (Shah ja Venkatesan 2018) Shahin ja Venkatesanin (2018) kehittämä protokolla on tehokas estämään Dos-hyökkäykset, MitM-hyökkäykset, sekä salasanan ennakoinnin. Prokolla on myös virrankulutukseltaan tehokas, heidän kehittämä algoritmi kulutti 646 mikrojoulea, kun taas julkisen avaimen salausalgoritmi ECC kulutti 10995 mikrojoulea. Kun Shahin ja Venkatesanin algoritmilla käytetään useampaa kuin kahta avainta, salauksen varmuus paranee. Luonnollisesti korkeammalla salasanojen määrällä myös laskentatehon vaativuus ja energian kulutus nousee. (Shah ja Venkatesan 2018) Tästä johtuen olisi mielenkiintoista tutkia heidän kehittämän algoritmin salasanojen määrän optimaalista suhdetta laskutehon vaativuuden ja tietoturvallisuuden kannalta.

6 IoT-verkon turvallinen toteuttaminen

Jotta IoT-verkko voidaan toteuttaa turvallisesti, tulee ymmärtää että siihen kohdistuvat hyökkäykset voidaan kohdistaa jokaiseen verkon osa-alueeseen. Koska hyökkäykset voidaan kohdistaa laitteen eri kerroksiin, algoritmien suunnittelijoiden, ohjelmistokehittäjien, sekä laitteistosuunnittelijoiden tulee ymmärtää toistensa työtä, ja tehdä yhteistyötä. Muuten yhdessä kerroksessa toteutettu suojaus voi olla turha, jos suojaus pystytään purkamaan jossain toisessa kerroksessa. (Kocher ym. 2011) Nämä hyökkäykset voidaan myös toteuttaa lukuisilla eri tavoilla ja tästä johtuen IoT-verkon suojaaminen on vaikea tehtävä. Tämän takia jo verkkoa suunnitellessa on hyödyllistä hyväksyä, että tietoturvauhilta ei ole absoluuttisen varmaa suojautumisratkaisua, tietoturvauhkien riskiä voidaan vain minimoida. Riskien minimoinnin lisäksi, tiedon salaaminen on vahva keino parantaa tietoturvaa. Jos ulkopuolinen tekijä pääsee murtautumaan järjestelmään, hänen haltuunsa saama tieto ei ole hyödyntämiskelpoista salauksen ansiosta.

Aikaisemmissa kappaleissa on puhuttu IoT-elementeistä ja kerrosarkkitehtuurista. Näissä kappaleissa on esitetty IoT-verkkojen ja IoT-laitteiden suurimmat heikkoudet ja keskeisimmät käsitteet. IoT-elementeissä mainittu identifiointi on tärkeä aspekti verkon toteuttamisessa. Kun verkon jokainen laite identifioidaan, pystytään verkossa toimivia laitteita valvomaan paremmin, ja estämään ulkopuolisten laitteiden lisääminen. Kommunikoiminen on IoT-verkon peruskiviä, laitteiden tärkein tehtävä on kerätä tietoa ympäristöstä ja lähettää sitä eteenpäin. Tämän takia kommunikoinnin suojaaminen on erittäin tärkeä suojausalue, ellei jopa tärkein. Kuten aiemmin mainittiin, kuljetuskerros hoitaa laitteiden kommunikoimisen. Siihen kohdistuvat MitM-hyökkäykset ja palvelunestohyökkäykset ovat hyvin yleisiä. Ne aiheuttavat datan vuotamista ulkopuolisille ihmisille, sekä verkon alistamista palvelunestohyökkäykseen. Tämän lisäksi verkon toiminta hidastuu, tai loppuu kokonaan.

Tässä tutkielmassa on esitetty tapoja parantaa IoT-verkon turvallisuutta erilaisten turvallisuusmoduulien avulla. Tämä vaikuttaa olevan tällä hetkellä helpoin keino parantaa verkon turvallisuutta merkittävästi, sillä turvallisuusmoduuli tarjoaa ohjelmistopäivityksien nopean ja turvallisen asentamisen, tietoliikenteen suodattamisen estäen matojen ja haittaohjelmien lataamisen, sekä vahvemman tunnistautumisen. Turvallisuusmoduuli pystyy myös estämään

vaarannetun laitteen toiminnan ja pitämään muun IoT-verkon toimintakunnossa. Vaikka turvallisuusmoduuli parantaa IoT-verkon tietoturvaa merkittävästi, sensoreiden ja muiden laitteiden tulee silti salata lähetettävä data. Salaamiseen voidaan käyttää useita eri algoritmeja, kuten AES:ia, tai IoT-laitteille kehitettyä kevyempää salausalgoritmia, kuten aiemmin esiteltyä TBDR:ää.

Kun laitteiden välinen kommunikoiminen on suojattu, laitteiden välillä liikkuva data suojattu ja ulkopuolisten laitteiden lisääminen verkkoon estetty, tulee vielä verkon päätesolmut, eli sensorit suojata. Tässä tutkielmassa on esitetty useita eri SCA-hyökkäyksiä, joiden avulla sensoreiden keräämää dataa päästään tutkimaan ennen sen salaamista. Nämä SCA-hyökkäykset vaativat laitteeseen käsiksi pääsemisen, sekä siihen kohdistuvien fyysisten muutoksien tekemisen. Tämän lisäksi laitteen keräämää dataa voidaan seurata mittaamalla laitteen elektromagneettista säteilyä. Tästä johtuen SCA-hyökkäysten estämiseksi jokaiseen IoT-laitteeseen tulisi toteuttaa suojausmekanismi, joka antaa ilmoituksen jos laitteeseen kajotaan. IoT-laitteet tulisi myös asettaa sellaisiin paikkoihin, joihin niissä on vaikeaa päästä käsiksi, tai paikkaan, jossa niitä on helppo valvoa. Vastuu turvallisen IoT-verkon toteuttamisesta on tällä hetkellä niin laitevalmistajalla, kuin asiakkaallakin. Tuotevalmistajien tulee kehittää IoT-laitteiden tietoturvaa ja suojata laite fyysiseltä muuntelulta. Tämän lisäksi asiakkaan tulee olla tietoinen IoT-laitteen mahdollisista tietoturvauhista, ja vaihtaa laitteen oletussalasana, sekä asentaa uusimmat ohjelmistopäivitykset.

Lähteet

Arjunan, Amuthan, Praveena Narayanan ja Kaviarasan Ramu. 2016. “Securing RSA Algorithm against Timing Attack”. *The International Arab Journal of Information Technology* 13 (4). https://iajit.org/index.php?option=com_content&task=blogcategory&id=106&Itemid=391.

Balasch, Josep, Benedikt Gierlichs, Oscar Reparaz ja Ingrid Verbauwhede. 2015. “DPA, bitslicing and masking at 1 GHz”. Teoksessa *International Workshop on Cryptographic Hardware and Embedded Systems,* 599–619. Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium: KU Leuven Dept. Electrical Engineering-ESAT/COSIC ja iMinds. https://eprint.iacr.org/2015/727.pdf.

Bharathi, M. V., R. C. Tanguturi, C. Jayakumar ja K. Selvamani. 2012. “Node capture attack in Wireless Sensor Network: A survey”. Teoksessa *2012 IEEE International Conference on Computational Intelligence and Computing Research,* 1–3. Joulukuu. doi:10.1109/ICCIC.2012.6510237.

Bilge, Leyla, ja Tudor Dumitraundefined. 2012. “Before We Knew It: An Empirical Study of Zero-Day Attacks in the Real World”. Teoksessa *Proceedings of the 2012 ACM Conference on Computer and Communications Security,* 833–844. CCS ’12. Raleigh, North Carolina, USA: Association for Computing Machinery. ISBN: 9781450316514. doi:10.1145/2382196.2382284. https://doi.org/10.1145/2382196.2382284.

Brumley, David, ja Dan Boneh. 2003. “Remote timing attacks are practical”. *SSYM’03: Proceedings of the 12th conference on USENIX Security Symposium* 12 (1): 1–1. https://dl.acm.org/doi/proceedings/10.5555/1251353.

Burhan, M., R. Rehman, B. Khan ja B-S Kim. 2018. “IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey”. *Sensors 2018, 18, 2796* 18 (9). https://www.mdpi.com/1424-8220/18/9.

Conti, M., N. Dragoni ja V. Lesyk. 2016. “A Survey of Man In The Middle Attacks”. *IEEE Communications Surveys Tutorials* 18 (3): 2027–2051. ISSN: 2373-745X. doi:10.1109/COMST.2016.2548426.

Coppens, Bart, Ingrid Verbauwhede, Koen De Bosschere ja Bjorn De Sutter. 2009. “Practical mitigations for timing-based side-channel attacks on modern x86 processors”. Teoksessa *2009 30th IEEE Symposium on Security and Privacy,* 45–60. IEEE. doi:10.1109/SP.2009.19.

Darwish, D. 2015. “Improved Layered Architecture for Internet of Things”. *International Journal of Computing Academic Research, Volume 4, Number 4* 4 (4): 214–223. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.736.5922&rep=rep1&type=pdf.

F-Secure. 2019. *Attack Landscape H1 2019.* Tekninen raportti. https://blog-assets.f-secure.com/wp-content/uploads/2019/09/12093807/2019$_{a}ttack_{l}andscape_{r}eport.pdf$: F-Secure.

Han, J., Y. Jeon ja J. Kim. 2015. “Security considerations for secure and trustworthy smart home system in the IoT environment”. Teoksessa *2015 International Conference on Information and Communication Technology Convergence (ICTC),* 1116–1118. Lokakuu. doi:10.1109/ICTC.2015.7354752.

Hori, Yohei, Toshihiro Katashita, Akihiko Sasaki ja Akashi Satoh. 2012. “Electromagnetic side-channel attack against 28-nm FPGA device”. 1-1-1 Umezono, Tsukuba, Ibaraki 305-8586, Japan: National Institute of Advanced Industrial Science ja Technology. https://staff.aist.go.jp/hori.y/articles/hori_wisa2012.pdf.

Kaspersky. 2019. *IoT: A Malware Story.* Tekninen raportti. https://www.kaspersky.com/about/press-releases/2019$_{i}ot-under-fire-kaspersky-detects-more-than-100-million-attacks-on-smart-devices-in-h1-2019$: Kaspersky.

Kocher, Paul, Joshua Jaffe, Benjamin Jun ja Pankaj Rohatgi. 2011. “Introduction to differential power analysis”. *Journal of Cryptographic Engineering* 1 (1): 5–27. doi:https://doi.org/10.1007/s13389-011-0006-y.

Lin, C., C. Tsai ja C. Kao. 2017. “Lower power data transport protection for Internet of Things (IoT)”. Teoksessa *2017 IEEE Conference on Dependable and Secure Computing,* 468–470. 2017 IEEE Conference on Dependable / Secure Computing. doi:10.1109/DESEC.2017.8073865.

Pammu, A. A., K. Chong, W. Ho ja B. Gwee. 2016. “Interceptive side channel attack on AES-128 wireless communications for IoT applications”. Teoksessa *2016 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS),* 650–653. Lokakuu. doi:10.1109/APCCAS.2016.7804081.

Peng, Sheng-Lung, Souvik Pal ja Lianfen Huang. 2019. *Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm.* Springer.

Rahmani, A., N. K. Thanigaivelan, Tuan Nguyen Gia, J. Granados, B. Negash, P. Liljeberg ja H. Tenhunen. 2015. “Smart e-Health Gateway: Bringing intelligence to Internet-of-Things based ubiquitous healthcare systems”. Teoksessa *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC),* 826–834. 2015 12th Annual IEEE Consumer Communications / Networking Conference (CCNC). doi:10.1109/CCNC.2015.7158084.

Rhee, S. 2016. “Catalyzing the Internet of Things and smart cities: Global City Teams Challenge”. Teoksessa *2016 1st International Workshop on Science of Smart City Operations and Platforms Engineering (SCOPE) in partnership with Global City Teams Challenge (GCTC) (SCOPE - GCTC),* 1–4. Huhtikuu. doi:10.1109/SCOPE.2016.7515058.

Roman, Rodrigo, Cristina Alcaraz, Javier Lopez ja Nicolas Sklavos. 2011. “Key management systems for sensor networks in the context of the Internet of Things”. *Computers Electrical Engineering* 37 (maaliskuu): 147–159. doi:10.1016/j.compeleceng.2011.01.009.

Rouse, Margaret. *What is business Logic Attack?* https://whatis.techtarget.com/definition/business-logic-attack. Tarkastettu: 15.3.2020.

Said, Omar, ja Mehedi Masud. 2013. “Towards Internet of Things: Survey and Future Vision”. *International journal of computer networks* 5 (1): 1–17. http://www.cscjournals.org/journals/IJCN/issue-manuscripts.php?v=5&i=1.

Sethi, P., ja S. Sarangi. 2017. “Internet of Things: Architectures, Protocols, and Applications”. *Journal of Electrical and Computer Engineering* 2017. https://doi.org/10.1155/2017/9324035.

Sfar, Arbia, Enrico Natalizio, Yacine Challal ja Zied Chtourou. 2018. “A roadmap for security challenges in the Internet Of Things”. *Digital Communications and Networks* 4 (2): 118–137. https://www.sciencedirect.com/journal/digital-communications-and-networks/vol/4/issue/2.

Shah, T., ja S. Venkatesan. 2018. “Authentication of IoT Device and IoT Server Using Secure Vaults”. Teoksessa *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE),* 819–824. Elokuu. doi:10.1109/TrustCom/BigDataSE.2018.00117.

Simpson, A. K., F. Roesner ja T. Kohno. 2017. “Securing vulnerable home IoT devices with an in-hub security manager”. Teoksessa *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops),* 551–556. doi:10.1109/PERCOMW.2017.7917622.

Yu, Tianlong, Vyas Sekar, Srinivasan Seshan, Yuvraj Agarwal ja Chenren Xu. 2015. “Handling a Trillion (Unfixable) Flaws on a Billion Devices: Rethinking Network Security for the Internet-of-Things”. Teoksessa *Proceedings of the 14th ACM Workshop on Hot Topics in Networks.* HotNets-XIV. Philadelphia, PA, USA: Association for Computing Machinery. ISBN: 9781450340472. doi:10.1145/2834050.2834095. https://doi.org/10.1145/2834050.2834095.