Lassi Haarala

Kyberfyysisten järjestelmien turvallisuusuhat

Tietotekniikan kandidaatintutkielma

31. toukokuuta 2024

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Lassi Haarala

Yhteystiedot: laelhaar@student.jyu.fi

Ohjaaja: Annemari Auvinen

Työn nimi: Kyberfyysisten järjestelmien turvallisuusuhat

Title in English: Cyber-physical system security threats

Työ: Kandidaatintutkielma

Opintosuunta: Tietotekniikka

Sivumäärä: 23+0

Tiivistelmä: Kyberfyysiset järjestelmät ovat muovautuneet hyvin isoksi osaksi elämäämme, vaikka tämä ei ole aina niin näkyvää. Näitä järjestelmiä löytyy esimerkiksi kodeista, autoista tai työpaikoista ja ne saattavat olla vastuussa äärimmäisen tärkeistä tehtävistä. Tästä syystä on tärkeää tutkia miten näitä järjestelmiä voi kehittää turvallisiksi. Tämä kirjallisuuskatsaus pyrkii siis tuomaan esiin oleellisia uhkia sulautettujen järjestelmien valmistuksessa ja tuomaan selkoa niiden turvalliseen kehitykseen.

Avainsanat: Kyberfyysiset järjestelmät, Sulautetut järjestelmät, Asioiden Internet, Kyberturvallisuus

Abstract: Cyber-physical systems have become an integral part of out lives even though this isn't always so apparent. These systems can be found in our homes, cars and workplaces and they can be responsible of very crucial tasks. Therefore it is important to study how embedded systems can be developed to be safe. This literature review aims to showcase relevant threats when building embedded systems and to also introduce ways to develop them more safely.

Keywords: Cyber-physical systems, Embedded systems, Internet of Things, Cyber security

Sisällys

1 Johdanto

Kyberfyysiset järjestelmät ovat muovautuneet hyvin tärkeäksi osaksi nykyaikaista automaation ja digitalisaation ilmapiiriä ja niiden käyttö kehittyy jatkuvasti. Nämä järjestelmät ovat yhdistelmä laitteistoa ja ohjelmistoa, mitkä toimivat esimerkiksi puettavan teknologian, kodinkoneiden, turvallisuusjärjestelmien tai ajoneuvojen älynä, prosessoiden dataa ympäristöstään tai ihmisiltä saamistaan syötteistä. Yleisesti jokainen järjestelmä on luotu vastamaan tietystä tehtävästä ja tästä seuraakin jokaisen järjestelmän hyvin yksilöllinen luonne (Yekini 2022). Järjestelmän kehittäjät joutuvat jo varhain miettimään esimerkiksi tarvittavan muistin määrää, järjestelmän mahdollista kommunikointiteknologiaa tai kulutetun virran määrää. Itse piirilevyn ominaisuuksien lisäksi voi olla tehtävän mukaan myös oleellista kerätä ympäristöstä dataa erilaisilla sensoreilla (Yekini 2022).

Tämä kandidaatintutkielma toimii johdatuksena kyberfyysisten järjestelmien turvalliseen kehitykseen. Tutkielmassa selvitetään kyberfyysisen järjestelmän karkeat hyökkäyspinnat, jotka ovat tämän tutkielman puitteissa ohjelmistotaso, laitteistotaso ja verkkotaso. Näiden tasojen haasteet tullaan selventämään ja tämän lisäksi haasteista syntyvät mahdolliset uhat pyritään kuvaamaan helposti ymmärrettävästi. Uhkiin tarjotaan myös tarjolla olevia ratkaisuja. Tutkielma on toteutettu kirjallisuuskatsauksena.

Kyberfyysisten järjestelmien turvallisuus on ajankohtainen ongelma ohjelmistokehityksessä. Älylaitteita voi löytyä terveydenhuollon välineistä, ajoneuvoista ja kodeistamme. Näiden laitteiden ero tavanomaisiin ohjelmistoihin on se, että niiden turvallisuusongelmat voivat johtaa suoraan jopa ihmisvahinkoihin. Esimerkiksi itseajava auto tai sydämentahdistin täytyy suunnitella niin, että se häiriö- tai hyökkäystilanteessa aiheuttaa mahdollisimman pientä vahinkoa.

Tutkielman seuraavassa luvussa käsitellään haasteita kyberfyysisten järjestelmien turvallisessa suunnittelussa. Kolmas luku tulee avaamaan oleellisimpia uhkia, joita näistä järjestelmien haasteista voi syntyä. Neljäs luku tarjoaa mahdollisia ratkaisuja näihin uhkiin ja haasteisiin ja viimeisessä luvussa tullaan poimimaan tärkeimmät havainnot tutkielmasta.

2 Kyberfyysinen järjestelmä

Kyberfyysinen järjestelmä on kokonaisuus, joka integroi fyysisen ympäristön ja siitä dataa keräävän ja sitä prosessoivan sulautetun järjestelmän (Marwedel 2021; Aloseel ym. 2020). Nämä prosessoivat laitteet voivat olla rakennettu käyttöjärjestelmättömien tietokoneiden (engl. *bare-metal*), vuorotusta hyödyntävien reaaliaikaisten käyttöjärjestelmien (engl. *Real time operating system*) tai kokonaisten käyttöjärjestelmien varaan (Salehi ym. 2023; Marwedel 2021; Amos 2020). Tämän lisäksi tutkielmassa on esillä myös asioiden internet (engl. *Internet of Things, IoT*), joka viittaa kattavaan monesta heterogeenisestä sulautetusta järjestelmästä koostuvaan verkostoon (Marwedel 2021; Ali, Ali ja Badawy 2015).

Termin 'kyberfyysinen järjestelmä' käyttö ei ole ristiriidatonta. Toisinaan oleelliseksi mielletään yhteys fyysiseen maailmaan ja toisinaan kommunikaatio on oleellista (Marwedel 2021). Kuitenkin tässä tutkielmassa tullaan jaottelu tekemään niin, että kyberfyysisistä järjestelmistä puhuttaessa painottuvat yhteydet laskennan ja fyysisen maailman välillä. Tällöin IoT-järjestelmistä puhuttaessa nousee kommunikointi laitteiden välillä, erityisesti internetin välityksellä, oleelliseksi.

2.1 Haasteet kyberfyysisissä järjestelmissä

Sulautettujen järjestelmien suunnittelussa on otettava huomioon monia haasteita, joita tuotetta kehittäessä voi nousta esiin.

Ensimmäisenä haasteena on järjestelmän luotettavuus. Tähän kategoriaan liittyy järjestelmän tietoturvallisuus, ihmisten henkiin kohdistuva turvallisuus, toimintavarmuus, korjattavuus ja saatavuus (Marwedel 2021, luku 1.3). Tietoturvallisuudella pyritään takaamaan tiedon luottamuksellisuus (engl. *confidentiality*), eheys (engl. *integrity*) ja saatavuus (engl. *availability*) (Marwedel 2021). Luottamuksellisuudella viitataan käyttörajoituksiin, jotka estävät väärien tahojen pääsyn informaatioon (Yee ja Zolkipli 2021). Eheydellä tarkoitetaan, ettei informaatiota ole kukaan pystynyt muokkaamaan (Yee ja Zolkipli 2021). Saatavuus taas takaa käyttäjän pääsyn omaan informaatioonsa (Yee ja Zolkipli 2021). Ihmishenkien turvaaminen on hyvin yksikäsitteistä. Laiteen järjestelmä- ja laitteistoviat ja -vauriot eivät saa aiheuttaa suurta

riskiä henkilövahingoille (Marwedel 2021; Xie ym. 2017). Toimintavarmuus kuvaa todennäköisyyttä sille, ettei laite hajoa määritellyssä aikamääreessä (Marwedel 2021). Korjattavuus ja saatavuus kuvaavat todennäköisyyttä sille, että vaurioitunut laite on korjattavissa, ja että laitetta on mahdollista ylipäänsä hankkia.

Toinen oleellinen haaste liittyy resurssien käyttöön. Tämä haaste voidaan jakaa alakategorioihin, jotka ovat energian kulutus, suoritusajan resurssien käyttö ja koodin sekä muistin koko (Marwedel 2021; Abbasi ym. 2019). Vähäinen energiankulutus on tärkeää sulautetun järjestelmän tehokkuuden ja kestävyyden kannalta, mutta myös esimerkiksi ekologisista ja ekonomisista syistä (Marwedel 2021; Guo ym. 2021). Vaikka korkeampi energiankulutus mahdollistaakin suuremman laskentatehon, on tärkeää ottaa huomioon myös energian luoma lämpö, jolla voi olla haittavaikutuksia järjestelmään (Marwedel 2021). Suoritusajan resurssien tehokkaalla käytöllä pyritään algoritmien ja laitteiston optimisointiin. Esimerkiksi voidaan pyrkiä vähentämään hitaampien ydinten ylikuormitusta järjestelmissä, joissa on useampia heterogeenisiä ytimiä (Marwedel 2021; Xiang ja Pasricha 2015). Koodin ja muistin koot joudutaan pitämään monessa sulautetussa järjestelmässä pienenä. Tämän seurauksena järjestelmään voi olla vaikeaa lisätä ominaisuuksia, ja raskaasti muistia hyödyntävää laskentaa voi olla mahdotonta suorittaa (Marwedel 2021; Abbasi ym. 2019).

Kyberfyysisen järjestelmän ongelmat voivat nousta esiin myös laitteen fyysisestä puolesta. Suojaamattomat ja usein pitkiä aikoja koskemattomana viettävät laitteet saattavat olla asetettuina mahdollisesti haitallisiin sijainteihin (Fysarakis ym. 2014). Esimerkiksi sensoreiden vauriot, lisääntyneeseen virrankulutukseen pyrkivät hyökkäykset tai pyrkimykset kerätä tietoa järjestelmästä fyysisen laitteen perusteella ovat kaikki oleellisia laitteen fyysisiä uhkia (Fysarakis ym. 2014; Aloseel ym. 2020).

Sulautettu järjestelmä on siis osa kybefyysisen järjestelmän kokonaisuutta ja toimiikin sen älynä (Marwedel 2021). Onkin siis tärkeää avata miten nämä yksittäiset sulautetut järjestelmät voivat omilla tavoillaan luoda uhkia riippuen ominaisuuksistaan.

Monet sulautetut järjestelmät ovat käyttöjärjestelmättömiä, joka tarkoittaa sitä, että ohjelma suoritetaan suoraan laitteistolla (Degani ym. 2023; Amos 2020; Salehi ym. 2023). Nämä ratkaisut sopivat tilanteisiin, joissa halutaan kuluttaa mahdollisimman vähän energiaa. Kuiten-

kin ne myös kärsivät rajoitetuista laitteiston resursseista ja usein myös muistinhallintayksikön (engl. *MMU, Memory Management Unit*) puutteesta (Zhou ym. 2022; Salehi ym. 2023). Nämä puutteet mahdollistavat sen, että laitteella olevalla koodilla on vapaa pääsy esimerkiksi kaikkeen muistiin sekä sulautettuun laitteisiin kytkettyihin sensoreihin (Salehi ym. 2023). Ohjelmoijat ovat useasti hoitaneet muistin hallinnan huolimattomasti ja aiheuttaneet sulautettuun järjestelmään muistiin liittyviä haavoittuvaisuuksia (Salehi ym. 2023).

Sulautettu järjestelmä voi toimia myös käyttöjärjestelmän varassa, jolloin yleensä on käytössä jokin Linux-jakelu (Li, Matsubara ja Takada 2018). Näin ollen Linux-pohjaiset sulautetut järjestelmät perivät Linux-ytimen (engl. *kernel*) turvallisuusongelmia. Koska nämä laiteet voivat olla käytössä pitkiä aikoja ilman päivityksiä, ei niissä havaittuja tietoturvaongelmia välttämättä ratkaista (Marwedel 2021).

Kuten aiemmin todettu, IoT-ratkaisuista puhuttaessa tullaan keskittymään erityisesti verkko-ominaisuuksien luomiin haasteisiin. IoT-ratkaisuista puhuttaessa, nouseekin oleellisiksi haasteiksi laitteiden heterogeenisyys, ongelmat kommunikaatioprotokollien toteutuksessa, kevyet algoritmit suojauksissa, sulautetuille laitteille tyypillinen resurssien niukkuus sekä yleisemmin ohjelmoinnin tietoturvallinen luonne, joka pystyisi estämään tyypillisten hyökkäysten tapahtumisen verkon välityksellä (Hassija ym. 2019; Khelif ym. 2020). Esimerkiksi SQL- tai koodi-injektiot, käyttäjien manipulointi, palvelunestohyökkäykset tai luvaton pääsy IoT-verkkoon ovat kaikki tilanteita, joihin tulisi varautua IoT-ratkaisuja kehittäessä. IoT-ratkaisujen hyökkäyspinta-ala onkin hyvin laaja sen muodostamien sulautettujen laitteiden ansiosta ja jokaisen yksittäisen sulautetun laitteen ongelmat voivat heijastua koko verkkoon (Hassija ym. 2019; Sadeghi, Wachsmann ja Waidner 2015).

3 Uhat ja haavoittuvaisuudet

Luvussa 2.1. kerrotut haasteet voivat siis synnyttää uhkia ja haavoittuvaisuuksia. Uhat voivat realisoitua kyberfyysisen järjestelmän eri kerroksilla, kuten laitteiston tasolla aistintakerroksella (engl. *sensing layer*) voi tapahtua sensoreihin kohdistuvia hyökkäyksiä. Ohjelmistotasolla ongelmat voivat taas esiintyä väliohjelmisto- tai verkkokerroksilla (engl. *middleware layer, network layer*). Tällöin uhat ovat esimerkiksi haittaohjelmia tai kommunikaatioprotokollien puutteita hyödyntäviä hyökkäyksiä (Hassija ym. 2019; Khelif ym. 2020).

Uhkiin johtavat haavoittuvaisuudet voivat olla seurausta monesta asiasta, mutta esimerkiksi jo aiemmin pohjustettu huolimaton ohjelmistokehitys on syynä moniin haavoittuvaisuuksiin, jotka mahdollistavat hyökkääjälle hallinnan järjestelmästä. Esimerkiksi puskurin ylivuodosta aiheutuvat ongelmat ja huolimaton muistinhallinnan suunnittelu aiheuttavat näitä mahdollisuuksia (Papp, Ma ja Buttyan 2014). Tätä haavoittuvaisuutta voidaan pitää seurauksena Marwedelin (2021) kuvailemista järjestelmän luotettavuuden takaamisen haasteista, joihin liittyi olennaisesti tietoturvallisuus. Toinen oleellinen haavoittuvaisuus on laitteiden pitkä itsenäinen toiminta-aika. Laitteiden oletetaan toimivan pitkiä aikoja itsenäisesti, jolloin niitä ei mahdollisesti päivitetä. Myös laitteiden sijainti ei aina ole otollinen, vaan laite voi vahingoittua fyysisesti (Papp, Ma ja Buttyan 2014; Fysarakis ym. 2014). Vielä yhdeksi tärkeäksi haavoittuvaisuudeksi nousevat heikon pääsynvalvonnan ja kryptografian käyttö. Esimerkiksi heikkojen salasanojen tai oletussalasanojen käyttö, sekä heikosti luodut kryptografiset avaimet lisäävät mahdollisten uhkien syntyä. Tätä haavoittuvaisuutta voi aiheuttaa haaste kyberfyysisten järjestelmien rajallisten resurssien oikeanlaisesta käytöstä (Papp, Ma ja Buttyan 2014; Fysarakis ym. 2014).

3.1 Ohjelmiston ja verkko-ominaisuuksien haasteista syntyvät uhat

Ohjelmistotason uhat syntyvät usein huonosti suoritetusta ohjelmistokehityksestä ja ne voivat olla seurausta haasteesta hallita laitteiden rajoitettuja resursseja (Papp, Ma ja Buttyan 2014; Marwedel 2021; Abbasi ym. 2019). Tämä heikko kehitys saattaa ilmentyä esimerkiksi heikkona puutteellisten kryptografisten ratkaisujen käyttönä, heikkona pääsynhallinta-

na (engl. *access control*) tai heikkoina tunnistautumismekanismeina (Papp, Ma ja Buttyan 2014). Tyypilliset salasanoihin pohjautuvat mekanismit voivat olla epäturvallisia monesta syystä. Ihmiset saattavat käyttää helposti muistettavia heikkoja salasanoja, laitteessa saattaa olla oletussana, jota ei tarvitse vaihtaa, tai laiteessa voi olla kehittäjän jättämiä keinoja ohittaa tunnistautumismekanismi (engl. *backdoor*) (Fysarakis ym. 2014; Papp, Ma ja Buttyan 2014). Hyökkääjän pääsyllä laitteisiin voi olla monenlaisia seurauksia. Suoria seurauksia voi olla esimerkiksi rahalliset menetykset tai tietovuodot. Pääsystä voi myös seurata epäsuoria haittoja (Papp, Ma ja Buttyan 2014). Hyökkääjä voi ajaa laitteella olevaa koodia haittatarkoituksessa tai mahdollisesti laite voidaan liittää osaksi botnettiä, jolloin konetta voidaan käyttää hyökkäysvälineenä uusiin kohteisiin (Papp, Ma ja Buttyan 2014).

Aiemmin todetut haasteet verkko-ominaisuuksissa, kuten puutteet ohjelmoinnissa, ongelmat protokollien toteutuksissa tai laitteiden heterogeenisyys voivat myös synnyttää erilaisten uhkien mahdollisuuksia sulautetuissa järjestelmissä. Nämä uhat voivat olla erilaisia hyökkäyksiä verkko- ja väliohjelmistokerroksilla, joita ovat esimerkiksi palvelunestohyökkäykset, man-in-the-middle -hyökkäykset tai SQL-injektiot (Khelif ym. 2020).

3.1.1 Haittaohjelmat, Botnetit ja DDoS

Haittaohjelmia on monenlaisia ja ne koskevat myös kyberfyysisiä järjestelmiä. Esimerkiksi näppäinpainalluksia tallentavat ohjelmat (engl. *keylogger*, vakoiluohjelmat tai laitteita saastuttavat botit, joiden tarkoitus on antaa laitteen hallintaoikeudet ulkopuoliselle hyökkääjälle, ovat haittaohjelmia, jotka voisivat vaikuttaa myös sulautettuihin laitteisiin (Or-Meir ym. 2019). Lisävaikeuksia voivat aiheuttaa aiemmin mainitut web-käyttöliittymät, joita jotkin sulautetut laitteet hyödyntävät. Koska sulautettujen järjestelmien oletetaan toimivan yksinään pitkiä aikoja, näiden laitteiden verkkopalvelinsovelluksia harvemmin päivitetään, jolloin niissä piilee mahdollisia korjaamattomia turvallisuusongelmia (Papp, Ma ja Buttyan 2014).

Botnet on laitteita saastuttava haittaohjelma, joka saastuttaessaan liittää uuden laitteen osaksi botnettiä, mikä mahdollistaa hyökkääjän käskyttää sitä (Jurcut ym. 2020). Erityisesti huonosti suojatut IoT-laitteet ovat potentiaalinen kohde saastuttamiselle, jolloin IoT-laitteita voi

käyttää hyökkäyksissä edelleen uusien kohteiden kimppuun (Jurcut ym. 2020; Antonakakis ym. 2017). Historiallisesti botnettejä on käytetty ainakin hajautettujen palvelunestohyökkäysten (engl. *DDoS attack, Distributed Denial of Service attack*) suorittamiseen (Antonakakis ym. 2017).

DDoS-hyökkäyksillä voidaan estää erilaisten palveluiden, laitteiden tai laitteiden yhdistävän verkon toiminta ainakin väliaikaisesti kuormittamalla kohteita suurilla määrillä laitteiden välistä kommunikaatiota (Fysarakis ym. 2014; Hassija ym. 2019). Erityisesti vanhat protokollat, joiden turvallisuusvaatimukset ovat melko heikkoja, ovat alttiita väärinkäytölle (Jurcut ym. 2020). Tällä keinolla pyritään usein kaatamaan esimerkiksi verkkosivu tai palvelin ja estämään aitojen käyttäjien saama hyöty (Fysarakis ym. 2014; Hassija ym. 2019).

3.1.2 Protokollien puutteet ja man-in-the-middle hyökkäykset

Palvelunestohyökkäysten tavoin MITM-hyökkäykset (engl. *Man-in-the-middle*) käyttävät hyödykseen kommunikaatioprotokollien toteutuksissa olevia puutteita. Esimerkiksi WiFi- ja Bluetooth-yhteyksillä on onnistuttu suorittamaan näitä hyökkäyksiä (Khelif ym. 2020). MITM-hyökkäyksissä hyökkääjä pyrkii pääsemään kahden kommunikoivan laitteen väliin ja tästä asemasta kaapata laitteiden välistä kommunikaatiota tai manipuloida sitä ja näin lähettää toiselle päätteelle muokattua informaatiota esittäen olevansa toinen kommunikaation pääte (Khelif ym. 2020; Olazabal, Kaur ja Yeboah-Ofori 2022). Mikäli hyökkääjä onnistuu suorittamaan tällaisen hyökkäyksen IoT-laitteiden kommunikoimalla datalla, kertoo se selvästi Yeen ja Zolkiplin (2021) kuvaileman eheyden haasteesta IoT-laitteissa. O´Connor, Jessee ja Campos (2021) havaitsivatkin 16 laitevalmistajan hyödyntävän epäturvallisia protokollia, joista voi seurata merkittäviä uhkia laitteiden ja niiden käsittelemän informaation eheydelle. Myös haaste rajoitetuista resursseista on oleellinen protokollien puutteita pohtiessa. Esimerkiksi IoT-laitteille suunniteltu kevyt MQTT-protokolla (engl. *Message Queue Telemetry Transport*) sopii hyvin resursseiltaan rajoitettuihin IoT-järjestelmiin kommunikaatioprotokollaksi, mutta esimerkiksi alkuperäisen MQTT-protokollan suunnittelussa turvallisuus jäi kokonaan toisten protokollien suoritettavaksi, eikä MQTT-protokollassa ole oletukselta monia turvallisuusmekanismeja käytössä, kuten esimerkiksi tiedon salausta (Dinculeanã ja Cheng 2019; Gebremichael ym. 2020).

3.2 Fyysisten haasteiden aiheuttamat uhat

Kyberfyysisen järjestelmän laitteisto voi kohdata uhkia hyvin laajalla skaalalla. Laitteet voivat olla fyysisesti haastavissa paikoissa, jolloin ne voivat kärsiä ympäristön vaikutuksista sekä fyysisistä hyökkäyksistä laitetta kohtaan. Aktiivisten fyysisten haittojen, kuten luonnonilmiöiden ja hyökkääjien lisäksi, laitteiden hankala sijainti voi myös viitata siihen, että niiden ohjelmiston päivittäminen voi olla vaikeaa (Aloseel ym. 2020; Papp, Ma ja Buttyan 2014). Tämä itsessään lisää uhkia myös muilla hyökkäyspinnoilla mahdollistamalla tunnettujen haavoittuvuuksien hyväksikäyttämisen laitteilla, joilla ongelmaa ei ole korjattu päivityksillä.

3.2.1 Laitteiston analysointi ja fyysiset haitat

Laitteiston toimintaa voi analysoida esimerkiksi sivukanavahyökkäyksillä (engl. *side-channel attacks*), joilla on tarkoituksena kerätä tietoa kyberfyysisestä järjestelmästä analysoimalla esimerkiksi virrankulutusta, suoritus-aikaa, elektromagneettisen säteilyn määrää tai, jos hyökkääjällä on suora pääsy sulautettuun laitteeseen, voi hän mahdollisesti tarkkailla datan siirtoa suoraan dataväylältä. Näitä analysoimalla voi hyökkääjä päätellä esimerkiksi salausavaimia tai viestejä (Hassija ym. 2019; Fournaris, Fraile ja Koufopavlou 2017; Liptak, Mal-Sarkar ja Kumar 2022).

Fyysiset haitat voivat olla monenlaisia. Hyökkääjä voisi esimerkiksi peukaloida laitetta eri tavoin, hajottaa sen osan tai koko laitteen tai vaihtaa yhden IoT-verkon laitteista toiseen, hyökkääjän ohjaamaan, laitteeseen (Atlam ja Wills 2019). Myös laitteistoon voi kohdistaa palvelunestohyökkäyksiä. Esimerkiksi radiotaajuushäirinnällä ja univajehyökkäyksillä voi estää laitteiden toiminnan. Univajehyökkäyksillä (engl. *sleep deprivation attack*) pyritään lyhentämään akku- ja patterikäyttöisten kyberfyysisten järjestelmien toiminta-aikaa, estämällä laitteen pääsyn virransäästötilaan (Fobe, Nogueira ja Batista 2022). Radiotaajuushäirinnällä taas estetään radiotaajuustunnistusta (engl. *radio frequency identification*) hyödyntävien laitteiden toiminta luomalla radiotaajuushäiriöitä (Atlam ja Wills 2019). Näissä molemmissa taustalla on haaste laitteiden fyysisestä suojaamisesta. Tätä haastetta niin ikään aiheutti järjestelmien haitalliset sijainnit ja valvonnan puute (Fysarakis ym. 2014).

3.2.2 Datan kaappaaminen ja sen manipulointi

Laitteiden suojaamattomasta sijainnista voi seurata myös esimerkiksi datan kaappaamista. Datan kaappaamisella viitataan passiiviseen salakuunteluna tunnettuun ilmiöön (engl. *eavesdropping*), jossa hyökkääjä pyrkii kaappaamaan sulautetun laitteen vastaanottamia tai lähettämiä paketteja. Paketit voivat sisältää mahdollisesti huonosti suojattua dataa, jota voidaan hyödyntää valheellisen paketin tai syötteen luonnissa (Papp, Ma ja Buttyan 2014; Hassija ym. 2019). Tämä ongelma on oleellisesti yhteydessä muihin kategorioihin, mutta taustalla vaikuttavana syynä toimii erityisesti laitteen suojaamaton sijainti.

Kuten todettu, paketteja tutkimalla voidaan siten luoda valheellisia manipuloituja paketteja, joita lähettää sulautetulle laitteelle. Toisaalta myös sulautettujen laitteiden saamia syötteitä voidaan manipuloida ja syöttää valheellisia arvoja sulautettuihin laitteisiin ja niiden kautta kokonaisiin IoT-verkkoihin (Atlam ja Wills 2019; Papp, Ma ja Buttyan 2014). Tämä on oleellinen haaste eheyden toteuttamisessa ja näillä keinoilla voidaan aiheuttaa protokollien, sulautettujen laitteiden ja IoT-verkkojen toimintahäiriöitä (Hassija ym. 2019; Papp, Ma ja Buttyan 2014; Yee ja Zolkipli 2021).

3.3 Ihmiselementti

Käyttäjien manipulointi on myös oleellinen uhka tietoturvallisuudelle ja ihmisiin kohdistuvat hyökkäykset voivat olla monenlaisia. Usein niissä kuitenkin pyritään hyväksikäyttämään ihmistä, jotta päästään käsiksi johonkin järjestelmään. Hyväksikäyttö voi tapahtua valehteluna tai psykologisin keinoin ihmisten manipulointina (Wang, Sun ja Zhu 2020). Näillä keinoilla halutaan saada tietoa järjestelmästä tai päästä siihen käsiksi. IoT-laitteet ovat tuoneet tähän yhtälöön lisää kaapattavaa dataa, uusia hyökkäyksen kohteita ja uusia kommunikaatiomahdollisuuksia (Wang, Sun ja Zhu 2020). Vaikka tälle on vaikeaa määritellä tiettyä haastetta sulautettujen järjestelmien suunnittelussa, järjestelmään käsiksi pääsy aiheuttaa ongelmia erityisesti muilla hyökkäyspinnoilla. Esimerkiksi laitteiden lähelle pääsyä vaativat sivukanavahyökkäykset, fyysisen vahingon tuottaminen tai haittaohjelmien syöttäminen laitteille ovat esimerkkejä tilanteista, joissa hyökkääjä voisi käyttää manipulointia päästäkseen laitteiden lähelle ja laukaistaakseen sieltä hyökkäyksiä (Papp, Ma ja Buttyan 2014).

4 Turvallisuusratkaisuja ongelmiin

Koska kyberfyysiset järjestelmät ovat monipuolisia, on niiden turvaamisessa otettava huomioon niiden kaikki ulottuvuudet. Turvallisessa järjestelmässä tulisi näin olla otettu huomioon ohjelmisto-, verkko- ja laitteistotason haasteet ja uhat. Puolustuksen suunnittelu kaikilla tasoilla tulisi olla kolmiosaista. Ensimmäisellä tasolla tulisi pyrkiä suunnittelemaan järjestelmä lähtökohtaisesti mahdollisimman tietoturvalliseksi hyödyntämällä salaustekniikoita ja tunnistautumista. Toisella tasolla keskitytään havaitsemaan käynnissä oleva hyökkäys analysoimalla laitteen toimintaa eri tavoin. Kolmannella tasolla pyritään oppimaan hyökkäyksestä ja parantamaan laitteen turvallisuutta kehittämällä suojausmekanismeja opituilla asioilla (Aloseel ym. 2020). Ratkaisut eivät kuitenkaan ole ilmiselviä kyberfyysisiä järjestelmiä suunnitellessa. Järjestelmän rajalliset laskennalliset resurssit, fyysisen kokonaisuuden suojaaminen, verkko-ominaisuuksien suojaaminen ja ihminen osana tätä kaikkea luovat prosessista hyvin vaikeasti lähestyttävän.

4.1 Ohjelmistotason uhkien ratkaisuista

Ohjelmistotasolla ongelmiksi totesimme siis heikot puolustusmekanismit, jotka altistavat laitteen erilaisille haittaohjelmille. Tämän ongelman syiksi havaittiin ainakin heikosti suoritettu ohjelmistokehitystyö sekä kyberfyysisten järjestelmien rajalliset laskennalliset resurssit.

Ohjelmistokehitystä voisi parantaa turvallisen suunnittelun ajattelumallilla (engl *security by design*), jossa pyritään luomaan kyberfyysisestä järjestelmästä mahdollisimman turvallinen suunniteluvaiheessa. Esimerkiksi hyvin varhaisessa suunnittelu- ja kehitysvaiheessa aloitettu vihamielinen testaaminen lisää kokonaisvaltaista tietoutta järjestelmän kohtaamista uhista (Aloseel ym. 2020; Atlam ja Wills 2019). Tietoturvallisuutta lisäisi myös parempien salaustekniikoiden käyttö, mutta haasteeksi nousevat jälleen rajalliset resurssit. Rajallisten resurssien ongelmaa voisi pyrkiä ratkaisemaan hyödyntämällä pilvi- tai sumulaskentaa (engl. *cloud computing, fog computing*). Pilvilaskentaa hyödyntäessä laskennallisesti hankalia prosesseja voidaan suorittaa toisella laitteella, jonne voidaan myös tallentaa suuria määriä dataa. Erityi-

sesti runsaita määriä dataa ympäröivästä maailmasta keräävät järjestelmät hyötyvät suuresti pilvilaskennan eduista (Jassas ym. 2017). Kuitenkin tehtävien oikeanlainen vuorottaminen algoritmeillä on tärkeää ottaa huomioon, jotta raskaat laskennalliset tehtävät suoritetaan pilvessä ja kevyet tehtävät voidaan hoitaa paikallisella laitteella, jolloin säästytään mahdolliselta latenssilta (Jassas ym. 2017). Sumulaskennan voi ajatella olevan laajennus pilvilaskennasta. Sumulaskennassa hyödynnetyt laitteet voivat olla mitä tahansa laitteita, joissa on laskentatehoa, tallennustilaa ja mahdollisuus verkkoyhteyksille (Atlam, Walters ja Wills 2018). Hyöty tässä tulee siitä, että laskentaa pystytään suorittamaan lokaalisti, jolloin latenssilta säästytään, erityisesti jos laitteet keräävät valtavasti dataa. Pilvi- ja sumulaskenta voivat siis parantaa laitteen turvallisuutta lisäämällä muistin ja laskennan resursseja laitteelle (Hassija ym. 2019; Atlam, Walters ja Wills 2018). Tämä toki tarkoittaa sitä, että yksittäiseen laitteeseen tulee laittaa lisää rahaa resurssien ja virrankulutuksen muodossa.

Uudelleenisännöinti (engl. *rehosting*) on myös eräänlainen tapa lisätä turvallisuutta kyberfyysisten järjestelmien kehityksessä. Se perustuu laitteiston emulointiin niin, että laiteohjelmistoa (engl. *firmware*) ja laitteistoa pystyttäisiin analysoimaan virtuaalisessa ympäristössä. Tämä virtuaalinen ympäristö mahdollistaa kokonaisvaltaisten turvallisuusanalyysin suorittamisen, sekä myös tarjoavat laajemmat mahdollisuudet esimerkiksi laitteiden muuttamiselle, skaalaamiselle ja tutkimiselle (Fasano ym. 2021). Virtuaaliympäristössä voi siis suorittaa turvallisuusanalyysejä, esimerkiksi fuzz-testausta (engl. *fuzzing*). Fuzz-testauksessa kohteeseen syötetään useita satunnaisia tai ennalta määriteltyjä arvoja, mikä mahdollistaa hyvin tehokkaan testaamisen (Fasano ym. 2021; Salehi ym. 2023)

4.2 Verkkotason uhkien ratkaisuista

Verkkotason uhista nousivat oleellisesti esille protokollien heikkouksia hyödyntävät hyökkäykset, kuten palvelunestohyökkäykset ja MITM-hyökkäykset. Luvussa 4.1. esitetty turvallisen suunnittelun ajattelumalli on oleellista pitää suunnittelussa mukana myös tämän kategorian haasteissa. Verkkotasolla voidaan myös suojella järjestelmää palomuurien ja tunkeutumista havaitsevien järjestelmien avulla, jotka kuitenkin vaativat aiempaa tietoutta haitallisesta ja hyväksytystä toiminnasta, jotta ne voisivat toimia halutulla tavalla (Hamza, Gharakheili ja Sivaraman 2020). Verkko-ominaisuuksien turvaamiseen voisi kehittää omia jär-

jestelmiä, jotka voisivat analysoida laitteiden kommunikaatiossa lähettämien pakettien pituuksien ominaisuuksia, lähetettyjen ja vastaanotettujen pakettien määriä tai muita ominaisuuksia haitallisen toiminnan havaitsemiseksi. Toinen keino lisätä turvallisuutta olisi myös suorituksenaikainen laitteiden viestinnän tarkkailu esimerkiksi erilaisilla tunnisteita, käytöstä tai poikkeavuuksia havaitsevilla järjestelmillä (Hamza, Gharakheili ja Sivaraman 2020). Tämän lisäksi esimerkiksi aiemmin mainittu sumulaskenta voi myös tarjota turvallisuutta, sillä sumukerros (engl. *fog layer*) lisää ylimääräisen suojakerroksen, joka esimerkiksi mahdollistaa epätavallisen toiminnan havaitsemisen IoT-verkossa (Hassija ym. 2019).

Botnettien laukaisemia palvelunestohyökkäyksiä voisi esimerkiksi havaita aiemmin mainituilla tunniste- ja poikkeavuuspohjaisilla mekanismeilla. Tunnistepohjaiset mekanismit havainnoivat paketteja ja etsivät niistä merkkejä aiemmin huomatuista hyökkäyksistä. Tapa on kuitenkin melko raskas ja uusia uhkia sillä ei välttämättä voi havainnoida. Poikkeavuuksien havainnointiin on ehdotettu esimerkiksi pakettien kokojen, yksisuuntaisten yhteyksien ja uniikkien porttien tarkkailua erilaisilla keinoilla (Hamza, Gharakheili ja Sivaraman 2020).

4.3 Laitteiston uhkien ratkaisuista

Laitteiston haasteet ja uhat ovat hyvin monipuolisia. Laitteiden haitallinen sijainti aiheuttaa laitteille fyysistä uhkaa hyökkääjistä ja muusta ympäristöstä. Hyökkäykset voivat olla esimerkiksi laitteiden peukalointia, hajottamista tai niiden analysointia.

Laitteella olisi oleellista olla erilaisten fyysisten iskujen kestävä suoja ympärillään, jotta suojan alla olevien komponenttien kimppuun ei päästä, mutta esimerkiksi Immler ym. (2019) ehdottavat artikkelissaan kattavampaa suojausta. Iskunkestävyyden lisäksi suoja tarkkailee käynnistyksessä ja ajon aikana järjestelmän eheyttä peukaloinnin ja vahingon varalta. Eheyden rikkominen esimerkiksi tuhoaisi kryptografiset avaimet. Tällä voisi fyysisen vahingonteon lisäksi suojautua myös sivukanavahyökkäyksiltä. Lisäksi fyysisen hyökkäyksen, kuten kuoren poraamisen, seurauksena laitteen ehdotetaan tuhoutuvan käyttökelvottomaksi. Nämä keinot voisivat poistaa hyödyn laitteisiin hyökkäämisestä tai tehdä hyökkäämisestä vaikeaa hyökkääjälle.

Sivukanavahyökkäyksiä voi pyrkiä torjumaan myös muilla keinoilla. Maskaamalla (engl.

masking) voidaan piilottaa salausavaimista riippuvaa tietoa. Maskaustekniikoita on erilaisia, mutta niillä kaikilla on tarkoituksena satunnaistaa lähetettyä tietoa vaikealukuiseksi, kunnes tiedon maskaus poistetaan. Virrankulutusta voidaan myös pyrkiä satunnaistamaan erilaisilla keinoilla. Operaatioiden suoritusjärjestystä voidaan sekoittaa tai tavanomaisen datarekisterin sijaan voitaisiin hyödyntää kaksoisdatarekisteriä (engl. *secure double data rate register*). Tässä kahta rekisteriä käytetään niin, että jokaisella kellopulssilla toiseen rekisteriin tallennetaan operaatiossa syntynyt arvo ja toiseen satunnainen arvo (Liptak, Mal-Sarkar ja Kumar 2022). Edellä mainitut keinot ovat ohjelmistopohjaisia ratkaisuja sivukanavahyökkäyksiltä suojautumiseen. Kuitenkin turvallisuutta voi myös edistää myös laitteistopohjaisilla ratkaisuilla. Järjestelmässä voisi käyttää epädeterminististä prosessoria, josta löytyy suoritusjärjestystä satunnaistava ylimääräinen laitteiston osa. Myös järjestelmän kellotaajuus voidaan asettaa jatkuvasti muuttuvaksi, jolloin virrankulutus eri kellosykleillä tulee vaihtelemaan (Liptak, Mal-Sarkar ja Kumar 2022).

5 Yhteenveto

Turvallisten kyberfyysisten järjestelmien kehittäminen ei ole helposti saavutettavissa, vaan se vaatii kehitystä vielä monilla osa-alueilla. Järjestelmien turvalliseksi kehittämisessä onkin tärkeää ottaa huomioon ohjelmisto-, laitteisto- ja verkkotaso. Vain kokonaisvaltainen ote turvallisuuden kehittämiseen voi tehdä järjestelmistä todellisesti turvallisen. Tutkielman haasteita ja uhkia tarkasteltaessa oleelliseksi huomioksi nousee se, että hyökkäyspinnat ovat yhteydessä myös toisiin hyökkäyspintoihin. Laitteiston heikko suorituskyky ja vähäiset resurssit vaikuttavat suoraan ohjelmiston tietoturvalliseen suunnitteluun. Tämä heikosti suunniteltu ohjelmisto on altis uhille, jotka saattavat vaikuttaa esimerkiksi IoT-järjestelmässä oleviin muihin laitteisiin tai kokonaan verkon välityksellä toisiin laitteisiin ja verkkoihin.

Toinen oleellinen huomio on kuitenkin se, etteivät ratkaisut ole ilmaisia ja yksinkertaisia. Vaikka ratkaisuja on jokaisella hyökkäyspinnalla useita tarjolla, niillä on usein myös hintansa. Resurssipulaa voi helpottaa pilvi- tai sumulaskennalla, mutta tässä on myös varjopuolia. Pilvilaskennassa voi esiintyä latenssia erityisesti raskaammissa laskusuorituksissa, mutta pilvipalvelut, paikalliset sumulaskentalaitteet tai itse kyberfyysisen järjestelmän resurssien lisääminen kustantavat kaikki lisää rahaa. Samankaltaiset ongelmat ilmenevät myös muita hyökkäyspintoja turvatessa. Niin verkko-ominaisuuksia tarkkailevat mekanismit, turvallisempi suoja järjestelmälle ja yleinen turvallisemmin tehty sovelluskehitystyö vaativat aikaa ja rahaa.

Haasteista huolimatta aiheesta on tehty paljon tutkimusta ja tuoreitakin artikkeleita löytyy runsaasti monista kyberfyysisten järjestelmien osa-alueista. Esimerkiksi kyberfyysisten järjestelmien turvallisuudesta on tehty paljon tutkimusta ja monia teoreettisia ja prototyyppitasolla olevia ratkaisuja löytyi paljon. Tämä antaa toivoa tulevasta, jossa laitteiden runsaammat resurssit, pilvilaskentapalvelut, paremmat fyysiset suojat ja turvallisemmat ohjelmointikäytänteet mahdollistavat kyberfyysisten järjestelmien laajemman kehityksen ja hyödyntämisen yhä useammalla tavalla. Lisätutkimusta aiheesta voisi toteuttaa monesta näistä näkökulmista. Erityisesti ratkaisut uhkiin ovat hyödyllisiä tulevaisuuden kannalta ja niissä voisi keskittyä esimerkiksi sumulaskentaan, uudelleenisännöintiin tai turvallisen suunnittelun ajattelumalliin ohjelmistokehityksen näkökulmassa ja parempien suojausmekanismien kehittämiseen

laitteistokehityksen näkökulmassa.

Lähteet

Abbasi, Ali, Jos Wetzels, Thorsten Holz ja Sando Etalle. 2019. “Challenges in Designing Exploit Mitigations for Deeply Embedded Systems”. *2019 IEEE European Symposium on Security and Privacy,* https://doi.org/10.1109/EuroSP.2019.00013.

Ali, Zainab H., Hesham A. Ali ja Mahmoud M. Badawy. 2015. “Internet of Things (IoT): Definitions, Challenges and Recent Research Directions”. *International Journal of Computer Applications,* https://doi.org/10.5120/ijca2015906430.

Aloseel, Abdulmohsan, Hongmei He, Carl Shaw ja Muhammad Ali Khan. 2020. “Analytical Review of Cybersecurity for Embedded Systems”. *IEEE Access,* https://doi.org/10.1109/ACCESS.2020.3045972.

Amos, Brian. 2020. *Hands-on RTOS with Microcontrollers.* Packt. https://www.oreilly.com/library/view/hands-on-rtos-with/9781838826734/.

Antonakakis, Manos, Tim April, Michael Bailey, Matthew Bernhatd, Elie Bursztein, Jaime Cochran, Zakir Durumeric ym. 2017. “Understanding the Mirai Botnet”. *USENIX,* https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-antonakakis.pdf.

Atlam, Hany F., Robert J. Walters ja Gary B. Wills. 2018. “Fog Computing and the Internet of Things: A Review”. *Big data and cognitive computing,* https://doi.org/10.3390/bdcc2020010.

Atlam, Hany F. ja Gary B. Wills. 2019. “IoT Security, Privacy, Safety and Ethics”. *Springer Nature Switzerland,* https://doi.org/10.1007/978-3-030-18732-3_8.

Degani, Luca, Majid Salehi, Fabio Martinelli ja Bruno Crispo. 2023. “Software-Based Intrusion Prevention for Bare-Metal Embedded Systems”. *European Symposium on Research in Computer Security,* https://doi.org/10.1007/978-3-031-51482-1_16.

Dinculeanã, Dan ja Xiaochun Cheng. 2019. “Vulnerabilities and Limitations of MQTT Protocol Used between IoT Devices”. *Applied Sciences,* https://doi.org/10.3390/app9050848.

Fasano, Andrew, Tiemoko Ballo, Marius Muench, Tim Leek, Alexander Bulekov, Brendan Dolan-Gavitt, Manuel Egele ym. 2021. “SoK: Enabling Security Analyses of Embedded Systems via Rehosting”. *ASIA CSS,* https://doi.org/10.1145/3433210.3453093.

Fobe, Jean Luc Antoine Olivier, Michele Nogueira ja Daniel Macêdo Batista. 2022. “A New Defensive Technique Against Sleep Deprivation Attacks Driven by Battery Usage”. *Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais,* https://doi.org/10.5753/sbseg.2022.224911.

Fournaris, Apostolos P., Lidia Pocero Fraile ja Odysseas Koufopavlou. 2017. “Exploiting Hardware Vulnerabilities to Attack Embedded System Devices: a Survey of Potent Microarchitectural Attacks”. *Electronics,* https://doi.org/10.3390/electronics6030052.

Fysarakis, Konstantinos, George Hatzivasilis, Konstantinos Rantos, Alexandros Papanikolaou ja Charalampos Manifavas. 2014. “Embedded Systems Security Challenges”. *nternational Conference on Pervasive and Embedded Computing and Communication Systems,* https://doi.org/10.5220/0004901602550266.

Gebremichael, Teklay, Lehlogonolo Ledwaba, Mohammed H. Eldefrawy ja Gerhard P. Hancke. 2020. “Security and Privacy in the Industrial Internet of Things: Current Standards and Future Challenges”. *IEEE Access,* https://doi.org/10.1109/ACCESS.2020.3016937.

Guo, Chen, Song Ci, Yanglin Zhou ja Yang Yang. 2021. “A Survey of Energy Consumption Measurement in Embedded Systems”. *IEEE Access,* https://doi.org/10.1109/ACCESS.2021.3074070.

Hamza, Ayyoob, Hassan Habibi Gharakheili ja Vijay Sivaraman. 2020. “IoT Network Security: Requirements, Threats, and Countermeasures”. *arXiv,* https://doi.org/10.48550/arXiv.2008.09339.

Hassija, Vikas, Vinay Chamola, Vikas Saxena, Divyansh Jain, Pranav Goyal ja Piblab Sikdar. 2019. “A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures”. *IEEE Access,* https://doi.org/10.1109/ACCESS.2019.2924045.

Immler, Vincent, Johannes Obermaier, Kuan Kuan Ng, Fei Xiang Ke, Jin Yu Lee, Yak Peng Lim, Wei Koon Oh, Keng Hoong Wee ja Georg Sigl. 2019. “Secure Physical Enclosures from Covers with Tamper-Resistance”. *IACR Transactions on Cryptographic Hardware and Embedded Systems,* https://doi.org/10.13154/tches.v2019.i1.51-96.

Jassas, Mohammad, Jortin Mathew, Akramul Azim ja Qusay H. Mahmoud. 2017. “A Framework for Extending Resources of Embedded Systems using the Cloud”. *Canadian Conference on Electrical and Computer Engineering,* https://doi.org/10.1109/CCECE.2017.7946662.

Jurcut, Anca, Tiberiu Niculcea, Pasika Ranaweera ja Nhien-An Le-Khac. 2020. “Security Considerations for Internet of Things: A Survey”. *SN Computer Science,* https://doi.org/10.1007/s42979-020-00201-3.

Khelif, Mohamed Amine, Jordane Lorandel, Olivier Romain, Matthieu Regnery, Denis Baheux ja Guillaume Barbu. 2020. “Toward a hardware Man-in-the-Middle attack on PCIe bus”. *Microprocessors and Microsystems,* https://doi.org/10.1016/j.micpro.2020.103198.

Li, Yixiao, Yutaka Matsubara ja Hiroaki Takada. 2018. “A Comparative Analysis of RTOS and Linux Scalability on an Embedded Many-core Processor”. *Journal of Information Processing,* https://doi.org/10.2197/ipsjjip.26.225.

Liptak, Christopher, Sanchita Mal-Sarkar ja Sathish A.P. Kumar. 2022. “Power Analysis Side Channel Attacks and Countermeasures for the Internet of Things”. *IEEE Physical Assurance and Inspection of Electronics (PAINE),* https://doi.org/10.1109/PAINE56030.2022.10014854.

Marwedel, Peter. 2021. *Embedded System Design - Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things.* Springer. https://link.springer.com/book/10.1007/978-3-030-60910-8.

Or-Meir, Ori, Nir Nissim, Yuval Elovici ja Lior Rokach. 2019. “Dynamic Malware Analysis in the Modern Era—A State of the Art Survey”. *ACM Computing Surveys,* https://doi.org/10.1145/3329786.

O'Connor, TJ, Dylan Jessee ja Daniel Campos. 2021. "Through the Spyglass: Towards IoT Companion App Man-in-the-Middle Attacks". *Cyber Security Experimentation and Test Workshop,* https://doi.org/10.1145/3474718.3474729.

Olazabal, Alessanda Alvarez, Jasmeet Kaur ja Abel Yeboah-Ofori. 2022. "Deploying Man-In-the-Middle Attack on IoT Devices Connected to Long Range Wide Area Networks (LoRaWAN)". *IEEE International Smart Cities Conference,* https://doi.org/10.1109/ISC25536.2022.9922377.

Papp, Dorottya, Zhendong Ma ja Levente Buttyan. 2014. "Embedded Systems Security: Threats, Vulnerabilities, and Attack Taxonomy". *Annual Conference on Privacy, Security and Trust (PST),* https://doi.org/10.1109/PST.2015.7232966.

Sadeghi, Ahmad-Reza, Christian Wachsmann ja Michael Waidner. 2015. "Security and Privacy Challenges in Industrial Internet of Things". *IEEE Design Automation Conference,* https://doi.org/10.1145/2744769.2747942..

Salehi, Majid, Luca Degani, Marco Roveri, Danny Hughes ja Bruno Crispo. 2023. "Discovery and Identification of Memory Corruption Vulnerabilities on Bare-Metal Embedded Devices". *IEEE Transactions on dependable and secure computing,* https://doi.org/10.1109/TDSC.2022.3149371.

Wang, Zuoguang, Limin Sun ja Hongsong Zhu. 2020. "Defining Social Engineering in Cybersecurity". *IEEE Access,* https://doi.org/10.1109/ACCESS.2020.2992807.

Xiang, Yi ja Sudeep Pasricha. 2015. "Run-time Management for Multicore Embedded Systems With Energy Harvesting". *IEEE Transactions on very large scale integration (VLSI) systems,* https://doi.org/10.1109/TVLSI.2014.2381658.

Xie, Guoqi, Yuekun Chen, Renfa Li ja Keqin Li. 2017. "Hardware Cost Design Optimization for Functional Safety-Critical Parallel Applications on Heterogeneous Distributed Embedded Systems". *IEEE Transactions on industrial informatics,* https://doi.org/10.1109/TII.2017.2768075.

Yee, Chai Kay ja Mohamad Fadli Zolkipli. 2021. “Review on Confidentiality, Integrity and Availability in Information Security”. *Journal of ICT in Education (JICTIE),* https://doi.org/10.37134/jictie.vol8.2.4.2021.

Yekini, Nureni. 2022. “Overview of Embedded system it’s application”, https://www.researchgate.net/publication/361562662_OVERVIEW_OF_EMBEDDED_SYSTEM_ITS_APPLICATION.

Zhou, Xia, Jiaqi Li, Wenlong Zhang, Yajin Zhou, Wenbo Shen ja Kui Ren. 2022. “OPEC: Operation-based Security Isolation for Bare-metal Embedded Systems”. *EuroSys,* https://doi.org/10.1145/3492321.3519573.