Janne Summanen

Tietoturva SOAR-järjestelmässä

Integraatioyhteyksien turvaaminen

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintätekniikan tutkinto-ohjelma

Insinöörityö

29.11.2018

Metropolia

Tekijä Otsikko Sivumäärä Aika	Janne Summanen Tietoturva SOAR-järjestelmässä - Integraatioyhteyksien turvaaminen 42 sivua 29.11.2018
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	tieto- ja viestintätekniikka
Ammatillinen pääaine	Communication Networks and Applications
Ohjaajat	Lehtori Marko Uusitalo CDC Teknologianomistaja Teemu Takanen

Tämän työn tarkoituksena oli tutkia tietoturvaloukkauksien käsittelyn automatisoimiseksi ja orkestroimiseksi luodun Security Orchestration, Automation and Response -järjestelmän soveltuvuuden arvioimisprojektin yhteydessä selvittää tietoturvariskejä, joita kyseisen järjestelmän käyttöönotto organisaatiolle aiheuttaa. Työnaikana tutustutaan järjestelmiin, kuten SIEM, EDR, TI ja asiakaspalvelujärjestelmät, joita nykyaikainen SOC käyttää päivittäisessä työssään ja joihin SOAR-järjestelmä tulisi integroitumaan.

Työssä kerrotaan lyhyesti, mitä Security Orchestration, Automation and Response -järjestelmiltä odotetaan ja mihin ongelmiin SOAR-järjestelmän toivotaan tuovan ratkaisu. Näitä ongelmia ovat esimerkiksi jatkuva uhkakuvan monimutkaistuminen ja analyytikon työmäärän lisääntymin sekä monimutkaiset manuaaliset työtavat, jotka aiheuttavat palvelunlaadun heikentymistä.

Työssä perehdytään perinteisen uhkamallinnuksen keinoin tehtyyn arvioon siitä, mitä riskejä SOAR-järjestelmä tuo mukanaan, ja niihin keinoihin, joita organisaatiolla on käytettävissä riskinhallintaan. Työn alkuolettama on, että yritys on kiinnostunut SOAR-järjestelmästä ja suunnittelee sen mahdollista käyttöönottoa. Työn aikana on tarkoitus kehittää tietoturvallinen suunnitelma SOAR-järjestelmän käyttöönotolle sen integraatioyhteyksien turvallisuuden kannalta.

Työssä löydettiin useita potentiaalisia riskejä, jotka liittyvät SOAR-järjestelmän integraatioyhteyksiin ja kehitettiin niille mahdollisia tapoja vähentää ja hallita niitä. Työn lopputuloksena on alustava suunnitelma ja esimerkinomainen ympäristö, johon SOAR voitaisiin asentaa tietoturvallisesti.

Avainsanat	SOAR, tietoturva, SOC, tietoturvasuunnittelu, uhkamallinnus

Author	Janne Summanen
Title	Information security in SOAR. Securing Integrations.
Number of Pages	42 pages
Date	29 November 2018
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Professional Major	Communication Networks and Applications
Instructors	Teemu Takanen, CDC Technology Stack Owner Marko Uusitalo, Senior Lecturer

The goal of this thesis was, as a part of a bigger Security Orchestration, Automation and Response system evaluation project to asses and evaluate the information security risks SOAR would bring if deployed. The focus of this thesis is securing the integration connections between SOAR and other security systems.

This thesis briefly introduces the core systems like SIEMs, EDRs, threat intel platforms and customer service and ticketing systems that a modern SOC or CDC analysts use in their everyday work. These are the core parts a SOAR would need to be integrated if an organization chooses to move forward with such a system.

We look briefly into what a Security Orchestration, Automation and Response is and what are the problem this new technology is hoped to solve. These problems include the ever-increasing complexity of an information threat landscape that causes many problems as the work gets more and more complicated but at the same time the amount of manual and boring steps increase. There is also a real service quality problem because of outdated work technics and policies.

We map the information security risks associated with SOARs integration connections using some traditional threat modeling technics and developed ways to solve or mitigate these risks. The end goal was to map the risks and develop an information security policy along with some control methods to decrease the overall risk SOAR would bring to the company.

The result of this thesis was a plan how an organization could deploy a SOAR product securely. This includes a brief description and some hints of the needed technologies and services.

Keywords	SOAR, SOC, CDC, Information security, Threat modelling

Sisällys

Lyhenteet

AES	Advanced Encryption Standard. Edistynyt yleisesti tietoliikenteessä käytetty salausalgoritmi.
API	Application Programming Interface. Ohjelmointirajapinta jonka tarkoitus on antaa ohjelmallinen pääsy palvelun resursseihin.
CIRT	Cyper Incident Response Team. Tiimi jonka tehtävänä on reagoida tietoturvaloukkauksiin.
DMZ	Demilitarisoitu alue joka on verkon fyysinen tai looginen aliverkko joka yhdistää organisaation verkon turvattomampaan alueeseen.
DNS	Domain Name System. Internetin nimipalvelujärjestelmä yhdistää tekstimuotoiset domain osoitteet IP-osoitteiksi.
DoD	Department of Defence. Yhdysvaltojen hallinnon puolustusministeriö.
EDR	Endpoint Detection and Response. Kategoria työkaluja ja ratkaisuja jotka keskittyvät epäilyttävien tapahtumien havaitsemiseen ja niiden tutkimiseen päätelaitteilla.
FAIR	Factor Analysis of Information Risk. Matemaattinen menetelmä informaatioriskin määrittelemiseksi ja sen hallintaan.
HTPT	Havainnointi, tilanteenarviointi, päätös, toiminta.
HTTP	Hyper Text Transfer Protocol. Protokolla jolla selaimet ja palvelimet kommunikoivat keskenään nykyaikaisessa internetissä.
HTTPS	Hyper Text Transfer Protocol Secure. Kuten http-protokolla, mutta liikenne kulkee salattuna palveluiden välillä.
IDS	Intrusion Detection System. Ohjelmisto tai laitteisto jonka tarkoitus on havaita ja tunnistaa erilaisia hyökkäyksiä.

IoC	Indicator of Compromise. Merkki onnistuneesta hyökkäyksestä.
IP	Internet Protocol. Numerosarja jota käytetään uniikkina osoitteena verkkolaitteiden välillä identifioimaan järjestelmät toisistaan.
IPS	Intrusion Prevention System. Verkkohyökkäyksiä tunnistava ja estävä ohjelmisto tai laitteisto.
LDAP	Lightweight Directory Access Protocol. Verkkoprotokolla jonka yleisin käyttötarkoitus on käyttäjäntunnistus ja käyttöoikeuksien tarkistaminen.
LEF	Loss Event Frequency. Tulevaisuudessa tapahtuvan menetystapahtuman todennäköisyys.
LM	Loss magnitude. Riskin toteutuessa sen aiheuttaman menetyksen suuruus.
MITM	Man-in-the-middle. Hyökkäystekniikka jossa hyökkääjä tunkeutuu kahden viestijän väliseen viestintäreittiin yhtenä viestin välittäjistä ja kykenee tämän jälkeen muokkaamaan ja lukemaan yhteyden paketteja.
MSSP	Managed Security Services Provider. Tietoturvapalveluntarjoaja. Yritys tai yrityksen osa joka tarjoaa tietoturvapalveluita.
OSI	Open System Interconnection Reference Model. Malli joka kuvaa tietoliikenteessä siirtoprotokollien yhdistelmiä seitsemässä eri tasossa.
PERT	Program Evaluation and Review Technique. Menetelmä jolla voidaan suunnitella monimutkaisen projektin kulkua, kun etukäteen ei ole mahdollista arvioida tarkkoja yksityiskohtia.
SEM	Security Event Manger. Tosiaikainen tietoturvalokien valvontatyökalu joka korreloi lokeja ennalta määriteltyihin sääntöihin.
SIEM	Security Information and Event Management. Ohjelmisto, jonka tarkoituksena on yhdistää tietoturvalokeja useista eri lähteistä ja tuoda niiden perusteella rikkomuksia analysoitavaksi.

SIM	Security Information Management. Tietoturvalokienhallintatyökalu joka keskittyy lokienviestien varastoimiseen.
SIRP	Security Incident Response Platform. Ohjelmistoalusta tai tiimi tietoturvaloukkauksien tutkimiseen ja niihin vastaamiseen.
SOA	Security Orchestration and Automation. Yleisnimitys tekniikoille ja ohjelmistoille joilla automatisoidaan ja orkestroidaan tietoturvanhallintaa.
SOAR	Security Orchestration, Automation and Response. Järjestelmä, jonka tarkoituksena on kerätä ja tiivistää tietoturvahälytyksiä, mahdollistaa komentojen ajaminen yhdestä paikasta ja luoda tarkoituksen mukaisia raportteja tietoturvahavainnoista.
SOC	Security Operations Center. Tietoturvan valvontaan keskittynyt organisaation osa.
SSL	Secure Sockets Layer. Salausmenetelmä tietoliikenteen salaamiseksi IP-verkoissa.
SSO	Single Sign-on. Kirjautumisjärjestelmä, jossa käyttäjän tarvitsee kirjautua vain kerran käyttääkseen kaikkia saman organisaation järjestelmiä.
TEF	Threat Event Frequency. Riskiin liittyvän uhan todennäköinen tapahtumistiheys.
TIP	Threat Intel Platform. Uhkatietoa tarjoava alusta tai ohjelmisto.
TLS	Transport Layer Security. Salausprotokolla jolla suojataan Internet-sovellusten liikennettä IP-verkoissa.
USB	Universal Serial Bus. Sarjaväyläarkkitehtuuri laitteiden liittämiseksi tietokoneeseen.
VLAN	Virtual Local Area Network. Tapa jakaa verkkoja osiin kytkimen sisällä.

XSS	Cross Site Scripting. Nimitys tekniikoille, joilla sivustolle saadaan lisättyä koodia, joka ajetaan käyttäjän laitteistolla.

1 Johdanto

Tietovuotojen, verkkohyökkäysten ja muiden tietoturvaloukkausten määrän jatkuvasti kasvaessa nykyaikainen tietoturvavalvomo kohtaa ongelman. Hälytysmäärien kasvamisen takia kaikkia tietoturvahälytyksiä ei enää ehditä arvioimaan kunnolla, ja mahdolliset oikeat tietoturvaloukkaukset jäävät tämän takia tutkimatta. Henkilöstö väsyy jatkuvien sekä itseään toistavien ja helposti automatisoitavissa olevien tehtävien suorittamiseen.

Yksi ongelman ratkaisemiseksi esitetty teknologia on Security Orchestration, Automation and Response (SOAR). Tämän insinöörityön tarkoituksena on tarkastella SOAR-integraatioyhteyksien lisäämiä tietoturvaongelmia ja kehittää tietoturvallinen suunnitelma tietoturvaoperaatiokeskukselle mahdollisen SOAR-järjestelmän integraatioyhteyksien tietoturvan suojelemiseksi. Työ on tehty SOAR-tekniikan soveltuvuutta SOC-käyttöön arvioivan projektin osana. Työn lopputuloksena toivotaan olevan tietoturvallinen suunnitelma siitä, miten SOAR järjestelmä voitaisiin ottaa käyttöön siten, että se aiheuttaisi mahdollisimman pienen riskin organisaatiolle.

Koska SOAR integroituu useisiin SOC:n käytössä oleviin järjestelmiin, on työn alussa tarkoitus esitellä lukijalle nykyaikaisen tietoturvavalvomon toimintaa ja sen käyttämiä erilaisia järjestelmiä. Tämän jälkeen esitellään ratkaisuksi esitettyä SOAR-tekniikkaa. Työn aikana SOAR-järjestelmälle kehitetään tietoturvasäännöt ja arvioidaan integraatioyhteyksiin kohdistuvia riskejä. Työssä kehitetään ratkaisuja havaitun riskin pienentämiseksi ja lopuksi esitellään esimerkki siitä, miten SOAR-järjestelmä tietoturvallisesti otettaisiin käyttöön.

Työ on tehty suomalaisen tietoturvapalveluntarjoajan, Nixu Oyj:n, näkökulmasta, mutta työssä esitellyt tekniikat ovat silti myös normaalin organisaation tietoturvavalvomon käytettävissä. Työn ratkaisuja voidaan myös soveltaa pienempiin ja vähemmän monimutkaisiin ympäristöihin jättämällä pois useat eri SOAR-järjestelmät ja keskittymällä ratkaisuihin, jotka liittyvät työssä esiintyvän keskusinstanssin tietoturvaan.

2 Tietoturvaoperaatiokeskus

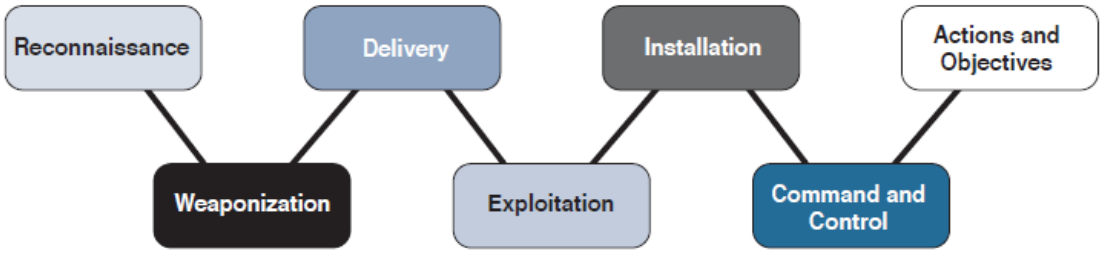
Koska tietoturvahyökkäykset ovat yhä monimutkaisempia ja niillä on yhä enemmän kehittyneitä vaikutuksia, täytyy modernin tietoturvapuolustuksen sisältää estävien toimien lisäksi tiedustelua ja monitorointia loukkauksien havaitsemiseksi. Onnistuneen hyökkäyksen kohteeksi joutuminen on enää vain ajan kysymys. Cisco Systemsin John Chambersin sanoin on olemassa kahden tyyppisiä yhtiöitä:ne jotka ovat joutuneet tietomurron uhreiksi ja ne jotka eivät vielä tiedä joutuneensa tietomurron kohteeksi. (1, s. 1.)

Kysymykset, joihin SOC organisaatiossa pyrkii tarjoamaan vastauksia, ovat esimerkiksi:

- Miten havaita tapahtunut tietomurto?
- Kuinka vakava tapahtunut tietomurto oli?
- Mitä vaikutuksia murrolla on organisaation toimintaan?

Yksi yleinen tapa käsitellä näitä kysymyksiä on alun perin sotilaalliseen puolustukseen kehitetty havainnointi, tilanteenarviointi, päätös, toiminta (HTPT) -silmukka, jonka on kehittänyt Yhdysvaltain ilmavoimien eversti John Boyd. HTPT pyrkii jatkuvan kehittymisen kautta käsittelemään puolustusta neljän askelman (havainnoi, tilannearvioi, päätä ja toimi) avulla. Tietoturvallisuuden ja kyberpuolustuksen yhteydessä näillä tarkoitetaan lokien ja tietovirtojen monitorointia, kerätyn datan analysointia, oikean toimintatavan määrittelemistä ja korjaavien tai estävien toimenpiteiden suorittamista. (1, s. 2-3.)

Jotta puolustus olisi tehokasta, täytyy ymmärtää hyökkääjä toimintatavat ja tavoitteet. Hyökkääjän tavoitteiden mallintamiseksi Lockheed Martinin Computer Incident Response Team (CIRT) on kehittänyt Cyper Kill Chain -mallin. Malli kuvaa hyökkääjän toimintaa seitsemän tavoitteen avulla, joista mitä tahansa häiritsemällä tai estämällä saadaan hyökkäys estettyä tai vähintään sen vaikutuksia lievennettyä.



Kuva 1. Cyper Kill Chain -malli. (1. s. 3)

Kuvan 1 mukaisen Cyper Kill Chainin vaiheet ovat tiedustelu, jossa hyökkääjä pyrkii keräämään julkisista lähteistä yhteystietoja, sosiaalisia yhteyksiä tai tietoa organisaation käyttämistä tekniikoista. Tämän jälkeen hyökkäykseen käytettävien työkalujen luomisesta, joka koostuu usein etäkäytön mahdollistavan troijalaisen ohjelmoimisesta. Kolmantena askelmana on luodun hyökkäystyökalun toimittaminen hyökättävään järjestelmään. Tämä tapahtuu usein sähköpostin liitetiedostona. Neljäntenä seuraa havaitun heikkouden hyväksi käyttäminen eli toimitetun hyökkäystyökalun ajaminen kohteessa. Viimeisenä hyökkääjä asentaa etäkäytön mahdollistavat työkalut ja luo komentokanavan. Tämän jälkeen hyökkääjä kontrolloi murrettua järjestelmää ja kykenee lopulta aloittamaan todelliset tavoitteensa, jotka usein sisältävät salaisen tiedon löytämisen yrityksen järjestelmistä ja sen vuotamisen organisaation ulkopuolelle. (1, s. 4-3.)

Tarve erilliselle ja kehittyneelle tietoturvaoperaatiokeskukselle on kasvanut uhkamaiseman monimutkaistuessa. Uudet automatisoidut hyökkäystyökaluja sisältävät jakelut kuten Kali- ja Backtrack-Linux mahdollistavat jopa tekniikasta tietämättömien hyökkääjien ajaa edistyneitä ja teknisesti monimutkaisia hyökkäyksiä nappia painamalla. Tämä on mahdollistanut tietomurtojen käyttämisen vaikuttamisen keinona yhä useammassa yhteydessä. (1, s. 6.)

2.1 Managed Security Services Provider (MSSP)

Managed Security Services Provider eli tietoturvapalveluntarjoaja tarjoaa ulkoistettua tietoturvaa yrityksille, joilla ei ole resursseja tai tietotaitoa kehittää omaa tietoturvavalvontakapasiteettiaan. Yleisiin palveluihin kuuluu tietoturvalaitteidenvalvonta ja -ylläpito, määräaikaisesti toistuvat haavoittuvuusskannaukset tai 24/7 SOC-valvonta. MSSP-palveluiden käyttäminen vähentää yrityksen tarvetta palkata omaa tietoturvahenkilöstöä ja

tarjoaa täten kustannussäästöjä ja helpottaa hyväksyttävän tietoturvatason ylläpitämistä. (2.)

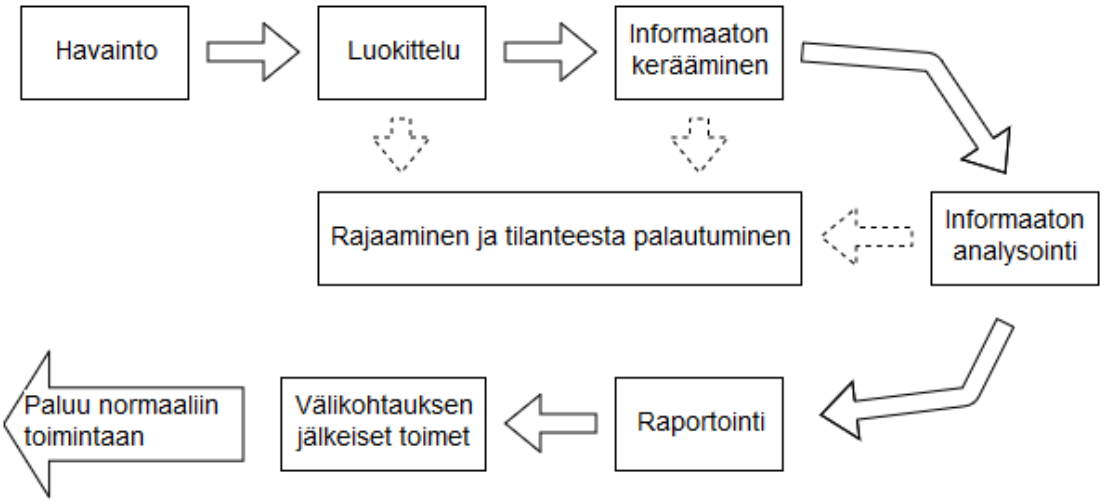
MSSP pyörittää usein hyvin kehittynyttä SOC-keskusta, joka kykenee havaitsemaan useita kehittyneitä hyökkäyksiä, jotka saattaisivat muuten jäädä organisaatiolta havaitsematta. Ulkoistamalla tietoturvan organisaatio kykenee keskittymään omaan ydinosaamiseensa samalla kun ulkopuolinen palveluntarjoaja lisää organisaation tietoturvantasoa, tuo ison tietoturvaorganisaation tietotaidon asiakkaansa käyttöön ja lisää asiakkaan näkyvyyttä heihin kohdistuviin tietoturvauhkiin. (3.) Usean asiakkaan SOC tarjoaa myös keskitettyä uhkatietoa, jolloin hyökkäyksiä kyetään ennaltaehkäisemään tehokkaammin, kun yhdessä ympäristössä kerättyä tietoa kyetään käyttämään toisen hyväksi.

MSSP:n SOC kohtaa myös omat uniikit haasteensa eri asiakkuuksien informaation pitämisessä erillään sekä pääsyn estämisessä asiakkaan ympäristöön toisen asiakkaan ympäristöstä. Verkkojen suuri määrä ja koko aiheuttavat myös omat haasteensa tietoturvan suunnittelulle. Lisäksi MSSP:n SOC-henkilöstö joutuu työskentelemään useissa eri yrityskulttuureissa, joka monimutkaistaa toimintaa entisestään.

2.2 Security Operations Centerin toiminta

Tietoturvaloukkauksien havaitseminen ja niihin reagoiminen on SOC:n pääasiallinen tehtävä. Tietoturvavalvonnasta vastaavan tiimin odotetaan monitoroivan organisaation ympäristöä ja reagoivan tietoturvaan liittyviin tapahtumiin ja loukkauksiin. Tämä sisältää mahdollisten tietomurtoindikaattorien (IoC) havaitsemisen ja tutkimisen. IoC:t ovat teknisiä tai ei-teknisiä merkkejä siitä, että ympäristössä on tapahtunut tietoturvaloukkaus. IoC voi olla esimerkiksi havainto siitä, että käyttäjä on laittanut USB-tikun kiinni tietokoneeseen ympäristössä, jossa kyseisten laitteiden käyttäminen on kiellettyä. (1, s. 14.)

Tyypillinen prosessi, jonka SOC käy läpi tietoturvaloukkauksen tutkinnassa, on esitelty kuvassa 2.



Kuva 2. Tietoturvahäiriöiden hallinta. (1, s.15.)

Prosessi alkaa tietoturvahäiriön havaitsemisesta. Tässä vaiheessa häiriön on raportoinut henkilö tai siitä on saatu hälytys teknisin keinoin. Prosessin alussa on tärkeätä tunnistaa, mistä hälytys on tullut, mitä kautta ja miten se olisi pitänyt raportoida. Luokitteluvaiheen tarkoitus on tunnistaa tarvittavat seuraavat askeleet häiriön hoitamiseksi. Luokitteluvaiheessa häiriö todennetaan, luokitellaan ja se nimetään jonkun hoidettavaksi. Informaationkeräämisen jälkeen tietoturvahäiriö analysoidaan, jotta se voidaan kategorisoida ja sen vakavuus arvioida. Toimet tietoturvahäiriön rajaamiseksi ja siitä palautumiseksi aloitetaan mahdollisimman ajoissa. Rajaaminen voi sisältää konfiguraatiomuutosten tekemistä, käyttäjätilien salasanojen nollaamista ynnä muuta. Tärkeätä on, että häiriön voidaan jossain vaiheessa todeta olevan ohi ja toiminnan palautuneen normaaliksi. Tämän jälkeen tilanne raportoidaan vielä kattavasti. Häiriön ratkaisemisen jälkeen ennen normaaliin toimintaan siirtymistä hoidetaan vielä häiriön jälkeiset toimet. Tämä viimeinen vaihe sisältää toimien tehokkuuden arvioinnin ja arvion siitä, mitä olisi voitu tehdä paremmin. Näin SOC-tiimi kehittyy jokaisen hälytyksen myötä hieman paremmaksi. (1, s. 15-20.)

Nykyaikainen SOC-keskus on usein hierarkkisesti jaettu neljään tasoon, jossa ensimmäisellä tasolla työskentelevän analyytikon pääasiallinen tehtävä on tutkia SIEM-järjestelmän (Security Event and Information Management) antamia hälytyksiä ja nopeasti todeta, onko kyseinen hälytys mahdollinen indikaattori hyökkäyksestä vai aiheetonhälytys. Mahdolliset hyökkäyksestä kertovat tapahtumat siirretään tason kaksi analyytikolle, jotka

hoitavat tapahtumien pääasiallisentutkinnan ja määrittelevät, onko hälytyksen perusteella syytä olettaa, että on tapahtunut hyökkäys ja tarvitseeko se vastatoimenpiteitä. Kolmannella tasolla toimivat henkilöt ovat yleensä erittäin kokeneita ja usein hyvinkin erikoistuneita tietoturva-alan ammattilaisia. Heidän tehtävänsä koostuvat todellisten tietoturvaloukkausten rikosteknisestätutkinnasta sekä aitojen hyökkäysten vaikutusten minimoimisesta erilaisin vastatoimin. Tämän lisäksi he suorittavat haavoittuvuuksien ja uhkien aktiivista metsästämistä valvottavissa ympäristöissä. SOC:n neljännen tason muodostaa johtohenkilöstö, joka huolehtii SOC:n tehokkaasta toiminnasta, sen jatkuvasta kehittämisestä ja tehokkaasta ja tarpeenmukaisesta asiakaskommunikaatiosta. (4.)

2.3 SOC:n käytössä olevat tekniikat

Koska SOC-keskuksien alkuaikoina valvottavia lokilähteitä oli huomattavasti vähemmän kuin nykyisin käyttivät analyytikot eri tuotevalmistajien omien tuotteiden kuten tunkeutumisenhavaitsemisjärjestelmän omia konsoleita. Nykyisin erilaisia lokilähteitä on kuitenkin niin suuri määrä, että edes kaikkein kokeneimmankaan analyytikon ei ole mahdollista havainnoida tarpeeksi suurella otannalla tapahtumia käyttämällä pelkästään tuotteiden omia konsoleita. Tämän takia SOC onkin ottanut käyttöön SIEM-järjestelmän, jonka tarkoituksena on kerätä kaikki lokilähteet yhteen paikkaan ja nostaa niistä analyytikoille hälytyksiä ennalta määrätyin perustein. Jotta hälytyksiin voitaisiin reagoida nopeasti ja tehokkaasti on SOC:lla usein myös käytössä jokin Endpoint Detection and Response -järjestelmä.

Hälytyksiin reagoimisen lisäksi hyvin tärkeäosa SOC-keskuksen tehtäviä on kerätä jatkuvaa uhkatietoa. Tätä varten keskuksella on usein järjestelmiä, jotka keräävät automaattisesti eri lähteistä saatavilla olevaa uhkatietoa yhteen paikkaan, jossa se on helposti analyytikon käytettävissä.

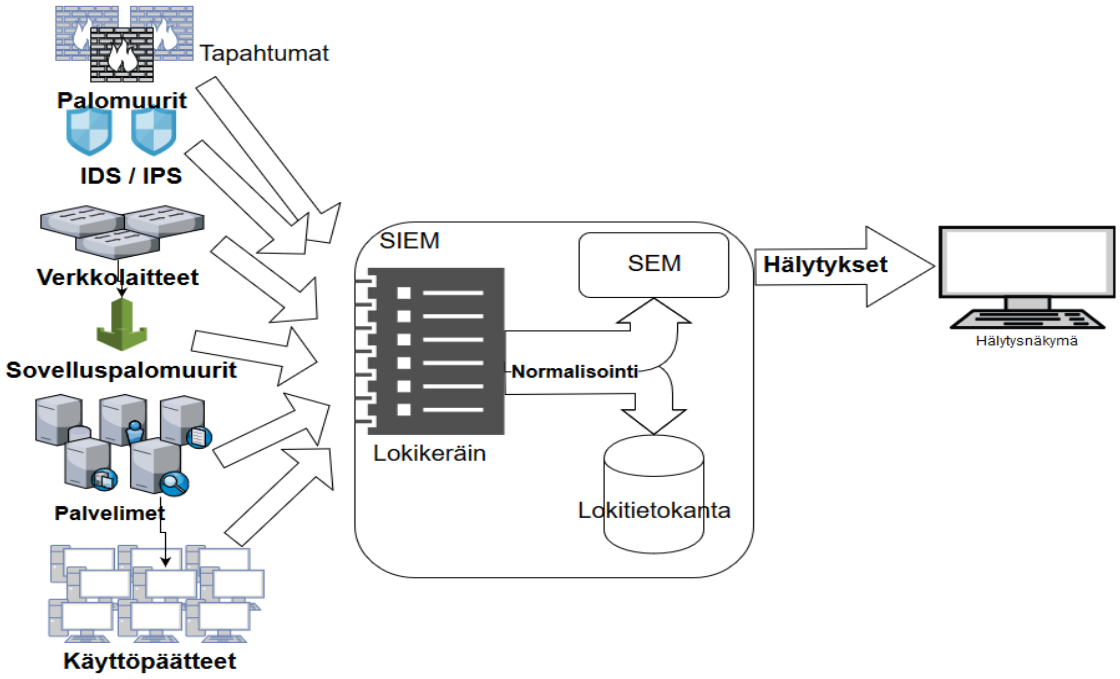
Kaiken tämän lisäksi tehokas kommunikointi organisaation muiden osien tai muiden organisaatioiden välillä on kriittisen tärkeätä. Tämän takia SOC:lla on usein käytössä jokin asiakaspalvelujärjestelmä, joka kasaa jokaiseen tapahtumaan liittyvät tapahtumat ja viestit yhteen paikkaan.

2.3.1 Security Information and Event Management -järjestelmät

Nykyaikainen organisaation verkkopuolustus muodostuu useista eri järjestelmistä. Verkossa on palomuureja, joiden tarkoitus on estää yhteyksiä, ja erilaisia välityspalvelimia joiden tarkoituksena on suodattaa liikennettä tai estää pääsyä eri sivustoille. Organisaatiolla on usein myös IDS- ja IPS-järjestelmiä, jotka pyrkivät havaitsemaan ja estämään mahdollisia hyökkäyksiä, sekä Web Application -palomuureja, jotka analysoivat www-palvelimille tulevia yhteyksiä mahdollisten hyökkäysten havaitsemiseksi. Organisaation verkossa on myös DNS-palvelimia ynnä muita normaaleja verkon toimivuuden kannalta elintärkeitä palveluita. Lisäksi sisään tulevia ja ulos lähteviä sähköposteja skannataan mahdollisten verkkokalastelun, haittaohjelmien ja tietovuotojen havaitsemiseksi. Työasemille ja palvelimille on asennettu erilaisia antivirusratkaisuja ja mahdollisesti Endpoint Detection and Response (EDR) -tuotteita. Lisäksi järjestelmälokeja valvotaan omituisten tapahtumien löytämiseksi. Lienee siis olevan selvää, että eri lähteistä tulevaa lokia ja hälytystä on niin paljon, ettei sitä ilman erillistä järjestelmää pystytä analysoimaan tai varastoimaan järkevästi.

Apuun tähän lokien hallinnan painajaiseen tulevat SIEM-järjestelmät, jotka ovat yhdistelmä tietoturvalokienhallinta- (SEM) ja tietoturvainformaationhallintajärjestelmistä (SIM). SIM-järjestelmien tarkoituksena on kerätä, varastoida ja analysoida lokidataa. SEM-järjestelmien tarkoitus on analysoida ja monitoroida reaaliaikaisesti samoja lokilähteitä. SIEM-järjestelmä yhdistää nämä kaksi lokienhallinta- ja analysointijärjestelmää ja lisää päälle vielä useiden lokilähteiden reaaliaikaisen korreloinnin toisiinsa. (5, s. ix.)

SIEM-järjestelmä kerää siis lokiviestit useista eri lähteistä yhteen paikkaan ja varastoi ne. Lisäksi SIEM korreloi erilähteistä tulleita lokeja luodakseen paremman kokonaiskuvan tapahtuneesta. Tyypillinen SIEM-järjestelmän arkkitehtuuri on esitelty kuvassa 3.



Kuva 3. Tyypillinen SIEM-arkkitehtuuri SOC-käytössä. (6 s. 36)

Kuten kuvasta 3 käy ilmi, SIEM-järjestelmä kerää ensin lokiviestit useista eri lähteistä ja normalisoi ne siten, että lokiviestien sisältämä tieto on helposti haettavissa sekä loogisesti samassa järjestyksessä. Tämän jälkeen SIEM-järjestelmä analysoi ja korreloi lokitietoa ja pyrkii keräämään toisiinsa liittyvät tapahtumat yhdeksi kokonaisuudeksi. Lopulta tapahtumia verrataan ennalta määriteltyihin sääntöihin ja niiden perusteella tapahtumista luodaan tarvittaessa hälytyksiä. (6, s. 36.)

SIEM korreloi useiden eri lokilähteiden tapahtumia etsimällä tunnettuja malleja sisään tulevasta lokivirrasta. Korrelointi voi esimerkiksi yhdistää http-välityspalvelimen ja virustorjuntaohjelmiston ja täten huomata, kun haittaohjelma ladataan käyttäjän koneelle. Korreloinnin ansiosta hälytyksestä on heti nähtävissä, mistä mahdollinen haittaohjelma on peräisin. Korrelointi perustuu usein menneisyydessä kerättyyn kokemukseen ja tietoon sekä heuristiseen analyysiin. Tämä aiheuttaa SIEM-järjestelmissä ongelman sääntöjen säätämisen suhteen. Korrelointi ja analysointi sääntöjen virittäminen on jatkuva prosessi havaintojen ja todellisten loukkauksien suhteen optimoimiseksi. Lisäksi vielä eilen toiminut sääntö ei enää tänään välttämättä toimi, koska hyökkäyskuvio on saattanut muuttua.

2.3.2 Endpoint Detection and Response -järjestelmät

EDR-järjestelmät ovat nouseva teknologia, jonka Gartnerin Anton Chuvak määritteli työkaluiksi, jotka keskittyvät ensisijaisesti havaitsemaan ja tutkimaan epäilyttäviä tapahtumia ja niiden mahdollisia jälkiä sekä muita ongelmia päätelaitteissa. EDR-tuotteet vastaavat tarpeeseen valvoa päätelaitteita edistyneidenhyökkäyksien varalta ja mahdollistavat vastatoimien aloittamisen keskitetysti yhdestä hallintapisteestä. (7.)

EDR-työkalut toimivat monitoroimalla päätelaitteen tapahtumia, kuten ajossa olevien prosessien muutoksia tai uusien verkkoyhteyksien muodostamista. Nämä tapahtumat tallennetaan ja yleensä siirretään pilvipalveluun tarkempaa analyysia varten. Kaiken tämän toiminnan pohjana toimii päätelaitteelle asennettava agentti. Päätelaitteelle asennettu agentti mahdollistaa yleensä myös laitteen etähallinnan keskitetystä hallintapaneelista. (7.)

EDR-työkalut ovat tärkeitä, koska jokainen verkkoon kytketty laite muodostaa potentiaalisenhyökkäysvektorin organisaation verkkoon. EDR-tuotteet auttavat työasemien valvomista niissä tapauksissa, kun normaalit virustorjuntaohjelmistot eivät huomaa hyökkäystä. Perinteiset virustorjuntaohjelmistot perustuvat merkkeihin tunnetuista pahoista ohjelmista ja mahdollisiin heuristisiin analyyseihin, kun taas EDR-tuotteet pyrkivät havaitsemaan työasemalla tapahtuvaa epäilyttävää tai normaalista poikkeavaa toimintaa. Tämän ansioista EDR-tuotteet pystyvät havaitsemaan kehittyneempiä ja kohdistetumpia hyökkäyksiä jotka pyrkivät kiertämään perinteisten virustorjuntaohjelmistojen havainnointimenetelmiä. (8.)

Alan johtava tuote on Gartnerin tutkimusten mukaan Carbon Black (9). Carbon Black:n mukaan EDR-tuotteiden tärkeimpiä ominaisuuksia ovat kattavan ja yhtenäisen datan tarjoaminen, ekspansiivisen näkyvyyden tarjoaminen ympäristöön, mahdollisuus reaaliaikaisiin vastatoimiin sekä integroitavuus muihin tietoturvatyökaluihin (10).

Tarpeeksi kattava data siitä, mitä työasemilla tapahtuu, mahdollistaa analyytikoiden havaita tietoturvaloukkauksia ja vastata niihin ennen kuin on liian myöhäistä estää vaurioita. Kattava näkyvyys auttaa havaitsemaan edistyneet, hitaasti etenevät ja normaalit automaattisethavainnointimenetelmät kiertävät hyökkäykset. Mahdollisuus reaaliaikaisiin vastatoimiin nopeuttaa vahinkojen minimoimista, hyökkäykseen vastaamista ja tilan-

teen tuomista takaisin organisaaton hallintaan. Integroimalla useat tietoturvatuotteet yhteen pystyvät analyytikot korreloimaan tapahtumia ja muodostamaan paremman kokonaiskuvan hyökkäyksestä. (10.)

EDR-tuotteet tuovat siis SOC:lle paljon kaivattua näkyvyyttä työasemille ja nopeuttavat jo tapahtuneiden tietoturvaloukkausten tutkintaa. Nykyaikainen palveluna myytävä SOC tuskin toimisi yhtä tehokkaasti ilman näitä tuotteita.

2.3.3 Uhkatieto ja uhkatietojärjestelmät

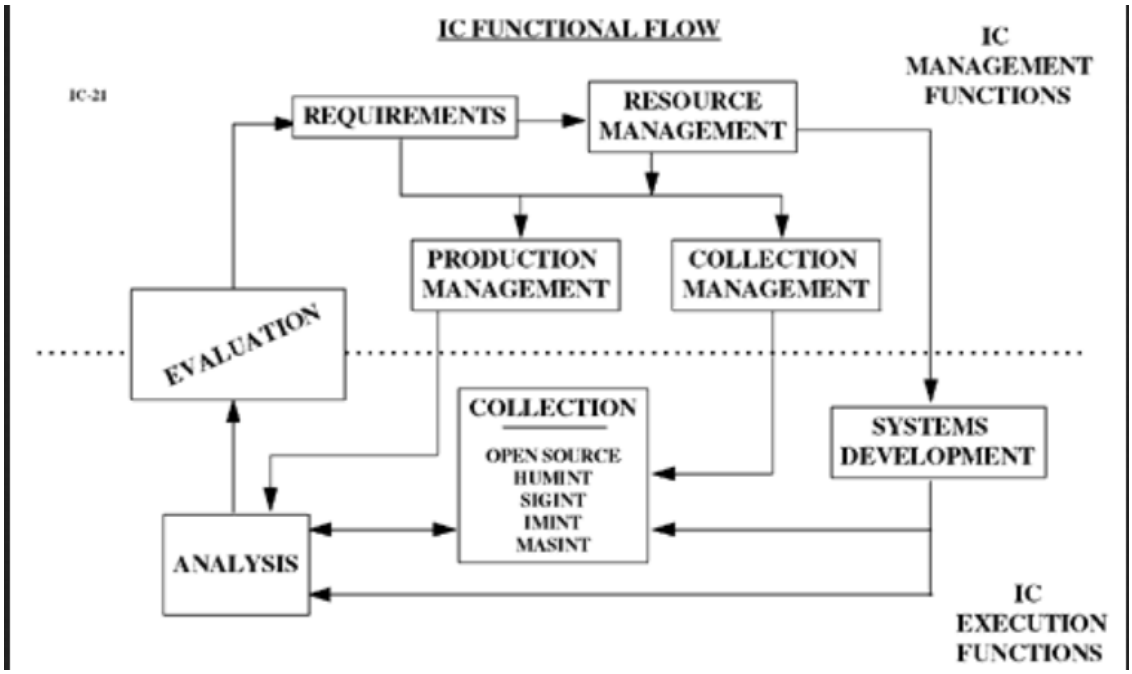
Uhkatieto on kokoelma todistusaineistopohjaista tietoa olemassa olevista ja potentiaalisista uhkista. Uhkatieto koostuu usein mekanismeista ja indikaattoreista tietyssä asiayhteydessä, niiden vaikutuksista ja joskus myös suositelluista ehkäisevistä toimenpiteistä. Uhkatiedon tärkein tehtävä on avustaa uhkan tunnistamisessa ja siihen reagoimisessa. (11.)

Uhkatiedon voidaan katsoa koostuvan seuraavista korkeantason käsitteistä: hyökkääjien identiteetistä, motivaatiosta, tavoitteista sekä mahdollisesta strategiasta. Toisaalta uhkatieto sisältää myös tietoa hyökkääjien käyttämistä taktiikoista, tekniikoista ja menetelmistä eli siitä, miten hyökkääjät tietyllä ajanhetkellä toimivat. Uhkatieto sisältää myös tietoa hyökkääjien käyttämistä työkaluista ja niistä merkeistä joita aktiivinen hyökkäys jättää jälkeensä sekä usein vain lyhyenajan tuoreena pysyvistä indikaattoreista kuten IP-osoitteista, DNS-nimistä tai tiedostojentiivistetiedoista. Tämän lisäksi uhkatieto sisältää myös tiedon hyökkääjien sen hetkisistä aktiivisista kohteista. (11.)

Uhkatiedon pääasiallinen tavoite on kuten perinteisen tiedustelutiedonkin, tehdä tuntemattomista tuntemattomista uhkista joko tunnettuja tuntemattomia tai tunnettuja tunnettuja uhkia (12). Tuntemattomalla tuntemattomalla, kuten Donald Rumsfeld on asian ilmaissut DoD:n selonteossa 2002, tarkoitetaan tilannetta, jossa emme tiedä, mitä emme tiedä, eli meillä ei ole edes tietoa siitä, että uhkaa ylipäätänsä on olemassa (12). Esimerkiksi joku kehitystiimistä on avannut kehityspalvelimen portteja internetiin eikä meillä ole siitä tietoa. Täten emme edes ole tietoisia siitä, että tätä palvelinta vastaan voidaan hyökätä. Tämän kaltaisia uhkia vastaan puolustautumisen voidaan katsoa olevan käytännössä mahdotonta. Tunnetulla tuntemattomalla tarkoitamme tilannetta, jossa olemme havainneet uhkan olemassaolon, mutta emme vielä tiedä tarkalleen, mitä se sisältää

(12). Tunnettu tunnettu uhka on uhka, jonka tiedostamme olevan olemassa ja jonka myös ymmärrämme hyvin (12). Esimerkiksi voimme olla tietoisia siitä, että palveluun kirjautumisen täytyy olla auki kaikille ja tämän takia siihen todennäköisesti kohdistuu salasanan arvaushyökkäyksiä, joita voimme hidastaa monimutkaisilla salasanoilla ja estää lukitsemalla tilejä, jos niille kirjautumisessa epäonnistutaan liian monesti.

Hyvänä pohjana kypsälle uhkatietosuunnitelmalle voidaan käyttää esimerkiksi Yhdysvaltojen senaatin Select Committeen vuonna 1996 ehdottamaa prosessikaaviota siitä miltä tiedustelutiedon käsittely näyttäisi, 2000-luvulla (12). Prosessikaavio on esitelty kuvassa 4.



Kuva 4. Tiedustelutiedonkäsittelyjakson prosessikaavio. https://www.gpo.gov/fdsys/pkg/GPO-IC21/html/figure1a.gif

Tiedustelutiedon kuten myös uhkatiedon kerääminen alkaa vaatimusten määrittelystä. Tässä tärkeässä askelmassa määritellään se, mistä organisaatio on kiinnostunut ja mitä uhkatietojärjestelmän tulisi tarjota. Vaatimusten määrittelyn jälkeen seuraava tärkeä askel on tiedustelutiedon kerääminen. Tämä tapahtuu usein keräämällä tietoa useista eri lähteistä, kuten muut uhkatietosyötteet, keskustelufoorumit, sosiaalisenmediansivustot ja niin edelleen. Uhkatietoa voidaan kerätä myös suoraan työskenteleviltä analyytikoilta. Kerätty tieto täytyy tämän jälkeen analysoida, jotta siitä saadaan hyötyä. Lopulta kerätyn uhkatiedon laatua täytyy vielä arvioida. Tämän jälkeen siirrytään joko suoraan takaisin

tiedon keräämiseen tai vaihtoehtoisesti määrittelemään uhkatiedon vaatimukset tarkemmin. (12.)

Uhkatietojärjestelmien tärkeimpiä ominaisuuksia ovat sen kyky kerätä automaattisesti valtavat määrät tietoa eri lähteistä ja normalisoida se siten, että siitä syntyvästä tietokannasta voidaan tehokkaasti hakea sekä korreloida tietoa. Lisäksi Uhkatietojärjestelmän helppo integroiminen muihin tietoturvajärjestelmiin tiedon levittämiseksi ja sen käyttäjälle tarjoamien työkalujenlaatu ja -helppokäyttöisyys tiedon analysoimisessa on tärkeätä. (13.)

2.3.4 Asiakaspalvelu- ja tehtävienhallintajärjestelmät

Ylläpitääkseen ja seuratakseen sopimuksissa luvattua palvelutasoa sekä sen toteutumista SOC tarvitsee jonkin ohjelmiston tai järjestelmän. Tämän järjestelmän tulisi samalla tehostaa kommunikointia asiakkaan suuntaan, helpottaa tehtävienseurantaa ja dokumentoida tehokkaasti jokaisen tutkinnan erivaiheet.

SOC-tiimin oletetaan seuraavan ja dokumentoivan kaikki potentiaaliset tietoturvaloukkaukset sekä niiden tutkinnan. Tämän takia tutkintaa tarvitsevista loukkauksista luodaan yleensä tapaus tehtävienhallintajärjestelmään, jossa se voidaan osoittaa tietyn henkilön hoidettavaksi ja jossa sitä voidaan seurata loppuun asti. SIEM-työkalujen, haavoittuvuus skannereiden ynnä muiden SOC:n työkalujen tulisi integroitua yrityksen jo olemassa oleviin asiakaspalvelu- ja tehtävienhallintajärjestelmiin. Mikäli tämä ei ole mahdollista, täytyy työkalujen itsensä toteuttaa tarvittavat ominaisuudet. (1, s. 64.)

Tärkeä asia huomioida järjestelmää tai järjestelmiä valittaessa on, että osaa tutkinnoista ei välttämättä pystytä hoitamaan täysin SOC:n sisäisesti ja että tutkinnan sidosryhmät saattavat olla kiinnostuneita tutkinnan etenemisestä. Tämän takia tapauksien osoittaminen organisaation ulkopuoliselle osapuolelle ja tapauksien vaiheista kommunikointi asiakkaan suuntaan on oltava helppoa. Lisäksi järjestelmän tulisi tarjota SOC:lle yhteistyöalusta, johon tiiminjäsenet voivat tallentaa laitteiden ja ohjelmistojen käyttöohjeet, toimintaohjeet ja muut yhteiset dokumentit kaikkien saataville. (1, s. 64-65.)

2.4 Ongelmat kun SOC laajentuu tarpeeksi

Nykyisin useat tietoturvatiimit kamppailevat, uhkia havaitsevien järjestelmien lisääntyessä, pysyäkseen mukana jatkuvasti kasvavassa hälytystulvassa. SOC:n toiminta perustuu myös usein edelleen pääasiassa perinteisiin manuaalisiin dokumenttipohjaisiin-toimintatapoihin, jotka hidastavat hälytystenkäsittelyä. Usein käytännöt ja tietotaivo pysyvät yksittäisten analyytikoiden päässä vaikeuttaen uusien työntekijöiden kouluttamista ja lisäävät aivovuodon riskiä. (14.)

Hyökkääjien kehittäessä menetelmiään ohittaa perinteisiä hyökkäyksenesto- ja havainnointimenetelmiä ja loppukäyttäjien edelleen, kuten aikaisemminkin, joutuessa sosiaalisen manipulaation uhreiksi on monen yrityksen hyökkäysrajapinta laajentunut huomattavasti. Tämän seurauksena useat tietoturvatiimit ovat yli-investoineet useisiin kehittyneisiin hyökkäyksenhavaitsemis- ja estämisjärjestelmiin (14). Tämä aiheuttaa tietoturva-analyytikoille päivittäisessä työssä niin kutsuttua hälytys- ja työkaluväsymystä, joka johtaa huonompaan tehokkuuteen ja turhiin virheisiin.

Vaikka aika hyökkäyksestä sen havaitsemiseen on jatkuvasti vähentynyt ei se edelleenkään ole tarpeeksi nopealla tasolla. Varsinkin uudentyyppiset aggressiivisemmat hyökkäystyypit jotka pyrkivät tuhoamaan ja vuotamaan immateriaalista omaisuutta sekä kiristämään yrityksiä vaativat yhä nopeampaa ja tehokkaampaa selkkauksiin puuttumista. (14.)

Laajeneva SOC kohtaa siis ongelman - vaadittava tietotaito kasvaa ja osaavaa henkilöstöä ei ole tarpeeksi saatavilla. Hyökkäykset tulisi havaita nopeammin ja niihin pitäisi kyetä reagoimaan tehokkaammin, joka puolestaan aiheuttaa henkilöstön väsymistä. Tästä seuraa suuri henkilöstönvaihtuvuus, joka aiheuttaa osaamisen ja käytäntöjen vuotamista ulos organisaatiosta. Varsinkin isoissa ja nopeasti kasvavissa yrityksissä tätä ongelmaa pyritään ratkaisemaan lisäämällä automatisaatiota ja orkestrointia.

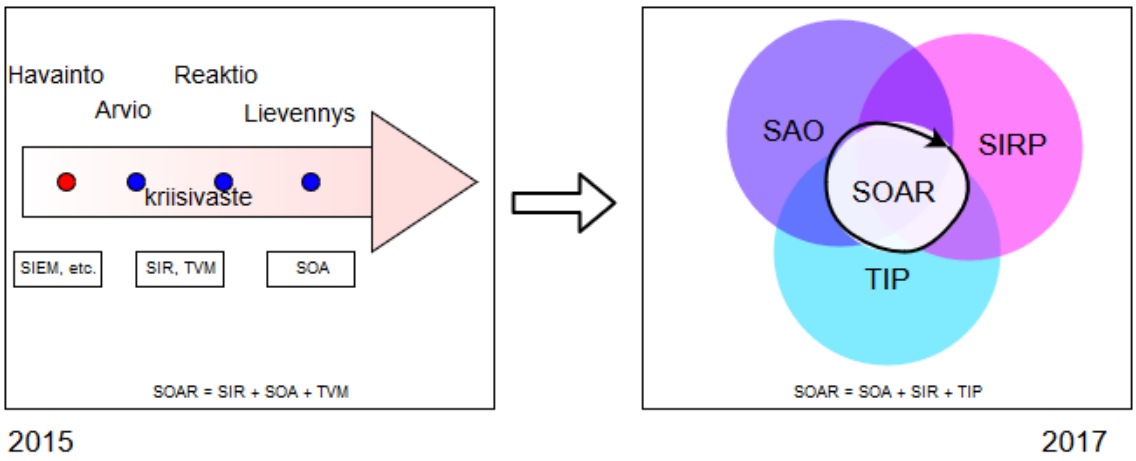
Orkestroinnin ja automatisaation tavoitteena on luoda järjestelmä, jossa yhdestä paikasta voidaan selkein ja tehokkain metodein havaita ja tutkia hälytyksiä. Yhdistelemällä tietoa useista eri järjestelmistä luoda kattava kuva hälytyksensyistä ja niihin mahdollisesti liittyvistä aikaisemmista toimista ja hälytyksistä. Automatisoida itseään toistavat selvät

toiminnot työtyytyväisyyden ja -tehokkuuden nostamiseksi sekä kerätä kaikki hälytykseen liittyvä tieto yhteen paikkaan helpomman raportoinnin ja seurannan mahdollistamiseksi.

3 Security Orchestration, Automation and Response

Security Orchestration, Automation and Response tai lyhyemmin SOAR määritellään Gartnerin artikkelissa teknologiaksi, joka mahdollistaa tietoturvauhkatietojen ja -hälytysten keräämisen eri lähteistä yhteen keskitettyyn näkymään. SOAR-järjestelmän avulla tapahtumien tutkiminen ja luokittelu voidaan tehdä tehokkaammin yhdistämällä sekä ihmis- että koneälyä. SOAR auttaa myös määrittelemään, priorisoimaan ja standardisoimaan tietoturvaloukkauksiin reagoimiseen liittyviä toimintamalleja ja käytäntöjä. (14.)

Aikaisemmin, vuonna 2015, SOAR-termi määriteltiin hieman eri tavalla ja nimellä Security Operations, Analytics, and Reporting. Silloinen tekniikka määriteltiin siten, että sen tarkoituksena oli tarjota analysointi-, raportointi- ja hallintaominaisuuksia hyväksikäyttäen koneluettavaa dataa. Markkinan kehittyessä termiä jouduttiin kuitenkin uudelleenmäärittelemään, kun aiemmin erillään olleet kolme pienehköä teknologiaa Security Orchestration and Automation (SOA), Security Incident Response Platform (SIRP) ja Threat Intel Platform (TIP) -järjestelmät yhdistyivät yhdeksi isoksi kokonaisuudeksi kuvan 5 mukaisesti. (14.)



Kuva 5. SOAR-määritelmän kehitys.

SOAR määritellään siis niiden toimintojen kautta, joita se SOC:n käytössä suorittaa. Näitä ovat tietoturvan orkestrointi ja automatisointi (SOA) ja tietoturvaloukkauksiin vastaamiseen keskittyvät järjestelmät ja niistä raportointi (SIRP). Lisäksi SOAR tuo uhkatietopalvelut (TIP) automaattisesti helposti saataville ja tarvittaessa helposti käytettäväksi tietoturvaloukkauksen tutkintaan liittyvissä toimissa.

Tietoturvan orkestroinnilla tarkoitetaan kykyä koordinoida päätöksentekoa sekä formalisoida ja automatisoida toimia perustuen riskin suuruuteen ja ympäristön sen hetkiseen tilaan. SOAR kerää hälytykset useista eri lähteistä, arvioi niiden prioriteetin ja koordinoi koko loukkaukseen reagoimisprosessin. Lopuksi SOAR mittaa prosessin tehokkuuden ja auttaa luomaan tarvittavat raportit tapahtumasta. Tutkinnanformalisointi tarkoittaa toimien ennalta määrittämistä jokaiselle loukkaustyypille. Tämä parantaa reaktion laatua ja nopeuttaa toimintaa, kun aikaa ja vaivaa ei kulu turhaan miettimiseen. Automatisointi kerää kaiken tarpeellisen informaation useasta eri lähteestä samaan paikkaan ja kokoaa samaan näkymään myös tietoturvapoikkeamaan liittyvät aikaisemmat tapahtumat. Tämä helpottaa tietoturvapoikkeaman analysointia, kun aikaa ei enää kulu työkalusta toiseen liikkumiseen. (14.)

Automatisoinnilla pyritään hoitamaan helposti ennalta määriteltävät tehtävät ennen kuin tutkintaa edes näytetään ihmiselle. Automatisointi nopeuttaa vasteaikaa tekemällä helposti automatisoitavia tehtäviä ja väliaikaisia muutoksia yrityksen tietoturvajärjestelmiin. Automatisointi voi esimerkiksi tarkistaa hälytykseen liittyvien IP-osoitteiden maineen ja maineen ollessa erityisen huono estää kyseisen IP-osoitteen pääsyn yrityksen verkkoon esimerkiksi palomuurisäännöin. Automatisointi voisi tämän jälkeen luoda asiasta vielä tiketin yrityksen asiakaspalvelujärjestelmään ja huomauttaa käyttäjää tehdyistä toimista. Automatisaation tulokset voidaan myös tuoda vielä ihmisen hyväksyttäväksi lopullisia toimenpiteitä varten. (14.)

SOAR-tuotteissa on yleisesti käytössä kaksi tapaa automatisoida tietoturvaloukkauksiin reagoimista. Toinen tavoista keskittyy työnkulun ja menettelytapojen automatisointiin ja toinen konfigurointimuutosten sekä korvaavien ja estävien toimien automatisointiin. Molempia voidaan automatisoida joko täysin tai vain osittain. Automatisointia varten ratkaisut sisältävät pelikirjoja, jotka ovat listoja eri tehtävistä tietyin ehdoin. Pelikirjojen sisältö perustuu yleisesti määriteltyihin parhaisiin toimintatapoihin kyseisen kaltaisissa tilan-

teissa. Pelikirjat suorittavat erilaisia skriptejä ajaakseen komentoja kolmansissa tuotteissa niiden API-rajapintojen kautta. Tällä tavoin hyökkäyksiin vastaamista formalisoidaan ja automatisoidaan parhaiten. Lisäksi tuotteet tukevat uusien, organisaation omien uusien pelikirjojen ja automatisaatioskriptien luomista. (14.)

Automatisoinnin ja orkestroinnin lisäksi SOAR tehostaa tietoturvaloukkausten tutkinnanhallintaa ja edistää analyytikoiden välistä yhteistyötä. Yksi erittäin tärkeä ominaisuus SOAR-järjestelmässä on sen kyky ylläpitää täydellistä historiaa tapauksessa ilmenneistä hälytyksistä ja toimista, joita eri analyytikot ovat tehneet. Tämän historian pohjalta voidaan asiakkaalle tai johtoryhmälle luoda merkityksellisiä raportteja ja yhteenvetoja tapahtuman kulusta ja tapahtumamääristä. Tutkinnan tehostamista varten SOAR-järjestelmä kerää yleensä SIEM-järjestelmien hälytykset ja luo niistä automaattisesti tutkintatiketin omaan järjestelmäänsä. Tämän jälkeen järjestelmä kerää ja analysoi tapaukseen liittyvää dataa useista eri lähteistä ja priorisoi hälytyksiä kerätyn tiedon perusteella käyttäen apuna tekoälyä ja koneoppimista. Näin SOAR pystyy vähentämään aiheettomia hälytyksiä sekä nostamaan tärkeän oloiset ja todennäköisesti oikeat hälytykset tutkintajonon kärkeen. (14.)

Uhkatietojärjestelmäintegraatioiden avulla SOAR pystyy rikastamaan tutkintaa siten, ettei analyytikon usein tarvitse edes käyttää muita järjestelmiä normaalin tutkinnan aikana. Raporttien luominen onnistuu myös suoraan SOAR-järjestelmästä, jonka lisäksi järjestelmä sisältää useita käytännöllisiä näkymiä SOC:n toiminnan seuraamiseen.

SOAR-järjestelmän toivotaankin tehostavan ja suoraviivaistavan päivittäistä toimintaa siten, että tulevaisuudessa pärjätään samalla analyytikkomäärällä, vaikka asiakasmäärät kasvaisivatkin. SOAR:n avulla pyritään myös parantamaan palvelun laatua standardisoimalla erilaisia toimintamalleja.

Käytännössä SOAR on siis ohjelmistokokonaisuus, jossa taustalla pyörivä prosessi ajaa integraatioskriptejä tasaisin väliajoin hakeakseen tietoturvatapahtumia useista lähteistä. Näistä tapahtumista luodaan tapauksia tietokantaan, jonka jälkeen järjestelmä liittää tapaukseen yhden tai useamman pelikirjan. Pelikirja sisältää viittauksia automaatioskripteihin, joilla tutkinnan rikastaminen ja automatisointi hoidetaan.

Ohjelmiston sisällä tapauksia yhdistellään toisiinsa koneoppimisen ja muiden korrelaatiometodien avulla. Ohjelmisto saattaa pyrkiä ehdottamaan toimintatapoja sen pohjalta, miten aikaisempia vastaavia tapauksia on hoidettu. Kaikki kerätty tieto tallennetaan mielellään erillisiin tietokantoihin, ja eri ympäristöjen pääsy pyritään rajaamaan vain omaan tietoonsa. Kaikki automatisaatioskriptit ajetaan virtuaalisäiliöissä.

Lisäksi ohjelmisto mahdollistaa useiden erityyppisten tietokantahakujen tekemisen raportoinnin helpottamiseksi. Monessa järjestelmässä raportointia varten on luotu omia editoreita, jotka täydentävät luotuun pohjaan lukuarvoja ja graafeja tietokannasta.

SOAR-järjestelmän teho automatisaatiojärjestelmänä perustuu siis paljolti pelikirjojen ja automatisaatioskriptien laatuun. Näillä skripteillä hoidetaan pääasiassa kaikki automatisaatioon liittyvät toimet. Lisäarvoa järjestelmä luo tuotteen tekoälyn avulla, joka pystyy korreloimaan tapauksia ja ehdottamaan toimia aikaisemman kokemuksen perusteella.

4 SOAR-integraatioyhteyksien suunnittelu

4.1 Tietoturvauhkien tunnistaminen

Tietoturvariskien tunnistamiseksi meidän täytyy ensin tuntea tietoturvallisuuden perimmäiset tavoitteet, joita ovat Shonin (15, s. 22) esittelemän mallin mukaan tiedon luottamuksellisuuden, saatavuuden ja eheyden takaaminen. Tietoturvallisuustyön päämääränä on siis turvata toiminnalle tärkeiden järjestelmien keskeytymätön toiminta, estää järjestelmien valtuudeton käyttö ja huolehtia, että tieto ei vääristy (16, s. 30).

Tietoturvariskin määrittelemiseksi joudumme ensin määrittämään termit haavoittuvuus, uhka ja uhkatekijä, joista tietoturvariski koostuu. Haavoittuvuus on yleisesti vastatoimien puute tai heikkous jo olemassa olevassa vastatoimessa. Haavoittuvuus voi olla lähes mikä tahansa tekijä ohjelmistossa, laitteistossa, menettelytavoissa tai henkilöstössä, jota voidaan hyväksikäyttää. Uhka on yksinkertaisesti mikä tahansa potentiaalinen vaara, joka liittyy haavoittuvuuden hyväksikäyttämiseen. Uhkatekijä taas on se tekijä, joka käyttää hyväksi haavoittuvuutta. (15, s. 26.)

Tietoturvariski on hieman yksinkertaistaen siihen liittyvän uhkan, sen toteutumisen todennäköisyyden ja toteutumisesta seuraavien vaikutuksien yhteinen tekijä (15, s. 140). Aiempaan termistöön viitaten uhkatekijä aiheuttaa uhan, joka hyväksikäyttää haavoittuvuutta josta seuraa riski. Riski taas toteutuessaan aiheuttaa vahinkoa ja mahdollistaa riskille altistumisen. (15, s. 27.)

Jotta voimme kartoittaa järjestelmään kohdistuvat tietoturvariskit, on meidän ensin kartoitettava järjestelmän käyttöympäristö ja käyttöympäristöön liittyvät uhkatekijät. Tämän jälkeen tulee meidän arvioida uhkien toteutumisen todennäköisyyttä ja niiden vaikutuksia uhkan toteutuessa. Vasta tämän jälkeen voimme arvioida ja kartoittaa järjestelmän eri osa-alueiden kokonaisriskiä. Rousku (17, s. 18) esittelee tämän riskienhallintaprosessin hyvin Valtionvarainministeriön Vahti 22/2017 julkaisussa, jossa kuvataan riskienhallinnan koostuvan toimintaympäristön kartoittamisesta, riskien tunnistamisesta, riskianalyysista, riskien merkityksen arvioinnista ja lopulta riskien käsittelystä.

Tietoturvariskien mahdollisimman tarkka ja kokonaisvaltainen hahmottaminen on siis tärkeätä, jotta voimme myöhemmin laatia mahdollisimman tehokkaan tietoriskienhallintasuunnitelman. Suunnitelma tietoriskien hallinnasta on osa tietoturvapolitiikkaa, ja sen tarkoitus on kuvata tietoriskienhallintaprosessi, tunnistetut uhat ja niiden vaikutukset sekä päättää toimenpiteistä, joilla riskiä hallitaan. (16, s. 35.)

4.1.1 Toimintaympäristön määrittäminen

Toimintaympäristöä määriteltäessä tehdään rajaukset siitä, mitä sisällytetään riskienarviointiin ja mitä jätetään sen ulkopuolelle. Toimintaympäristön määrittelyssä rajataan: järjestelmän ulkoinen ja sisäinen toimintaympäristö, riskienhallintaprosessin toimintaympäristö kokonaisuudessaan ja riskikriteerit, eli miten riskejä hallitaan. (17, s. 19-20.)

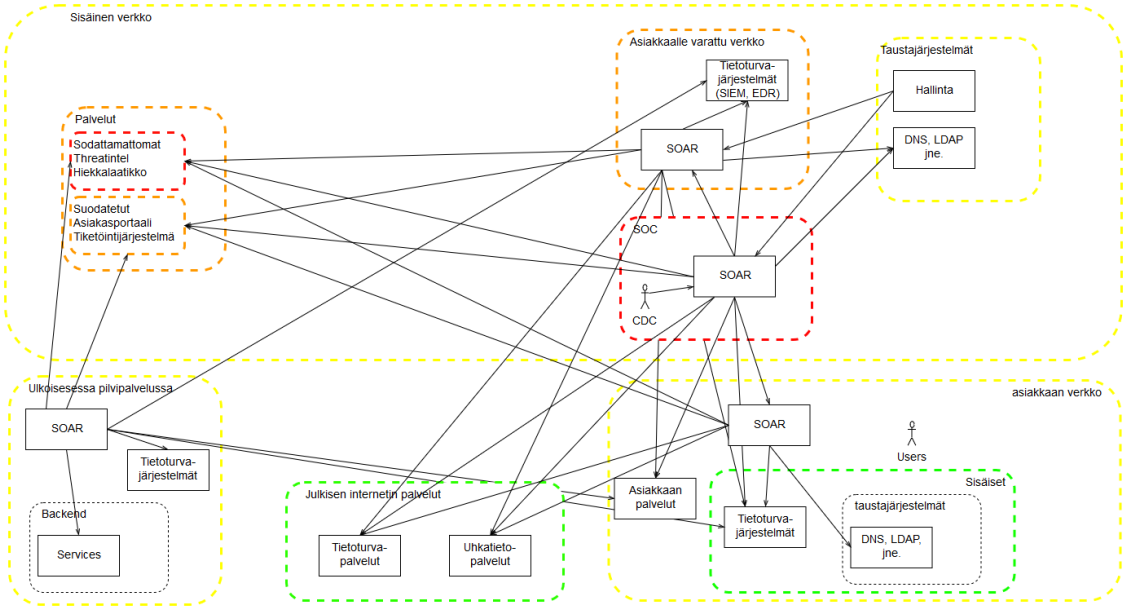
Aloitamme SOAR-järjestelmän toimintaympäristön kartoittamisen määrittelemällä ne paikat, joihin SOAR-järjestelmän osia tullaan todennäköisesti asentamaan ja miten eri suojaustason henkilöstö pääsee niitä käyttämään. SOAR-palvelin voidaan MSSP:n käytössä asentaa ainakin

- keskusinstanssina, joka sijaitsee valvovan organisaation ydinverkossa ja josta on yhteydet kaikkiin valvottaviin ympäristöihin ja niissä sijaitseviin sensoreihin ja lokilähteisiin.
- jaettuna instanssina, joka sijaitsee valvovan organisaation verkossa ja josta on yhteydet moneen eri organisaation valvottavaan ympäristöön
- eristettynä instanssina, joka sijaitsee valvovan organisaation verkon eristetyssä osassa ja josta on pääsy vain yhteen valvottavaan ympäristöön
- eristettynä instanssina, joka sijaitsee kokonaisuudessaan valvottavan asiakkaan omassa ympäristössä sekä hallinnassa ja näin ollen valvontaa suorittavan organisaation ympäristön ulkopuolella
- pilvipalveluna.

Käyttäjien pääsyä järjestelmään on järkevää rajata siten, että keskusinstanssiin annetaan oikeudet vain valvovan organisaation omalle SOC:ssa työskentelevälle henkilöstölle. Jaettuun instanssiin tämän lisäksi pääsy myös niille asiakkaille, joiden ympäristön valvontaan tätä instanssia käytetään ja eristettyihin instansseihin vain kyseisen asiakkaan määräämälle henkilöstölle. Näistä jaetun instanssin turvaaminen osoittautuu nopeasti erittäin monimutkaiseksi, jollei lähes mahdottomaksi, koska SOAR mahdollistaa käyttäjän ajaa hänen itsensä määrittelemää koodia SOAR-palvelimella. Vaikka ohjelmisto eristäisikin eri asiakkaat toisistaan täydellisesti, ei mikään silti estä asiakasta näkemästä toisen asiakkaan verkkoon SOAR-järjestelmän avulla. Tämän takia toteammekin, että turvallisen jaetun instanssin luominen on lähes mahdotonta, ja jätämme sen turvaamisen tutkimisen tähän.

Diagrammin piirtäminen on hyvä tapa kuvata mitä ollaan rakentamassa, ja hyvä tapa aloittaa on piirtää diagrammi siitä, miten tieto virtaa järjestelmässä. Tämän jälkeen kuvaan on hyvä lisätä rajat, jotka näyttävät kuka hallitsee mitäkin osaa järjestelmästä. Näitä rajoja kutsumme luottamusrajoiksi. (18, s. 6-7.)

Seuraavaksi määrittelemämme ympäristön eri osat, niiden tietoturvatasot, luottamusrajat ja yhteydet eri osien välillä. Tämän määrittelyn tulokset esitellään kuvassa 6. Kuvassa luottamusrajat on määritelty katkoviivoin ja ympäristön eri osat neliöin. Yhteydet ympäristön eri osien välillä esitetään nuolin, jotka kuvaavat informaatiopyyntöjen suuntaa sekä tietoturvatasot eri värein vihreästä punaiseen siten, että punainen on kaikkein korkein tietoturvataso ja vihreä kaikkein alhaisin. Tietoturvatasot on määritelty sen mukaan, kuinka paljon eri asiakkaiden luottamuksellista tietoa alueen sisällä säilytetään ja käsitellään.



Kuva 6. SOAR-toimintaympäristö. Toimintaympäristöön on merkitty luottamusrajat värein, siten että punainen on kaikkein luottamuksellisin alue ja vihreä vähiten luottamuksellinen. Rajat perustuvat alueella säilytettävän luottamuksellisen tiedon määrään.

Kuten kuvasta 6 voimme nopeasti todeta, on SOAR-järjestelmän toimintaympäristö hyvin monimutkainen. Jotta emme eksyisi tämän monimutkaisen järjestelmän tutkimisessa, käsittelemme sitä pienemmissä osissa. Jaamme ympäristön osiin aikaisemmin määrittelemiemme asennuspaikkojen mukaisesti. Tämän johdosta joudumme ottamaan huomioon eri osien välillä olevat yhteydet erikseen, mutta koska jakomme perustui SOAR-tuotteen asennuspaikkaan, ovat nämä yhteydet lähinnä SOAR:sta SOAR:iin.

Kun diagrammi on pilkottu osiin, voimme todeta, että eri paikkoihin asennetut SOAR-järjestelmät toimivat lähes identtisissä ympäristöissä. Ne ovat tietoturvatasoiltaan hieman eroavaiset ja joudummekin ottamaan tämän myöhemmin huomioon, mutta jokainen SOAR-asennus tarvitsee yhteydet samoihin palveluihin muutamaa poikkeusta lukuun ottamatta.

SOAR-integraatioyhteydet on esitelty taulukossa 1. Taulukossa pyritään kuvaamaan, mistä mihin eri yhteyksiä SOAR-järjestelmässä tarvitaan, ylittääkö yhteys luottamusrajoja ja onko tietoa liikkumassa korkeammalta suojaustasolta alemmalle aiheuttaen riskin tietovuodoista.

Taulukko 1. Eri tyyppiset yhteydet SOAR-järjestelmässä ja arvio siitä liittyykö niihin riskiä tietovuodosta

Mistä	**Mihin**	**Ylittääkö luottamusrajan**	**Tietovuotoriski?**
Keskus SOAR	SOAR	Kyllä, mutta informaatio liikkuu vastakkaiseen suuntaan	Ei, kunhan kysely ei sisällä luottamuksellista informaatiota
Keskus SOAR	Tietoturvajärjestelmiin	Kyllä, mutta informaatio liikkuu vastakkaiseen suuntaan	Ei, kunhan kysely ei sisällä luottamuksellista informaatiota
Keskus SOAR	Sisäiset luottamukselliset palvelut	Ei	Ei
Keskus SOAR	Sisäiset palvelut	Kyllä	Ei
SOAR	Tietoturvajärjestelmiin samassa ympäristössä	Ei	Ei
SOAR	Palveluihin Internetissä	Kyllä	Kyllä, tiedostot yms. saattavat sisältää luottamuksellista tietoa
SOAR	Sisäisiin luottamuksellisiin palveluihin	Kyllä	Kyllä, Uhkatieto saattaa sisältää luottamuksellista tietoa
SOAR	Sisäiset palvelut	Kyllä	ei

4.1.2 Tietoturvauhkien tunnistaminen

Tietoturvauhkien tunnistamisen aloitamme hyvin yksinkertaisin metodein. Ensimmäisenä kysymme, mistä uhista olemme huolissamme. Tämän jälkeen pidämme vapaan aivoriihen, jonka aikana pyrimme tunnistamaan toimintaympäristön eri kohtiin liittyviä uhkia, jotka liittyvät määrittelemiimme huoliin. (18, s. 29-30.)

SOAR-integraatioyhteyksien tapauksessa olemme huolissamme, että

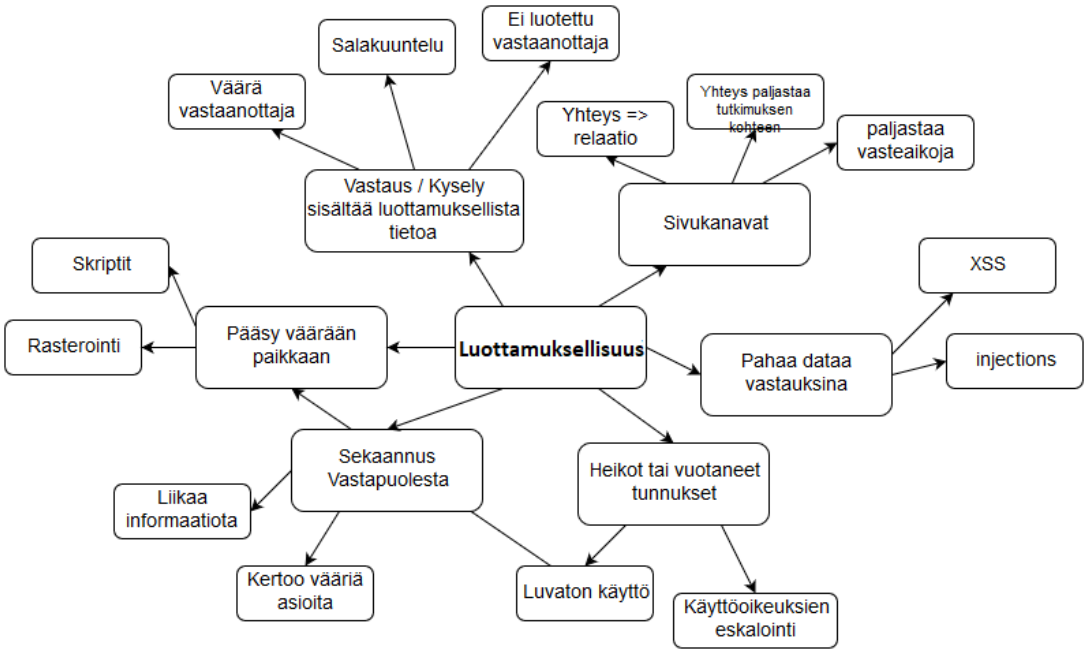
- asiakkaan salaista tai luottamuksellista tietoa vuotaa toiselle asiakkaalle tai yleiseen jakeluun SOAR-integraation kautta
- SOAR mahdollistaa komentojen ajamisen asiakkaan ympäristössä ilman valtuutusta
- SOAR-automatisaatioiden suorittamat toimet paljastavat yksityiskohtia käynnissä olevasta tutkinnasta
- palvelunestohyökkäyksistä SOAR-järjestelmää vastaan tai sitä hyväksikäyttäen.

Perinteinen ja edelleen yleisesti käytössä oleva tapa kartoittaa järjestelmään kohdistuvia tietoturvauhkia on kerätä muutama tietoturva-asiantuntija samaan huoneeseen ja yhdessä miettiä, mitä uhkia järjestelmää kohtaan voisi olla (18, s. 31-32). Näiden aivoriihien aikana on tarkoitus päästää ilmoille mahdollisimman paljon erilaisia uhkia, joita järjestelmään voisi kohdistua. Uhkien ei tarvitse välttämättä olla realistisia vaan tärkeintä on, että eri tyyppisiä uhkia saadaan kartoitettua mahdollisimman kattavasti.

Aloitamme uhkien kartoittamisen jakamalla ongelman kolmeen osaan tietoturvallisuuden kolmen peruspilarin, luottamuksellisuuden, saatavuuden ja eheyden mukaan. Kartoitamme jokaista perusarvoa kohtaan kohdistuvia uhkia piirtämällä niistä ajatuskartan. Lopuksi kasaamme näistä ajatuskartoista järjestelmään kohdistuvat tietoturvauhat yhteen taulukkoon. On tärkeätä huomata, että tietoturvauhkien kartoittaminen on prosessi, joka ei valmistu koskaan, vaan tarkoituksena on luoda mahdollisimman kattava kartta mahdollisista uhista (18). Tietoturvauhkien kartoittaminen voikin ja sen jopa tulisi jatkua myöhemmin uudella ajatusriihikierroksella, koska kaikkia uhkia ei saada koskaan kartoitettua kerralla.

Esittelen kuvassa 7 SOAR-järjestelmän luottamuksellisuuteen kohdistuvia mahdollisia uhkia. Uhkat liittyvät ainakin integraatioyhteyksien vastauksien tai API-kyselyjen sisältämään luottamuksellisen tiedon paljastumiseen, yhteyksien mahdollisuuteen päästä käsiksi väärään ympäristöön ja sekaannuksesta vastaanottajasta tai vastauksien mahdollisesti sisältämistä pahansuovista skripteistä tai vastaavista. Myös heikot tai tavalla tai toisella vuotaneet tunnukset aiheuttavat uhan luottamuksellisuudelle. Lisäksi hyökkääjä saattaa saada luottamuksellista tietoa sivukanavia pitkin yhdistelemällä useista eri lähteistä saatavaa tietoa.

Sivukanavahyökkäyksiksi voimme kutsua kaikkia niitä tapoja kerätä tietoa järjestelmästä tai vaikuttaa sen toimintaan ilman suoraa yhteyttä järjestelmään. Hyökkääjä voi esimerkiksi yhdistellä yhteyksien IP-osoitetietoja ja julkisista rekistereistä saatavilla olevaa omistajatietoa ja saada näin selville tutkintaa suorittavan tahon tai muuta tutkintaan liittyvää luottamuksellista tietoa.



Kuva 7. Tietoturvauhkia tiedon luottamuksellisuudelle SOAR:n yhteyksissä

Koska SOAR tarvitsee pääsyn jokaiseen integroitavaan järjestelmään, aiheutuu sen käytöstä uhka, että SOAR:a käyttävä henkilö tai sen automatisaatio pääsee käsiksi ympäristöön, johon hänellä ei kuuluisi olla oikeutta. Esimerkiksi SOAR:n rasterointitoiminto, joka ottaa käyttäjän puolesta ruutukaappauksen käyttäjän määrittelemästä sivustosta, voi vuotaa tietoa muista ympäristöistä. Jos SOAR pääsee rajoittamattomasti käsiksi asiakasympäristöihin voi hyökkääjä nähdä tämän toiminnon avulla toisen organisaation rajatun verkon sisällä olevia palveluita ja resursseja.

Integraatioyhteyden lähettämät kyselyt tai vastaukset saattavat sisältää luottamuksellista tietoa. Mikäli integraatioyhteyttä ei salata oikeaoppisesti voi hyökkääjä mahdollisesti lukea sen matkan varrella tai toimia välittäjänä esittäen olevansa integraation toinen pää. Integraatioyhteys saattaa myös päätyä väärälle vastaanottajalle esimerkiksi DNS-kyselyjen vastauksia muokkaamalla, ellei vastapuolen identiteettiä varmisteta. Tietoa voidaan myös lähettää organisaation ulkopuoliselle vastaanottajalle, joka ei välttämättä ole luotettu, jolloin on olemassa uhka, että tämä tieto päätyisi vääriin käsiin.

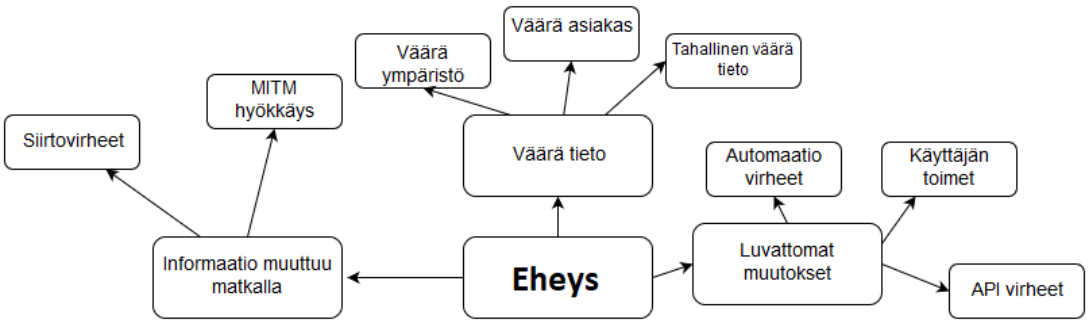
Heikot tai vuotaneet tunnukset päästävät hyökkääjän käsiksi luottamukselliseen tietoon, jonka lisäksi hän saattaa saada SOAR-järjestelmästä lisää tunnuksia toisiin järjestelmiin.

SOAR-järjestelmän täytyy tallentaa kaikkien integraatioiden tarvitsemat tunnukset johonkin ja käytettävä niitä yhteyksien muodostamisvaiheessa. Näillä tunnuksilla hyökkääjä voi esiintyä SOAR-integraationa ja tehdä samoja asioita, mihin SOAR-järjestelmällä on oikeudet.

Koska osa integraatioyhteyksien vastapuolista ei välttämättä ole SOAR-järjestelmää hallitsevan organisaation hallussa saattavat integraatioyhteyksien saamat vastaukset sisältää mitä tahansa dataa mukaan lukien ilkeämielisiä XSS- tai injektiohyökkäyksiä. Cross Site Scripting (XSS) avulla hyökkääjä kykenee suorittamaan komentoja palvelua käyttävän käyttäjän selaimessa ja injektiohyökkäykset mahdollistavat tiedon vuotamisen, muokkaamisen tai poistamisen tietokannoista tai käyttäjänhallintajärjestelmistä. Sekä äärimmäisissä tapauksissa hyökkääjän määrittelemän koodin ajamista kohdejärjestelmässä.

Lisäksi SOAR-järjestelmän muodostamat yhteydet saattavat paljastaa tietoa, vaikka hyökkääjällä ei olisi mitään näkyvyyttä itse järjestelmään. Hyökkääjä voi esimerkiksi saada selville asiakassuhteita IP-osoitteiden perusteella, jos SOAR ottaa yhteyttä asiakkaan järjestelmiin. Vastaavasti taitava hyökkääjä voi selvittää käynnissä olevia tutkintoja ja niiden kohteita vertaamalla omia tekemisiään kohdeorganisaation järjestelmissä ja SOAR-järjestelmän muodostamia yhteyksiä ja niiden ajoituksia.

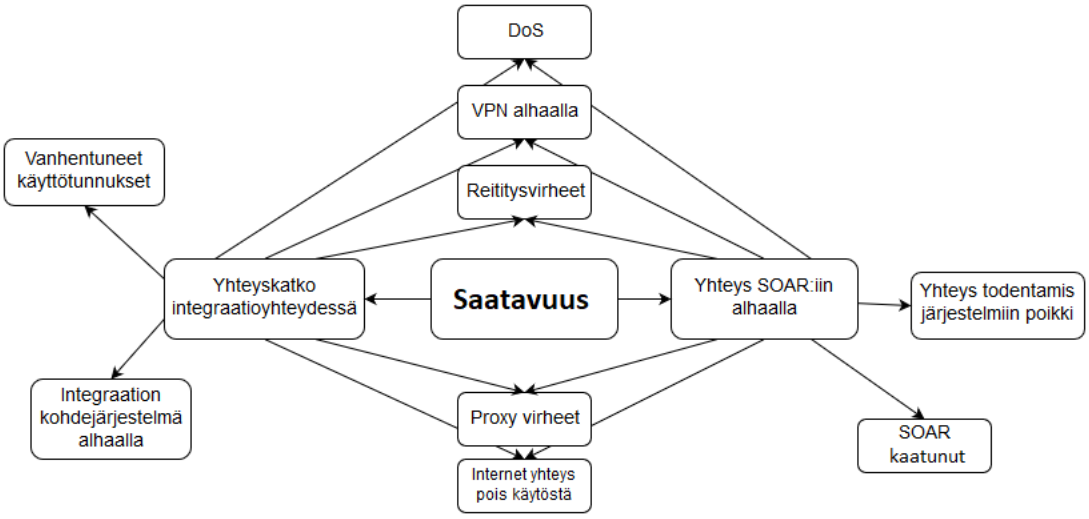
Kuvassa 8 käymme läpi tiedon eheydelle mahdollisesti aiheutuvia uhkia.



Kuva 8. Tietoturvauhkia tiedon eheydelle SOAR:n yhteyksissä

Tiedon eheys vaarantuu, jos informaatio muuttuu matkalla tai informaatioon tehdään luvattomia muutoksia. Lisäksi SOAR saattaa saada väärää tietoa, joka aiheuttaa luvattomia tai sen kaltaisia muutoksia tietoon. Tieto voi muuttua siirron aikana siirtovirheiden takia tai jonkun tahallisesti muuttaessa sitä esimerkiksi man-in-the-middle-hyökkäyksellä. MITM-hyökkäyksessä hyökkääjä hallitsee yhtä tai useampaa laitetta tiedonsiirtoreitin varrella ja voi näin vaikuttaa tiedonsiirtoon. Luvattomia muutoksia voi tehdä joko käyttäjä, SOAR:n automaatioskripti tai API-rajapinnassa olevat virheet. Koska SOAR tallentaa tietoa usean eri ympäristön järjestelmiin on olemassa uhka, että SOAR-järjestelmälle syötetty tieto aiheuttaa luvattomia muutoksia muissakin järjestelmissä. Tämä voi tapahtua ympäristön tai asiakkuuden sekaantuessa tai jos SOAR -järjestelmälle syötetään tarkoituksella väärää tai pahansuopaista tietoa.

Lopuksi mietimme tiedon saatavuudelle voivan aiheutua ainakin kuvassa 9 esiteltyjä uhkia.



Kuva 9. Tietoturvauhkia tiedon saatavuudelle SOAR:n yhteyksissä.

SOAR-järjestelmän integraatioyhteyksien saatavuudelle aiheutuu uhkia kahdella eri tavalla. Joko SOAR ei saa yhteyttä integraation kohteeseen tai käyttäjät eivät saa yhteyttä SOAR-järjestelmään, jolloin integraatiota ei voida käyttää. Kuten kuvasta 9 huomaamme, molemmissa tapauksissa katkokseen johtavat syyt ovat lähes samat. Joko järjestelmiä on kaatunut, verkkolaitteissa on tapahtunut konfigurointivirheitä tai ne ovat

pois käytöstä. Käyttäjätunnuksissa on ongelmia tai niitä ei voida varmentaa, koska taustajärjestelmät eivät toimi oikein.

4.1.3 Tietoturvariskit

Riski on siis käsite, joka sisältää uhan, sen todennäköisyyden ja sen mahdolliset vaikutukset. Uhkakartoituksen apuvälineeksi kehitetyn FAIR-järjestelmä määrittelee riskin olevan tulevaisuudessa tapahtuvan menetyksen todennäköinen yleisyys (Loss Event Frequency, LEF) ja todennäköisen suuruuden (Loss magnitude, LM) tulo (19, s.27). Riskin määritelmä perustuu siis menetykselle altistumiseen ja arvioihin siitä, kuinka todennäköistä tämä on ja kuinka suuret vahingot riskin toteutumisesta seuraa.

Loss Event Frequency (LEF) tarkoittaa siis sitä todennäköisyyttä, jolla tietty riski materialisoituu määrätyn ajan kuluessa (19, s. 28). LEF koostuu uhkan toteutumistodennäköisyydestä (TEF) tiettynä ajanjaksona ja siihen liittyvästä haavoittuvuudesta. TEF:llä tarkoitetaan sitä todennäköisyyttä, jolla jonkin uhkatoimijan teko johtaa uhkan toteutumiseen tietyn normalisoidun ajanjakson aikana. Tähän todennäköisyyteen vaikuttaa se, kuinka usein uhkatekijällä on mahdollisuus olla kanssakäymisissä uhkan kohteen kanssa, uhkatekijän taito ja uhkaan liittyvän haavoittuvuuden hyväksikäyttämisen vaikeus. Uhkaan liittyvä haavoittuvuus sisältää tässä mallissa siis myös haavoittuvuuden hyväksikäyttämisen vaikeusasteen. (19, s. 28-29.)

Loss Magnitude (LM) koostuu uhkan toteutumisesta organisaatiolle aiheutuvista todennäköisistä ensisijaisista menetyksistä ja toissijaisista riskeistä. Ensisijaisilla seurauksilla tarkoitetaan uhkan kohteelle suoraan aiheutuvia vahinkoja kuten esimerkiksi katkoksen aiheuttamat menetetyt tulot. Toissijaisilla riskeillä tarkoitetaan riskiä siitä, että jokin toisen osapuolen reaktio ensisijaisen riskin johdosta, aiheuttaa uuden riskin ja sitä kautta uusia mahdollisia menetyksiä organisaatiolle. (19, s. 35-39.)

Koska riski itsessään sisältää huomattavan määrän epätarkkuuksia todennäköisyyksien ja arvioiden muodossa on riskin määritteleminenkin tehtävä todennäköisyyksien ja arvioiden perusteella. Miten riskin suuruutta sitten kannattaa arvioida? Yksi tapa hallita monimutkaisia arvioihin perustuvia projekteja on 1950-luvulla Yhdysvaltojen laivaston kehittämä PERT-menetelmä (19, s. 78). PERT-menetelmän avulla voimme arvioida ta-

pahtuman todennäköisyyttä arvioimalla, mikä on sen pienin mahdollinen todennäköisyys, todennäköisin todennäköisyys ja korkein todennäköisyys. Voimme vielä painottaa jotain näistä kolmesta arvosta sen mukaan, minkä uskomme olevan suurin vaikuttavatekijä. PERT-laskentakaava määritellään seuraavasti

$$Odotusarvo = (\frac{a * t_{min} + b * t_{tod} + c * t_{max}}{a + b + c})$$

Tässä t_{min} on pienin tekijä, t_{tod} on todennäköisin tekijä ja t_{max} suurin mahdollinen tekijä. Arvot a, b ja c kuvastavat varmuuttamme siitä, että kyseinen tekijä on totuudenmukainen. (19, s. 78-79.)

Voimme laskea riskin mahdolliset vaikutukset samaa kaavaa hieman muokkaamalla

$$odotettu\ vaikutus = (\frac{v_{min} * t_{min} + v_{tod} * t_{tod} + v_{max} * t_{max}}{t_{min} + t_{tod} + t_{max}})$$

jossa v_{min}, v_{tod} ja v_{max} ovat arvioita vaikutuksen suuruudesta ja t_{min}, t_{tod} ja t_{max} arvioiden todennäköisyyksiä. Lopulta voimme laskea riskin suuruuden kertomalla uhkan toteutumisen todennäköisyyden sen vaikutuksien suuruudella.

Aloitamme riskien ja niiden suuruuden määrittelemisen kasaamalla ja tiivistämällä edellä käsitellyt uhat yhteen tauluun. Tässä vaiheessa pyrimme karsimaan pois ne uhat, joiden emme usko koskaan voivan toteutua tai joiden vaikutuksien tiedämme olevan olemattomat. Pyrin samalla tiivistämään samaa uhkaa kuvaavat tapahtumat yhdeksi kokonaisuudeksi. Tämän jälkeen laskemme jokaiselle uhkalle todennäköisyyden ja todennäköisten vaikutuksien suuruuden uhkan toteutuessa. Painotamme uhkan todennäköisyydessä todennäköisintä neljällä ja muita yhdellä. Todennäköisyys kuvastaa todennäköisyyttä sille, että uhka toteutuu kerran vuodessa. Vaikutukset normalisoimme siten, että 0 tarkoittaa ei vaikutuksia ja 1 tarkoittaa yritykselle katastrofaalisia vaikutuksia. Näin päädymme seuraavanlaiseen taulukkoon.

Taulukko 2. Havaitut tietoturvariskit ja niiden arvioidut toteutumistodennäköisuudet sekä vaikutukset.

Uhka	t_{est}	v_{est}	Riski
Hyökkääjä saa tietoonsa korkean tason käyttäjä salasanan	0,1	0,9	0,09
Hyökkääjä saa tietoonsa normaalin käyttäjän salasanan	0,2	0,6	0,12
Hyökkääjä saa tietoonsa alhaisen tason käyttäjän salasanan	0,4	0,42	0,17
Hyökkääjä saa tietoonsa järjestelmän API avaimen tai avaimia	0,2	0,7	0,14
Hyökkääjä pystyy kuuntelemaan API kutsuja tai vaikuttamaan niihin	0,2	0,72	0,14
Luottamukselliset API kyselyt päätyvät väärälle vastaanottajalle	0,52	0,5	0,26
Hyökkääjä voi huijata käyttäjän selaimen suorittamaan komentoja kirjoittamalla API kutsujen käyttämiin lähteisiin	0,35	0,5	0,18
Injektiohyökkäys API integraatiota vastaan	0,35	0,5	0,18
Sisäisten integraatiojärjestelmien API vastaukset sisältävät luottamuksellista tietoa toisesta asiakkaasta	0,95	0,68	0,65
Käyttäjä pystyy lukemaan dataa luottamuksellisesta ympäristöstä integraation avulla	0,72	0,68	0,49
Käyttäjä pystyy kirjoittamaan yhteen tai useampaan luottamukselliseen ympäristöön integraatioiden avulla	0,48	0,98	0,48
palvelunestohyökkäys järjestelmän käyttämiä yhdyskäytäviä vastaan	0,73	0,32	0,23
Hyökkääjä voi hyödyntää integraatiota palvelunestohyökkäyksen toteuttamisessa	0,33	0,5	0,17
0-päivä haavoittuvuus SOAR:ssa	0,32	0,5	0,16
Käyttäjä pystyy laajentamaan pääsyään alhaisemmasta järjestelmästä pääjärjestelmään integraatio yhteyden avulla	0,28	0,88	0,25
Hyökkääjä pystyy päättelemään luottamuksellista tietoa integraatioyhteyksien ajoituksesta	0,42	0,62	0,26
Hyökkääjä kykenee päättelemään luottamuksellista tietoa integraatioyhteyksien kohteesta	0,48	0,62	0,3
Hyökkääjä saavuttaa täyden hallinnan yhteen tai useampaan asiakasympäristöön integraatioiden avulla	0,1	1	0,1
Palvelunestohyökkäys järjestelmän osaa vastaan	0,28	0,32	0,09

Riski sisältää usein myös tekniikan, jolla haavoittuvuutta voidaan hyväksikäyttää. Jos kykenemme tunnistamaan nämä tekniikat ja estämään niiden käytön, pystymme pienentämään riskin toteutumisen todennäköisyyttä tai jopa täysin estämään sen. (15.) Avaamme siis seuraavaksi sitä, miten hyökkääjä voisi hyväksikäyttää havaitsemiamme riskejä.

Hyökkääjä voi saada tietoonsa käyttäjätunnuksia tai API-avaimia arvaamalla, yrittämällä useita eri yhdistelmiä, lähettämällä käyttäjille tietojenkalasteluviestejä tai onnistuneesti salakuuntelemalla kirjautumisia. Lisäksi API-avaimet voivat olla tallennettuina selkokielisinä SOAR-järjestelmässä. API-avaimet ovat jo lähtökohtaisesti pitkiä ja monimutkaisia, eikä niiden arvaaminen ole helppoa. Järjestelmään tallennettujen avaimien säilytykseen emme juurikaan voi vaikuttaa muuten kuin käyttämällä yhtä avainta vain yhdessä järjestelmässä ja vaihtamalla sen säännöllisesti. Käyttäjien salasanojen tulisi olla mieluummin

salalauseita, jossa yhden sanan sijasta käyttäjää ohjeistetaan luomaan helposti muistettava lause ja korvaamaan joitain merkkejä numeroilla ja erikoismerkeillä (20).

API-kutsujen kuunteleminen tai muokkaaminen vaatii hyökkääjältä vähintään mahdollisuuden hallita yhtä pistettä pakettien matkanvarrella ja todennäköisesti jonkinasteisesta salauksenpurkamista. API-kutsujen ohjautuminen väärään osoitteeseen voi johtua reititys- tai palomuurijärjestelmienkonfiguraatiovirheistä tai vihamielisestätoiminnasta, jossa hyökkääjä kontrolloi yhtä tai useampaa pistettä matkan varrella. Kaikki nämä antavat hyökkääjälle käytännössä mahdollisuuden toimia jonkin toisen käyttäjän nimissä ja hänen oikeuksillaan. HTTPS-protokolla kunnollisilla sertifikaateilla ja asetuksilla estää kuitenkin käytännössä tämäntyyliset hyökkäykset kokonaan.

Organisaation alun perin sisäiseen käyttöön suunnittelemat järjestelmät sisältävät usein huomattavankin arkaluontoista tietoa asiakkaista, eikä tämän tiedon käyttöoikeuksien rajaamista ole välttämättä ajateltu kovin kattavasti. Integraatioyhteyksien täytyy kuitenkin saada asiakasympäristöistä luottamuksellista tietoa ja jossain tapauksissa myös kyetä tekemään muutoksia kyseisiin järjestelmiin. Mikäli hyökkääjä hallitsee API-kutsujen tekemistä tai kykenee kuuntelemaan tai muokkaamaan API-kutsuja matkanvarrella saa hän todennäköisesti pääsyn ympäristöön tai tietoon johon hänellä ei muuten olisi oikeuksia.

Injektiohyökkäykset voivat kohdistua joko SOAR-järjestelmän käyttäjiä kohtaan tai itse järjestelmään. Jos järjestelmä ei kykene suodattamaan integraatioiden tuomaa dataa oikein saattaa käyttäjän selain suorittaa esimerkiksi vihamielistä javascript-koodia. Vaihtoehtoisesti hyökkääjä voisi kohdistaa hyökkäyksen haavoittuvaa integraatiokoodia kohtaan esimerkiksi python object-injektio -hyökkäysmenetelmällä.

Palvelunestohyökkäykset järjestelmän osia kohtaan hidastavat normaalin käyttäjän toimintaa tai estävät sen kokonaan. Lisäksi pakottamalla integraatiot tekemään useita kyselyitä voi hyökkääjä toteuttaa palvelunestohyökkäyksiä integraatiojärjestelmiä vastaan. Julkiseen verkkoon kohdistuvat hyökkäykset ovat huomattavasti todennäköisempiä kuin järjestelmän sisäisessä verkossa pyöriviin järjestelmiin kohdistuvat, mutta näillä on kuitenkin samankaltainen vaikutus järjestelmän toimintaan.

Koska integraatiot hakevat tietoa julkisista lähteistä ja tutkivat järjestelmiä, jotka saattavat olla hyökkäyksen kohteena, voi hyökkääjä päätellä luottamuksellisia asioita yhdistelemällä eri lähteistä saamaansa tietoa. Estääksemme tämän tyyppisiä hyökkäyksiä täytyy meidän piilottaa vähintään alkuperäisten kyselyiden alkuperä ja niiden aloitusaika.

Viimeiseksi voimme todeta, että tuntemattomia tai tunnettuja mutta paikkaamattomia aukkoja hyväksikäyttämällä hyökkääjä voi joko korottaa oikeuksiaan alemmasta järjestelmästä ylempään tai kiertämään tuotteen kirjautumisvaatimukset kokonaan. Molemmissa tapauksissa hyökkääjä kykenee tämän jälkeen hallitsemaan järjestelmää lähes täydellisesti.

4.2 Tietoturvariskienhallinta

Tietoturvariskinhallinta on prosessi, jonka tarkoituksena on tunnistaa riski, arvioida se ja vähentää sitä hyväksyttävälle tasolle (15, s. 70). Riskin todennäköisyyttä ja sen vaikutusta voidaan pienentää erilaisilla vastatoimilla, jotka voidaan jakaa karkeasti seuraaviin kategorioihin

- hallinnolliset
- tekniset
- fyysiset. (15, s. 28.)

Hallinnolliset vastatoimet ovat lähinnä ohjeistuksia, sääntöjä ja koulutuksia työntekijöille. Tekniset vastatoimet sisältävät kaikki teknisiä järjestelmiä hyväksikäyttävät pääsyä rajoittavat tai käyttöä valvovat menetelmät. Näitä ovat esimerkiksi palomuurit, IPS-järjestelmät tai salausmenetelmät. Fyysisiä vastatoimia ovat kaikki fyysiset esteet, jotka pyrkivät turvaamaan tiloja, henkilöstöä tai resursseja. Näitä ovat esimerkiksi aidat, vartijat tai jopa valaistus. (15, s. 27-28.)

Turvallisuuden lisäämiseksi käytettävien vastatoimien toiminnallisuus voidaan jakaa kuuteen eri kategoriaan. Estävät vastatoimet pyrkivät estämään riskin toteutumista. Pidättelevät vastatoimet taas pyrkivät lannistamaan hyökkääjän ennen kuin riski ehtii toteutumaan. Korjaavat toimet pyrkivät edistämään riskin toteutumisesta palautumista ja palauttavat vastatoimet palauttamaan ympäristön normaaliin toimintakuntoon riskin toteutumisen jälkeen. Lisäksi tarvitsemme havaitsevia vastatoimia, jotka nimensä mukaan

pyrkivät havaitsemaan mahdollisen tunkeutujan. Näiden lisäksi vastatoimi voidaan lajitella korvaavaksi, jolloin se pyrkii tarjoamaan vaihtoehtoisen tavan toteuttaa jokin toinen vastatoimi. (15, s. 34.)

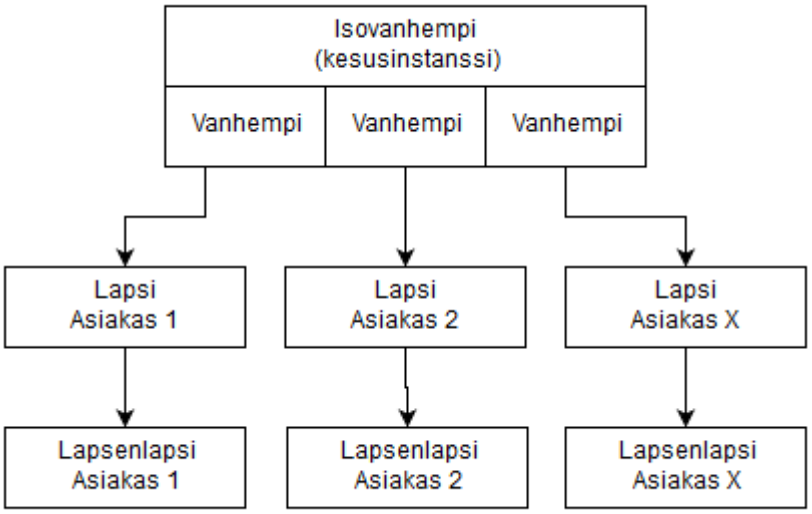
Aloitamme tietoturvariskienhallinnan määrittelemällä ympäristölle hallinnollisen tietoturvamallin. Mallin tarkoituksena on ohjeistaa kaikkea järjestelmään liittyvää toimintaa siten, että tietovuotoja ja tietoturvaloukkauksia ei pääsisi tapahtumaan. Malli ei tietenkään kykene estämään kaikkea, mutta sitä noudattamalla päästään jo pitkälle. Tietoturvamallia käytämme luotettavan tietojenkäsittelyjärjestelmän muodostamisessa. Luotettavan tietojenkäsittelyjärjestelmän tarkoitus on estää käyttäjää tekemästä asioita joita hän ei tietoturvamallin mukaan saisi tehdä. Tämän järjestelmän periaatteena on minimoida tarvittavien poikkeuksien määrä ja täten taata mahdollisimman turvallinen ympäristö. (21.)

Koska ympäristössämme on tarkoitus suojella tietoa leviämiseltä tai pitää sitä salassa muilta käytämme tietoturvasääntöjemme pohjana Bell David Eliotin ja La Padula J. Leonardin kehittämää Bell-LaPadula mallia. Bell-LaPadula on alun perin Yhdysvaltojen puolustusministeriölle kehitetty malli salaisen ja luottamuksellisen tiedon pitämiseksi salaisena. Bell-LaPadula esittelee kolme tietoturvasääntöä, joista kaksi on pakollisia pääsynvalvontametodeja ja yksi harkinnanvarainen, sekä näiden lisäksi vanhempi-lapsi-mallin. Malli esittelee seuraavat tietoturvaominaisuudet:

- Yksinkertainen suojausominaisuus esittää, että tietyn tietoturvatason kohde ei saa lukea miltään korkeamman suojaustason kohteelta.
- * (tähti) ominaisuus esittää, että kohde jolla on pääsy useammalle tietoturvatasolle, ei saa kirjoittaa millekään alemmalla turvatason kohteelle.
- Harkinnanvarainen tietoturvaominaisuus esittää, että harkinnanvarainen pääsynvalvonta määritetään käyttömatriisin avulla. (22.)

Mallin tähtiominaisuudelle on määritelty poikkeus. Jos kohde on luokiteltu luotettavaksi, eli sen ei oleteta rikkovan luottamusketjua, ei tähtiominaisuus päde siihen (22). Supistamme tätä määritelmää hieman SOAR-järjestelmänintegraatioiden tapauksessa ja määrittelemme tähtiominaisuuden ja kohteen luotettavuuden siten, että järjestelmän luotetaan pitävän omat luottamukselliset tietonsa itsellään ja näin ollen järjestelmä voi itse kirjoittaa alemmalle tietoturvatasolle.

Määrittelemme tietoturvamallin vanhempi-lapsi-relaatiot kuvan 10 mukaisesti



Kuva 10. Järjestelmän vanhempi-lapsi-suhteet.

Järjestelmässä keskusinstanssissa eri asiakkuudet on eristetty erillisiin säikeisiin ja tietokantoihin. Jokainen asiakkuus luo oman vanhempi-lapsi-rakenteensa. Tämän lisäksi määritämme, että lapsi saa puhua vain vanhemmalleen tai tämän vanhemmalle sekä poikkeusmatriisissa mainituille uhkatietopalveluille sekä mahdollisille taustajärjestelmille.

Lopullinen tietoturvamallimme on siis

- tietyn tietoturvatason kohde ei saa lukea eikä kirjoittaa sitä ylemmälle tasolle
- kohde ei saa kirjoittaa alemmalle tietoturvatasolle, jos sillä on pääsy myös korkeammalle tietoturvatasolle pois lukien kohde itse
- harkinnanvaraiset pääsyoikeudet määritellään erillisellä matriisilla.
- lapsi saa puhua vain vanhemmalleen ja tämän vanhemmalle.

4.3 Tietoturvariskien käsittely

Riskin tunnistamisen jälkeen sitä voidaan käsitellä neljällä eri tavalla. Riski voidaan joko hyväksyä tai siirtää tai riskiä voidaan välttää tai vähentää. Riskiä voidaan siirtää organisaation ulkopuolelle esimerkiksi ostamalla vakuutuksia tai siirtämällä vastuuta toiseen

järjestelmään. Riskiä voidaan myös välttää jättämällä asioita tekemättä, jos niiden luoma riski on organisaation mielestä liian suuri. Mikäli riskiä ei voida siirtää tai välttää, täytyy se lopulta aina hyväksyä ja tämän takia riskiä pyritäänkin usein vähentämään hyväksyttävälle tasolle. Riskiä voidaan vähentää toteuttamalla yksi tai useampi kontrolli. Erilaisia kontrolleja voivat olla esimerkiksi palomuurit, IDS- ja IPS-järjestelmät tai käyttäjien kouluttaminen. (15, s. 97-98.)

Riskiä ei siis voida poistaa kokonaan, vaan sitä voidaan ainoastaan vähentää hyväksyttävälle tasolle (15, s. 93). Riskinhallintaa ja riskien käsittelyä eniten rajoittava tekijä on riskinhallintaan käytettävässä oleva raha. Mikäli riskin hallinta tulee kalliimmaksi kuin riskin toteutuminen ei organisaation kannata yrittää enää hallita riskiä vaan hyväksyä se (15, s. 93). Täten riskin hallitsemiseksi ehdotettujen toimien tulee olla suhteessa riskin suuruuteen. Pyrimme pitämään ehdottamamme kontrollit toteuttamiskelpoisina ja suhteellisina arvioituun riskiin, mutta jokainen organisaatio joutuu kuitenkin arvioimaan riskin vaikutukset heidän ympäristössään itse ja tekemään päätöksen kontrollien toteuttamisesta oman arvionsa perusteella.

Aloitamme havaitsemiemme riskien käsittelyn suurimmasta ja etenemme kohti pienintä. Samalla keräämme samankaltaiset riskit yhdeksi kokonaisuudeksi, johon voimme vaikuttaa samoilla hallintakeinoilla.

Sisäisessä käytössä olevien järjestelmien käyttäjä- tai pääsynhallinta ei välttämättä aina ole tarpeeksi korkealla tasolla, jotta palveluita voitaisiin avata suoraan asiakkaille. Pitkäaikainen ratkaisu tähän ongelmaan olisi tietysti kehittää kyseisiä järjestelmiä siten, että ne pystyisivät tarjoamaan tarvittavat käyttöoikeudet ja niiden rajaukset, mutta tämä on helposti hyvin kallis ja aikaa vievää projekti. Lyhytaikainen ja huomattavasti halvempi vaihtoehto olisi kehittää API-välityspalvelin, joka suodattaa vastauksista pois luottamuksellista tietoa sisältävät kentät ja toteuttaa tehokkaat käyttäjäntunnistusmenetelmät. Tällä vältämme riskiä poistamalla tiettyjä ominaisuuksia käytöstä ja siirrämme sitä toisen järjestelmän hallittavaksi. Samalla kokonaisriski kuitenkin laskee.

SOAR-järjestelmänintegraatioiden toteutuksissa voi hyvinkin olla haavoittuvuuksia, jotka päästävät käyttäjän ohittamaan järjestelmän rajoituksia ja ajamaan kyselyitä SOAR-koneen oikeuksilla ja pääsyllä eri ympäristöihin. Voimme vähentää riskiä sallimalla palo-

muurisäännöin SOAR-palvelimelle pääsyn vain niihin ympäristöihin, joihin sen on välttämätöntä saada yhteys. Tämä ei kuitenkaan ole aina riittävä toimenpide riskin pienentämiseksi, varsinkin jos kyseisellä palvelimella tarvitsee olla pääsy useampaan kuin yhteen ympäristöön. SOAR-järjestelmässä eri asiakkuudet on kuitenkin eristetty toisistaan siten, että ne eivät pääse käsiksi toistensa tietoihin. Voimme siten vähentää riskiä vielä enemmän ohjaamalla kaikki asiakasympäristöihin menevät yhteydet käänteisenvälityspalvelimen kautta. Tämä välityspalvelin tarkistaa yhteyksien oikeudet joko käyttäjätunnuksin tai sertifikaatein.

Mikäli käyttäjä kuitenkin pystyy kontrolloimaan SOAR-palvelinta jonkin tuntemattoman haavoittuvuuden avulla enemmän kuin hänen pitäisi, pystyy hän jaetussa ympäristössä todennäköisesti kuitenkin lukemaan myös muiden asiakkuuksien salaisuuksia. Tämän takia on järjestelmässä järkevää jakaa eri asiakkuudet erillisille palvelimille. Tämä on ikävä kyllä ominaisuus, jota useimmat SOAR-järjestelmät eivät tue, joten vaihtoehtoisesti voimme hyväksyä riskin siitä, että käyttäjä kykenisi äärimmäisissä tilanteissa laajentamaan käyttöoikeuksiaan SOAR-järjestelmässä ja tämän avulla saavuttamaan pääsyn myös ympäristöihin, joihin hänellä ei kuuluisi pääsyä olla. Tämän riski on kuitenkin ainakin MSSP:n tapauksessa liian suuri hyväksyttäväksi, joten tämä on otettava huomioon jo tuotetta valittaessa.

Pystymme vähentämään riskiä erilaisista sivukanavahyökkäyksistä, jotka hyväksikäyttävät joko ajoitusta tai yhteyksien IP-osoitetietoja reitittämällä kyselyt pilvipalvelussa sijaitsevien välityspalvelimien kautta. Nämä välityspalvelimet peittävät alkuperäisen IP-osoitteen ja aiheuttavat kyselyn kestoon satunnaista viivettä, kun paketit reititetään välityspalvelimen kautta.

Jotkin tietoturvaloukkauksen tutkimiseksi tehtävät kyselyt saattavat sisältää luottamuksellista tietoa, eivätkä nämä kyselyt tietenkään saisi päätyä kolmansille osapuolille. SOAR-järjestelmiin rakennetaan kuitenkin jatkuvasti uusia julkisia integraatioita, eikä organisaatio välttämättä ehdi ennalta tarkistamaan jokaisen palvelun luotettavuutta. Voimme kuitenkin rajata SOAR-järjestelmän pääsyä julkisiin palveluihin palomuurisäännöin ja välityspalvelimen avulla. Palomuurisäännöt estävät järjestelmän suoran pääsyn internettiin ja välityspalvelin sallii yhteydet ainoastaan luotettuihin palveluihin.

Yhteyksien salakuuntelua ja erilaisia man-in-the-middle-hyökkäyksiä pystymme estämään tehokkaasti käyttämällä päästä päähän salausta kaikissa yhteyksissä. Nykyisin yhteydet voidaan salata tehokkaasti ja helposti käyttämällä SSL-tekniikkaa. SSL perustuu epäsymmetriseen julkisen avaimen salaukseen (23). Epäsymmetrisissä salaus menetelmissä viestin salaamiseen ja purkamiseen käytetään kahta eri avainta, jotka liittyvät toisiinsa matemaattisesti, mutta eivät ole identtisiä (24). Julkisella avaimella salatun viestin voi avata vain siihen liittyvällä yksityisellä avaimella eikä yksityistä avainta voida helposti laskea vastaavasta julkisesta avaimesta (24). SSL-järjestelmää suunniteltaessa on kuitenkin käytettävä tarpeeksi suuria vähintään 3072-bittisiä avaimia, suojeltava julkiseen avaimeen liittyvää yksityistä avainta (23, 25). Lisäksi avainketju täytyy varmentaa ja avaimet hankkia luotettavasta lähteestä. (23.)

Turvallisen järjestelmän toteuttamisessa tulisi käyttää turvallisia protokollia, kuten TLS v1.1 tai TLS v1.2. Salausalgoritmin tulisi olla vähintään 128-bittinen AES. Lisäksi Perfect Forward Security -protokollan käyttäminen on suositeltavaa. (23; 25.)

Järjestelmän pääsynhallintaan olisi hyvä lisätä usean tekijän tunnistautuminen, jolloin käyttäjänimen ja salasanan lisäksi järjestelmään kirjautumiseen tarvitaan vielä jokin kolmas komponentti. Lisäksi on suositeltavaa rajata järjestelmän kirjautumisruutuun pääsy vain luotettuihin IP-osoitteisiin palomuurisäännöin sekä erillisellä Single Sign-On (SSO) -tekniikkaa hyväksikäyttävällä käänteisellä välityspalvelimella. Näin tuotteen kirjautumisruutu eristetään tehokkaasti vain järjestelmään oikeutetuille henkilöille verkko- ja sovellustasolla. SSO:n tarkoitus on pakottaa käyttäjä kirjautumaan vain kerran, jonka jälkeen SSO-palvelu huolehtii käyttäjän todentamisesta. SSO:ta käyttämällä pystymme keskittämään kriittiset kirjautumispalvelut yhteen järjestelmään ja helpotamme niiden ylläpitoa. Samalla parannamme käyttäjän käyttökokemusta.

Eristämällä SOAR-järjestelmään pääsyn ja siihen kirjautumisen jo todennetuille käyttäjille vähennämme myös tehokkaasti riskiä siitä, että oikeudeton käyttäjä pääsisi kontrolloimaan ympäristöä tai ympäristöjä, joihin hänellä ei ole oikeuksia. Lisäksi päätöksemme jakaa järjestelmä erillisille palvelimille asiakkuuksien perusteella mahdollistaa pääsyn sallimisen keskitettyyn näkymään vain hyvin rajatulle joukolle. Tämä vähentää altistumistamme siinä tapauksessa, että jokin järjestelmän osa onnistutaan murtamaan.

Erilaisten injektiohyökkäyksien estämisessä voimme joko luottaa SOAR-tuotteen omaan kykyyn siistiä sisään tulevasta tiedosta pois potentiaaliset hyökkäykset tai asentaa SOAR:n ja integraatiojärjestelmien väliin Web-sovelluspalomuurin. Injektiohyökkäyksissä hyökkääjä pyrkii sisällyttämään haitallista koodia tai komentoja normaalin datan sekaan siinä toivossa, että ohjelmisto ei validoi ja siisti dataa riittävissä määrin. Onnistuneen injektiohyökkäyksen seurauksena järjestelmä suorittaa komentoja joita sen ei ollut tarkoitus suorittaa tai vastaavasti tarjoaa käyttäjälle suoritettavaa koodia jota sen ei ollut tarkoitus tarjota. (26.) Web-sovelluspalomuuri on tietynlainen käänteinen välityspalvelin, joka pyrkii havaitsemaan HTTP-liikenteestä tunnettuja injektiohyökkäyksiä ja estämään niitä (27).

Koska SOAR-järjestelmämme on jo hyvin eristetty ja siihen kohdistuvat käyttöoikeudet on rajattu hyvin pienelle joukolle, on siihen suoraan kohdistuvan tai sitä hyväksikäyttävän palvelunestohyökkäyksen todennäköisyys hyvin pieni. Mahdollisesta hyökkäyksestä aiheutuva haitta on myös lähinnä väliaikaista eikä todennäköisesti aiheuta suuria menetyksiä. Järjestelmän käyttämiin verkkoyhdyskäytäviin kohdistuvien hyökkäyksien toteamme olevan tämän järjestelmän ulkopuolella. Voimme siis hyväksyä riskin palvelunestohyökkäyksistä sellaisenaan, koska aiheutuva riski on niin pieni verrattuna sen ehkäisemiseen tarvittaviin resursseihin.

Kertauksena tarvitsemamme kontrolli toimenpiteet ovat seuraavat

- Palomuurit estämään epätoivottuja yhteyksiä sekä SOAR-järjestelmään että siitä poispäin.
- Käänteisiä välityspalvelimia tarjoamaan sovellustason palomuureja sekä rajoittamaan pääsyä järjestelmään tiukemmin kuin palomuurisäännöin olisi mahdollista.
- Välityspalvelimen salliaksemme pääsyn luotettavaksi katsomiimme palveluihin SOAR-järjestelmästä.
- Välityspalvelimen vahvalla tunnistuksella toimimaan yhdyskäytävänä asiakas kohtaisiin ympäristöihin.
- HTTPS / TLS v1.2 salausmenetelmän kaikkiin järjestelmässä tehtäviin kyselyihin vähintään 3072 bittisillä avaintiedostoilla sekä vähintään 128 bittisellä AES salauksella.
- Sovellustasolla tietoa suodattavan API-välityspalvelimen poistamaan mahdollisesti luottamuksellista tietoa sisältävät kentät vastauksista hakuihin jotka tulevat organisaation ulkopuolelta.

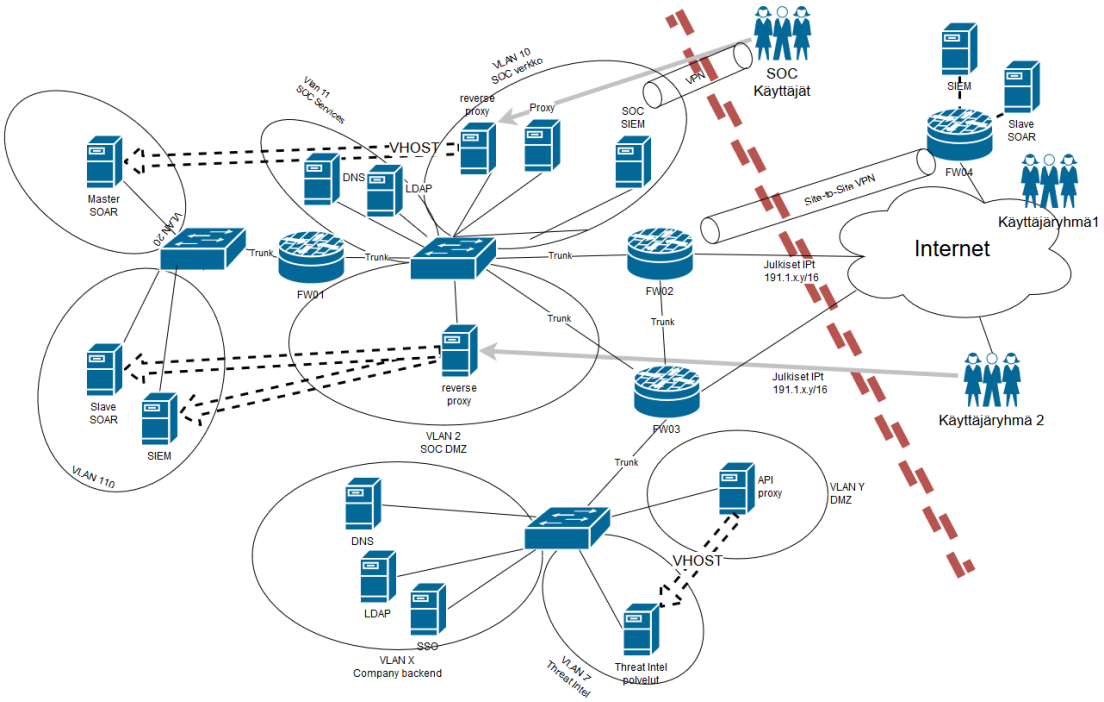
- Pilvipalvelussa ajettavia välityspalvelimia peittämään mahdollisten arkaluontoisten kyselyiden lähteen.

5 SOAR-integraatioyhteyksien toteuttaminen tietoturvallisesti

5.1 Esimerkkiympäristö

Integraatioyhteyksien turvaaminen alkaa turvallisen verkkoympäristön suunnittelusta. Turvallinen ja mahdollisimman pitkälle eristetty verkkoinfrastruktuuri estää monet mahdolliset hyökkäysvektorit jo ennalta. Koska SOAR käsittelee tietoa, joka on alun perin sijainnut rajoitetussa verkossa, on luontevaa asentaa myös SOAR samankaltaiseen verkkoon.

Kuvassa 11 on kuvattu yksi esimerkki siitä, miten SOAR-järjestelmän verkko vois olla toteutettu.



Kuva 11. Suunniteltu verkko SOAR:lle

SOAR-järjestelmän isäntäpalvelin on eristetty palomuurin taakse, eikä sinne sallita suoria yhteyksiä kuin käänteisistä välityspalvelimista. Isäntäpalvelin itse voi ottaa yhteyksiä välityspalvelimen kautta internetin valkolistattuihin palveluihin, muihin ennalta sallittuihin tietoturvajärjestelmiin eri asiakasympäristöissä ja yrityksen omiin järjestelmiin. Lisäksi isäntäpalvelin saa muodostaa suoria yhteyksiä muihin SOAR-palvelimiin sekä SOC:n omiin taustajärjestelmiin, mutta sisään tulevat uudet yhteydet on silti estetty.

Käyttäjäryhmä 2 kuvastaa MSSP:n asiakasta, joka on ostanut SOAR:n palveluna. Heitä varten on varattu julkinen IP-osoite ja DNS-nimi. Tämä julkinen osoite on ohjattu NAT-tekniikalla DMZ-verkossa sijaitsevalle käänteiselle välityspalvelimelle, jonka kautta käyttäjät pääsevät heille luotuun ympäristöön. Pääsyä välityspalvelimelle rajataan palomuurisäännöin niin, että vain tietyistä asiakkuuteen liitetyistä IP-osoitteista sallitaan liikenne palomuurin läpi. Käänteinen välityspalvelin varmentaa käyttäjän oikeuden käyttää SOAR-järjestelmää organisaation LDAP- tai SSO-palvelimilta. Käyttäjäryhmä 2:n SOAR-järjestelmä saa ottaa yhteyksiä SOC:n taustajärjestelmiin, organisaation omaan SIEM-järjestelmään ja välityspalvelimen kautta uhkatietojärjestelmän API-välityspalvelimeen sekä valkolistattuihin internetpalveluihin. Näille käyttäjille sallitaan SOAR-järjestelmässä vain rajalliset oikeudet.

Käyttäjäryhmä 1 kuvastaa MSSP:n asiakasta, joka haluaa ylläpitää itse omaa SOAR-järjestelmää, jonka MSSP on hänelle myynyt. Asiakas on tiiviissä yhteistyössä päivittäin SOC:n kanssa tietoturvahälytysten tutkimisessa. Tämän järjestelmän pääsynrajauksesta vastaa asiakas itse seuraavin poikkeuksin. Isäntä SOAR:n on sallittu muodostaa VPN-tunnelin kautta suora yhteys asiakkaan SOAR-järjestelmään. Asiakkaan SOAR-järjestelmälle on järjestetty pääsy uhkatietopalvelun API-välityspalvelimelle ja muut yhteydet MSSP:n verkkoon on estetty.

Lisäksi SOAR-järjestelmän luomat lokiviestit välitetään kaikki ulkoiselle auditointipalvelimelle. Lokit voidaan välittää tältä palvelimelta takaisin organisaation SIEM-järjestelmään, jossa niiden avulla voidaan kehittää sääntöjä, jotka havaitsevat SOAR-järjestelmässä tapahtuvat tietoturvaloukkaukset.

5.2 Hallintamenetelmien toteuttaminen

Käytimme esimerkkiympäristössämme pääsynrajaamiseen palomuureja, välityspalvelimia ja VLAN-tekniikkaa verkkojen eristämiseen toisistaan. SOAR-järjestelmän eri osat on eristetty SOC:n normaalista verkosta välityspalvelimia lukuun ottamatta kokonaan. Pääsynhallinnan takia verkossa on erillinen DMZ-alue, joka sisältää palvelimet joihin ohjataan liikennettä organisaation ulkopuolelta. Näin emme joudu avaamaan pääsyä kriittisempiin verkkoihin palveluiden tarjoamiseksi.

VLAN on kytkimissä toteutettu tekniikka, joka eristää eri VLAN:ssa laitteet toisistaan OSI-mallin kakkoskerroksella (28). VLAN:t määritellään kytkimen sisällä, ja kytkin päättää niiden mukaan, mihin portteihin paketteja voidaan lähettää (28). Oikein konfiguroituna VLAN-tekniikka tarjoaa tehokkaan tavan eristää eri verkoissa olevat koneet toisistaan ja sallii samalla siirtää kaiken liikenteen erillisten runkoporttien kautta palomuurille, reitittimelle tai toiselle kytkimelle.

Rajaamme verkkoliikennettä palomuuriratkaisuilla estääksemme pääsyn eri verkosta toiseen. Kaikki esimerkissä esittelemämme palomuurit toimivat valkolistausperiaatteella, jossa kaikki liikenne on lähtökohtaisesti kiellettyä, ellei palomuuriin ole konfiguroitu sitä sallivaa sääntöä. Lisäksi voimme käyttää niin kutsuttuja seuraavan sukupolven palomuuriratkaisuja, jotka analysoivat IP-osoitteiden ja porttien lisäksi palomuurin läpi kulkevaa dataa (29). Nämä kehittyneemmät palomuurit kykenevät tunnistamaan liikennettä protokolla tasolla ja täten havaitsemaan myös sellaista liikennettä, joka pyrkii naamioitumaan joksikin muuksi käyttämällä esimerkiksi epästandardeja portteja, kuten HTTP:n käyttämää TCP/80 porttia johonkin aivan muuhun tarkoitukseen. (29).

Lisäksi eristämme SOAR-järjestelmänkäyttöliittymän käänteisenvälityspalvelimen taakse. Välityspalvelimena voi toimia esimerkiksi Nginx Reverse Proxy, joka muokkaa HTTP-pyynnön otsakkeita ja välittää kyselyt tämän jälkeen kohdepalvelimelle. Nginx on avoin HTTP-palvelin ja HTTP-välityspalvelinohjelmisto, johon on saatavilla huomattava määrä erilaisia turvallisuutta lisääviä lisäosia. Esimerkkiympäristössä asentaisimme nginx-palvelimelle modsecurity-lisäosan toteuttamaan tarvitsemamme sovelluspalomuurin ja OpenID Connect tai SAML-lisäosan huolehtimaan käyttäjientunnistamisesta SSO-tekniikkaa hyväksikäyttämällä.

Nginx modsecurity -lisäosa on avoimenlähdekoodin web-sovelluspalomuuri, joka tunnistaa ja estää suuren määrän OSI-mallin 7 tason heikkouksia hyväksikäyttävistä hyökkäyksistä. Näitä ovat esimerkiksi erilaiset injektiohyökkäykset kuten SQL-injektiot ja paikallisten tiedostojen sisällyttäminen sivuun. Modsecurity havaitsee ja estää myös monia XSS-hyökkäyksiä. (30.) SSO-kirjautumisen voimme toteuttaa joko OpenID Connect- tai SAML-tekniikkaa käyttämällä.

OpenID liimaa perinteisen oAuth2.0-protokollalla toteutetun käyttäjän kirjautumisen ja JSON Web Token -merkit siten, että Nginx-palvelimen läpi voidaan päästää vain kirjautuneet käyttäjät ja ohjata kirjautumattomat kirjautumissivulle (31). SAML, eli Security Assertion Markup Language on toinen tapa toteuttaa samanlainen toiminta. SAML itsessään on määritelty kieli, jota useat tunnistautumisjärjestelmät tukevat. (32.)

OAuth2 on lupajärjestelmä, joka mahdollistaa kolmannen osapuolen ohjelmiston saada oikeudet järjestelmään käyttäjän valtuutuksella. OAuth2-järjestelmässä käyttäjä saa kirjautumisen jälkeen erillisen valtuutusmerkin, jonka avulla hän voi tunnistautua muille järjestelmille. Käyttäjä voi myös välittää valtuutuksensa toiselle palvelulle, joka voi tämän jälkeen toimia hänen sijastaan. (33.)

Koska haluamme sallia SOAR-järjestelmälle hyvin rajatun pääsyn internetiin ohjaamme kaiken ulos menevän verkkoliikenteen välityspalvelimen kautta. Tämän välityspalvelimen pääasiallinen tehtävä on estää pääsy sivustoille, joihin emme luota, ja sallia vain ne sivustot, jotka on erikseen mainittu. Välityspalvelin voidaan toteuttaa esimerkiksi Squid-ohjelmistolla.

Squid on täysiverinen välitys- ja välimuistipalvelin ohjelmisto, jonka avulla voidaan ohjata ja välittää http-, FTP -ja muita yleisiä web-protokollia. Squid voi toimia myös SSL-yhteyksien kanssa. Lisäksi Squid tarjoaa kattavat pääsynhallinta- ja valvontamenetelmät liikenteestä, joka välityspalvelimen läpi kulkee. (34.)

Tarjotaksemme uhkatietopalvelua myös muille kuin SOC:n omalle henkilöstölle, kehitämme API-välityspalvelimen, joka vaatii vahvan kirjautumisen erilaisten API-kutsujen tekemiseksi. Tämän kaltaisten API-välityspalvelimien kehittämiseksi on olemassa useita

eri ratkaisuja. Tämä uusi API-palvelin toteuttaa vahvan käyttäjäntunnistuksen esimerkiksi oAuth2-tekniikkaa hyväksikäyttämällä ja rajaa käyttäjille suunnatuista vastauksista mahdollisesti luottamuksellista tietoa sisältävät kentät.

6 Yhteenveto

Olen tämän työn aikana esitellyt lukijalle useita eri järjestelmiä ja sovelluksia, joita SOC:n henkilöstö käyttää päivittäisessä työssään. Järjestelmien suuri määrä ja jatkuvasti monimutkaistuva tietoturvamaisema aiheuttaa SOC:lle jatkuvasti kasvavan ongelman osaavan ja motivoituneen henkilöstön saamisesta ja pitämisestä organisaatiossa. Lisäksi vanhoilliset toimintatavat ja suuri manuaalisen työn määrä aiheuttavat palvelun laadun heikkenemistä ja kustannusten kasvua.

SOAR-järjestelmä tarjoaa potentiaalisen ratkaisun useisiin SOC:n kohtaamista ongelmista ja lupaa tehostaa ja automatisoida nykyisiä työtapoja ja menetelmiä. SOAR-järjestelmänä tuo kuitenkin organisaatiolle omat tietoturvahaasteet, joita olemme kartoittaneet neljännessä kappaleessa hyvinkin kattavasti. Kuten esitetystä ratkaisumallista voidaan huomata eivät nämä haasteet kuitenkaan ole ylitsepääsemättömiä ja esittämäni ratkaisun tulisi tarjota vähintään hyvä aloitus tietoturvallisen SOAR-järjestelmän pystyttämiseksi.

Tässä työssä esitellyistä hallintamenetelmistä suuren osan toteuttamalla pitäisi SOAR:n aiheuttaman riskin alentua huomattavasti. Kuten aina riskien hallinnassa ei tämä prosessi ole missään nimessä valmis. SOAR päätöksen ja ensimmäisen toteutuksen jälkeen on tärkeätä arvioida järjestelmään kohdistuvat riskit uudestaan ja arvioida jo toteutettujen hallintamenetelmien tehokkuutta sekä tarvetta mahdollisille lisäkontrolleille. Tämän työn pitäisi kuitenkin tarjota hyvä ymmärrys käytettävissä olevista tekniikoista ja siitä, mistä SOAR-järjestelmän tietoturvan miettiminen kannattaa aloittaa.

Riski on aina organisaatio kohtaista, eikä tässä työssä esitellyistä keinoista kaikki tietenkään sovellu organisaation nykyiseen infrastruktuuriin. Toteuttavan organisaation onkin tärkeätä tehdä oma riskikartoituksensa esiteltyjen menetelmien avulla sekä toteuttaa ne

hallintamenetelmät, jotka vähentävät heidän kokonaisriskiänsä SOAR-järjestelmän suhteen. Osa hallintakeinoista on varmasti jo käytössä useissa organisaatioissa, mutta myös näiden käytössä olevien tekniikoiden kriittinen tarkastelu on tärkeätä.

Kaiken kaikkiaan työn aikana onnistuttiin toteamaan SOAR-järjestelmän olevan varteenotettava vaihtoehto. SOAR:n aiheuttavan selviä mutta hallittavissa olevia tietoturvariskejä. Lisäksi esiteltiin varteenotettavia vaihtoehtoja riskien hallitsemiseksi. Esiteltyä suunnitelmaa soveltamalla organisaatio kykenee ottamaan SOAR-järjestelmän käyttöön tietoturvallisesti ainakin integraatioyhteyksien osalta.

Lähteet

1 Muniz, Joseph. McIntyre, Gary. AlFardan, Nadhem. 2016. Security Operations Center Building, Operating, and Maintaining Your SOC. United States: Cisco Press

2 Managed Security Service Provider (MSSP). Verkkoaineisto. <https://www.gartner.com/it-glossary/mssp-managed-security-service-provider>. Luettu 30.7.2018.

3 Miller, Matt. 2017. What is an MSSP (Managed Security Services Provider). Verkkoaineisto. <https://www.beyondtrust.com/blog/mssp-managed-security-services-provider/>. Luettu 30.7.2018.

4 How to build a Security Operations Center (On a budget). Verkkoaineisto. <https://www.alienvault.com/resource-center/ebook/building-a-soc/soc-team>. Luettu. 30.7.2018.

5 Dorigo, Sander. 2012. Security Information and Event Management. Radbound University. Nijmegen.

6 Bhatt, Sandeeb. Manadhata, Pratyusa K. Zomlot, Loai. 2014. The Operational Role of Security Information and Event Management Systems. Verkkoaineisto. IEEE Security & Privacy. <https://doi.org/10.1109/MSP.2014.103>. Luettu 2.8.2018.

7 Lord, Nate. 2018. What is Endpoint Detection and Response? A Definition of Endpoint Detection & Response. DigitalGuardian. Verkkoaineisto. <https://digitalguardian.com/blog/what-endpoint-detection-and-response-definition-endpoint-detection-response>. Luettu 22.9.2018.

8 Petters, Jeff. 2018. Endpoint Detection and Response (EDR): Everything You Need to Know. Verkkoaineisto. <https://blog.varonis.com/endpoint-detection-and-response-edr/>. Luettu 22.9.2018.

9 Reviews for Endpoint Detection and Response Solution. Verkkoaineisto. <https://www.gartner.com/reviews/market/endpoint-detection-and-response-solutions>. Luettu 22.9.2018.

10 What Is Endpoint Detection and Response (EDR)? Verkkoaineisto. <https://www.carbonblack.com/resources/definitions/what-is-endpoint-detection-and-response/>. Luettu 22.9.2018.

11 Mavroeidis, Vasileios. Bromander, Siri. Cyper Threat Intelligence Model: An Evaluation of Tazonomies, Sharing Standards, and Ontologies within Cyper Threat Intelligence. University of Oslo. Norway. Verkkoaineisto. <https://www.duo.uio.no/handle/10852/58492>. Luettu 22.9.2018.

12 Chismon, David. Ruks, Martyn. Threat Intelligence: Collecting, Analysing, Evaluating. Verkkoaineisto. <https://www.ncsc.gov.uk/content/files/protected_files/guidance_files/MWR_Threat_Intelligence_whitepaper-2015.pdf>. Luettu 22.9.2018.

13 Features of MISP, the open source threat sharing platform. Verkkoaineisto. <https://www.misp-project.org/features.html>. Luettu 22.9.2018.

14 Neiva, Claudio. Lawson, Craig. Bussa, Toby. Sadowski, Gorka. 2017. Innovation Insight for Security Orchestration, Automation and Response. Verkkoaineisto. Gartner Inc. <https://www.gartner.com/doc/3834578/innovation-insight-security-orchestration-automation>. Luettu 24.9.2018.

15 Shon, Harris. 2013. All in one CISSP Exam Guide Sixth Edition. United States: McGraw-Hill

16 Andreasson, Ari. Koivisto, Juha. 2013. Tietoturvaa Toteuttamassa. Tallinna: Tietosanoma Oy

17 Rousku, Kimmo. 2017. Ohje riskienhallintaan Vahti 22/2017. Valtiovarainministeriö

18 Shostack, Adam. 2014. Threat Modeling: Designing for Security. e-kirja. John Wiley & Sons, Inc. Indiana

19 Freud, Jack. Jones, Jack. 2015. Measuring and Managing Information Risk a Fair Approach. UK: Elsevier Inc.

20 Pham, Thu. 2017. Verkkoaineisto. Cisco. <https://duo.com/blog/nist-update-passphrases-in-complex-passwords-out>. Luettu 10.10.2018.

21 DEPARTMENT OF DEFENSE TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA. 1985. Verkkoaineisto. <https://csrc.nist.gov/csrc/media/publications/white-paper/1985/12/26/dod-rainbow-series/final/documents/std001.txt>. Luettu 10.10.2018.

22 Bell, D.E. La Padula, L.J. 1976. SECURE COMPUTER SYSTEM: UNIFIED EXPOSITION AND MULTICS INTERPRETATION. Verkkoaineisto. <https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/bell76.pdf>. Luettu 12.10.2018.

23 SSL Best Practices: a Quick and Dirty Guide. 2015. Verkkoaineisto. <https://www.ssl.com/guide/ssl-best-practices-a-quick-and-dirty-guide/>. Luettu 17.10.2018.

24 What is Public-key Cryptography? Verkkoaineisto. <https://www.globalsign.com/en/ssl-information-center/what-is-public-key-cryptography/>. Luettu 17.10.2018.

25 Kryptografiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen - kansalliset suojaustasot. 2015. Verkkoaineisto. Viestintävirasto. <https://www.viestintavirasto.fi/attachments/tietoturva/Kryptografiset_vahvuusvaatimukset_-_kansalliset_suojaustasot.pdf>. Luettu 26.10.2018.

26 Injection Flaws. 2015. Verkkoaineisto. <https://www.owasp.org/index.php/Injection_Flaws>. Luettu 18.10.2018.

27 Web Application Firewall. 2016. Verkkoaineisto. <https://www.owasp.org/index.php/Web_Application_Firewall>. Luettu 18.10.2018.

28 Hucaby, David. McQuerry, David. 2002. VLANs and Trunking. Verkkoaineisto. Cisco Press. <https://www.ciscopress.com/articles/article.asp?p=29803>. Luettu 18.10.2018.

29 Next-Generation Firewalls (NGFWs). Verkkoaineisto. Gartner Inc. <https://www.gartner.com/it-glossary/next-generation-firewalls-ngfws>. Luettu 21.10.2018.

30 Memon, Faisal. 2017. Compiling and Installing ModSecurity for NGINX Open Source. Verkkoaineisto. NGINX Inc. <https://www.nginx.com/blog/compiling-and-installing-modsecurity-for-open-source-nginx/>. Luettu 21.10.2018.

31 Grilly, Liam. 2017. Verkkoaineisto. NGINX Inc. <https://www.nginx.com/blog/authenticating-users-existing-applications-openid-connect-nginx-plus/>. Luettu. 21.10.2018.

32 Cantor, Scott. Kemp, John. Philpott, Rob. Maler, Eve. 2015. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. Verkkoaineisto. <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>. Luettu 25.10.2018.

33 D. Hardt, Ed. 2012. The OAuth 2.0 Authorization Framework. Verkkoaineisto. IETF. <https://tools.ietf.org/html/rfc6749>. Luettu 21.10.2018.

34 Squid - Proxy Server. Verkkoaineisto. Ubuntu Documentation. <https://help.ubuntu.com/lts/serverguide/squid.html.en>. Luettu 21.10.2018.