SOC-työntekijän päiväkirja

Juho Mikkonen

Haaga-Helia ammattikorkeakoulu

Amk-opinnäytetyö

2021

Tradenomin tutkinto

Tiivistelmä

Tekijä(t)
Juho Mikkonen

Tutkinto
Tradenomi

Raportin/Opinnäytetyön nimi
SOC-työntekijän päiväkirja

Sivu- ja liitesivumäärä
66

Opinnäytetyö on päiväkirjamainen kooste SOC-työntekijän päivittäisistä työtehtävistä, joita opiskelija on päivittäin kohdannut. Näitä merkintöjä analysoidaan laajemmin viikon päätteeksi 8 kertaa. Viikoittaiset analyysit reflektoivat aiempaa toimintaa, missä on samalla mahdollista tarkastella vaihtoehtoisia toimintatapoja ja ratkaisumalleja verrattuna aiempiin työskentelymenetelmiin seurantaviikkojen aikana.

Opiskelija on opinnäytetyön aikana töissä PK-yrityksessä, jossa on noin 80 henkilöä töissä, ja yritys tuottaa päätuotteenaan asiakkaille toimitettavia virtuaalipalvelinratkaisuja, ja tarkennettuna näitä toimitetaan IaaS-palvelumallilla, jossa vastuu palvelusta jaetaan asiakkaiden ja tuottajan välillä mallin luonteen mukaisesti. Palvelu on samalla julkinen pilvipalvelu, eli se toimitetaan julkisen internetin kautta loppuasiakkaalle.

Työtehtävät koostuvat pääosin kahdentyyppisistä asioista, mitkä voidaan jakaa Python-kielellä tapahtuvaan sovelluskehitykseen, mikä tukee SOC-työskentelyä, että tehostaa SOC-työskentelyn prosesseja, ja toiseksi työtehtävät koostuvat SOC-tehtävissä erilaisten asiakkaiden aiheuttamien tietoturvapoikkeamien ratkaisuista. Jälkimmäisenä mainittuja asioita käsitellään opinnäytetyön aikana vähemmän, sillä ne ovat luonteeltaan toistuvia työtehtäviä, ja pääosin niiden ratkaisumalli on perimmäiseltä luonteeltaan aina samanlainen.

Työtehtävien lopputuloksena syntyy useimmiten joko täysin käyttövalmiita sisäisiä tuotteita, palveluita tai parannuksia aiempiin menetelmiin. Joissakin tapauksissa lopputuloksena on prosessien kehittyminen. Työn aikana ei siis seurata mitään suurta yksittäistä projektia, vaan kyse on enimmäkseen työhön liittyvien prosessien kehittämistä joko suoraan tai epäsuorasti.

Asiasanat
Tietoturva, Linux, tietosuoja, turvallisuustoiminta

Sisällys

1 Johdanto

Tätä päiväkirja-tyypin opinnäytetyötä kirjoitetaan Haaga-Helian raportointiohjeen ja tietojenkäsittelyn koulutusohjelman päivätoteutuksen opinnäytetyön ohjeiden mukaisesti 06.12.2021 alkaen kappaleiden 1 ja 2 osalta, mutta varsinainen päiväkirjaosuus alkaa vasta 03.01.2022, ja päättyy arviolta huhtikuussa 2022. Aikatauluun voivat vaikuttaa esimerkiksi lomapäivien käyttäminen tai sairaspoissaolot. Merkintöjä on tarkoitus kirjoittaa työpäivisin, ja niiden koostetta on tarkoitus analysoida tarkemmin työviikon lopussa.

Työtehtävät koostuvat pääosin sovelluskehityksestä, valvonnasta, ja reagoinnista loppuasiakkaiden aiheuttamiin tietoturvapoikkeamiin, ja äärimmäisen harvoin maksuvälineriitatilanteiden ratkaisutilanteisiin, joten voi sanoa, että työtehtävät muistuttavat eniten turvallisuusoperaatiokeskuksessa työskentelyä, mistä käytetään englanniksi nimeä SOC (Security Operations Center). Virallinen työnimikkeeni on ”Fraud and Abuse Analyst” siitäkin huolimatta, että teen myös sovelluskehitystä, joka tietysti tukee muita työtehtäviä.

Raportoinnin kohteena olevan työpaikan tarjooman päätuote on virtuaalipalvelinten tuottaminen, mitä loppuasiakkaat käyttävät epähallinnoidusti eli työpaikka on julkisten IaaS-pilvipalveluiden tuottaja. Työtehtäväni on olemassa sen vuoksi, että loppuasiakkaat ovat sitoutuneet noudattamaan yrityksen käyttöehtoja, ja paikallisesti sovellettavaa lainsäädäntöä, joten on tärkeää katsoa, että mahdolliset valitukset ja poikkeamat, joita havaitaan tulevat käsitellyiksi. Työpaikassa on tällä hetkellä noin 80 henkilöä töissä, mistä valtaosa työskentelee Helsingissä pääkonttorilla, mutta osa työskentelee ulkomailla etätöissä.

Työn vaatimusten takia on tärkeää, että kyseisessä työtehtävässä hallitsee erinomaisesti sekä asiakaspalvelutaidot, että myös tekniset taidot työtehtävien toteuttamiseen itsenäisesti ilman erityistä esimiesohjausta.

Tietoperusta työssä perustuu sekä kirjalliseen tietoon, että organisaatiosta hankittuun tietoon, ja omaan vuosien varrella hankittuun kokemusperäiseen tietoon, ja aiempiin korkeakouluopintoihin henkilöstöhallinnosta, ja johtamisesta.

Kirjallista tietoa on tarkoitus käyttää päiväkirjamerkintöjen laajemmissa analyyseissä reflektoinnin pohjana. Ennakoivasti lupaavaksi materiaaliksi olen valinnut The Modern Security Operations Center (Muniz, Lakhani, Santos & Frost, 2021) kirjan, koska se vaikuttaa kattavan laajasti oman työni kuvan SOC-työntekijän tuottamista ydinpalveluista ja lupaa tarjoavan helposti sovellettavan mallin, joka pystyy vastaamaan käytännössä melkein mihin tahansa tietoturvauhkaan.

Toinen lupaavalta vaikuttava kirja eli Designing and Building Security Operations Center (Nathans, 2014) on taas valittu ainakin tarkasteltavaksi sillä perusteella, että kirja lupaa esitellä eräänlaisen pohjapiirroksen, jonka avulla rakennetaan yritykseen toimiva SOC-toiminto, joka suojelee työpaikan ulkoisia sidosryhmiä eli pääosin asiakkaita, ja sisäisiä sidosryhmiä eli muita organisaatioyksiköitä eli se voi toimia hyvin myös urakehityksen tukena.

Muut tärkeät ja oleelliset teokset ovat listattu tämän opinnäytetyön lähdeluettelossa, ja on tärkeää huomata, että nämä muut lähteet ovat huomattavasti tärkeämmässä roolissa opinnäytetyöhön liittyen viikkoanalyysien osalta, ja tämä johtuu käytännössä siitä, että työtehtäväni ovat tietyssä määrin ainutkertaisia, ja vain tietynlaiset rutiinit toistuvat, joten muutamia yksittäisiä teoksia ei voi käyttää työtehtävien analysointiin viikkoraporteissa. Tästä johtuen pyrin sisällyttämään aina useita lähteitä kerralla viikkoraportteihin mukaan.

1.1 Käsitteet

- AAA-periaate
    - Tulee sanoista authentication, authorization ja accounting. Tällä varmistetaan, että johonkin tietojärjestelmään kirjautuvat tahot kulkevat roolipohjaisen sisäänkirjautumismallin kautta niin, että kirjautumisesta jää jonkinlainen merkintä lokitietoihin tai lokitietokantaan.
- Headerit
    - Synonyymi otsaketiedoista, joita käytetään mm. sähköpostien ollessa kontekstina sähköpostiviestin reitin selvittämiseen vastaanottajan näkökulmasta.
- OSI-malli
    - Open Systems Interconnection -malli kuvaa tasoissa fyysisestä tasosta sovellustasolle ja useimmiten seitsemän eritellyn tason kautta tietojärjestelmiä. Alun perin mallin tarkoitus oli kuvata tietoverkkoja, mutta mallia käytetään nykyään soveltavasti.
- Security Operations Center
    - Turvallisuustoimintakeskus. Johdettu yksikkö organisaatiossa, joka vastaa esimerkiksi tietoturvauhkiin päivystäen.
- SOC
    - Lyhenne Security Operations Centeristä
- Tiketöintijärjestelmä
    - Asiakaspalvelussa käytettävä keskustelupohjainen viestintäjärjestelmä.
- Tilausansa
    - Huijaussivustoa muistuttava verkkosivusto, joka houkuttaa useimmiten käyttäjänsä tilaamaan luottokortillaan jotain arvokasta, mitä on vaikea irtisanoa.
- Turvallisuustoimintakeskus
    - Käännös Security Operations Centeristä.

2 Lähtötilanteen kuvaus

2.1 Oman nykyisen työn analyysi

Arkipäiväisesti pääasiassa valvon ja reagoin loppuasiakkaiden aiheuttamiin väärinkäyttötilanteisiin, joita asiakkaat aiheuttavat tahattomasti tai tahallaan, ja väärinkäyttötilanteiden havainnointi ennen reagointia voi tapahtua sisäisesti esimerkiksi niin, että verkkoa valvova henkilö havaitsee jostakin asiakkaan virtuaalipalvelimelta lähtevän runsain määrin liikennettä, mistä informoidaan minua suoraan, tai sopivalla hälytysjärjestelmällä, mistä alkaa tilanteen analyysi. Nämä ratkaistavat tilanteet syntyvät nimenomaan asiakkaiden omilla virtuaalipalvelimilla aina, mitä yritys tuottaa asiakkaiden tilattavaksi.

Ulkoisesti kaikki ilmoitukset saapuvat sähköpostilla käsiteltäviksi työssä asentamaani ja ylläpitämääni tiketöintijärjestelmään, missä on mahdollista keskustella myös loppuasiakkaan kanssa, ja tarvittaessa myös PGP-salauksen kera. Mitä enemmän, ja eri lähteistä eli uhkatietosyötteistä saapuvia ilmoituksia on, sitä helpompaa saadun tiedon varmentaminen on. Oleellista on, että juuri ulkoisissa ilmoituksissa kuvataan tarkasti haitallisen toiminnan tyyppi, havaitut kellonajat, ja mahdollisesti todisteena lokitiedostot, joissa näkyy konkreettisia jälkiä raportoidusta ei-toivotusta toiminnasta.

Yleisimpiin ulkoisesti raportoituihin väärinkäyttötilanteisiin kuuluvat mm.

- Roskapostitus
- Murtoyritykset muita verkkopalveluita kohtaan
- Huijaussivustot tai niitä muistuttavat tilausansat
- Tekijäinoikeusloukkaukset
- Haittaohjelmien komentopalvelimien ylläpito
- Lisenssirikkomukset

Tietoturvatapahtuman lähteestä tai havaintotapahtumasta riippumatta syntyy analyysin jälkeen väistämättä keskustelua asiakkaan kanssa tapauksesta, paitsi tietysti silloin, jos annetut tiedot voidaan todeta suoraan vääriksi. Vääräksi todettava tieto voisi koskea vaikkapa hyökkäystä, jonka lähde on väärennetty, kuten UDP—perusteisissa palvelunestohyökkäyksissä voi olla protokollan yhteydettömän luonteen vuoksi.

Tavoitteena työssäni on, että kaikenlainen haitallinen, lainvastainen, tai käyttöehtoja rikkova toiminta saadaan loppumaan, ja mahdollisimman nopeasti. Tämä voidaan toteuttaa kehottamalla loppuasiakasta toimimaan toisin antamalla asiakkaalle varoituksia asianmukaisten lisätietojen kera, joilla asiakkaan tulee itse korjata ongelmaa. Yrityksessä ei korjata asiakkaan omaa ongelmaa tämän jälkeen suoraan, vaan valvotaan, että raportoitu haitallinen toiminta on varmasti loppunut. Tämä johtuu tarjotun palvelun luonteesta, joka on hallinnoimaton (englanniksi: unmanaged) eli asiakas vastaa OSI-mallia referoiden

käyttöjärjestelmätasolta ylöspäin kaikesta itse, mitä hänen itse luomassaan virtuaalikoneessa tapahtuu, ja on sitoutunut olemaan aiheuttamatta käyttöehtosopimuksiin tai lakeihin kirjattuja rikkomuksia.

Varoituksiin ja ilmoituksiin reagoimatta jättäminen saattaa äärimmäisessä tilanteessa eli rikkomuksessa aiheuttaa asiakkuuden lopettamisen käyttöehtojen mukaisesti. Jokainen saapunut ulkoinen ilmoitus käsitellään tapauskohtaisesti keskustellen asiakkaan kanssa, ja perustuen siihen miten asiakas on menneisyydessä ratkaissut vastaavia mahdollisia tapauksia. Suurempien asiakkaiden kanssa voidaan sopia erityisistä kontaktimetodeista, joilla ilmoituksia toimitetaan.

Edellä mainituilla tavoilla käsitellään siis käytännössä kaikki muut, paitsi harvinaiset maksutapahtumiin liittyvien riitatilanteiden ratkaisut, jotka hoituvat käytössä olevien maksuvälittäjien tarjoamien portaalien, ja työkalujen avulla. Ennakoivasti epäilyttäviä asiakkuuksia voidaan tunnistaa valmiiksi koneoppimiseen perustuvan järjestelmän avulla, tai manuaalisesti, jotta potentiaalisiin maksuongelmiin voidaan puuttua ajoissa.

Esimerkki erittäin huolestuttavasta asiakkuudesta voisi olla vaikka sellainen, joka käyttää palvelua VPN-palvelun kautta salatakseen alkuperäisen IP-osoitteensa, ei toimita tarkkoja osoitetietoja itsestään ja jolla on maksuhistoriassaan monta epäonnistunutta maksua. Järjestelmä osaa poimia tutkittavaksi tämänkaltaisia löytöjä päivittäin.

Joskus ratkaisen myös tilanteita, joissa tulee tarpeen toteuttaa jonkin suomalaisen viranomaisen määräys, eli toteutan myös yrityksen viranomaisviestintää ja siihen liittyvien tietojen prosessointia.

Tämän lisäksi käytän itse tuottamiani tai itse ylläpitämiäni työkaluja työtehtävieni avustamiseen, minkä ytimessä toimii hallinnoimani Freescout-tiketöintijärjestelmä, siihen liittyvä Linux-pohjainen palvelininfrastruktuuri, ja tiketöintijärjestelmään toteutetut integraatiot. Työkalujen pääsääntöinen tarkoitus on minimoida virheitä, ja vähentää toistuvaa manuaalista työtä, jotta voin keskittyä tietoturvan hallinnan prosessiin. Näiden työkalujen kehitystyö, tai jatkokehitys on erittäin vahvassa osassa tässä opinnäytetyössä ensimmäisten seurantaviikkojen jälkeen.

Kaikki pysyvät ratkaisut, kuten tietojärjestelmät, joita toteutan tai hallinnoin dokumentoidaan yrityksen käytäntöjen mukaisesti, ja niiden lähdekoodit varastoidaan keskitettyyn versionhallintajärjestelmään.

Käytännössä työtehtävissä tarvittava osaaminen jakaantuu teknisiltä taidoilta ohjelmointitaitoihin Python, PHP ja JavaScript-ohjelmointikielillä. Järjestelmien ylläpitotaitojen osalta on tärkeää hallita ja ymmärtää miten tietoverkot toimivat, miten erilaiset internetin yleisimmät protokollat toimivat yhdessä, ja tietysti täytyy pystyä ylläpitämään Linux-palvelimia. Ideaalia on, että harrastaa alaa myös vapaa-ajalla, jotta omat taidot pysyvät ajankohtaisina työelämälle.

Tietoturvan hallinnan osalta on tärkeää tuntea alaan liittyvää sovellettavaa lainsäädäntöä, sekä yleisimpiä alan prosesseja, joita voi soveltaa alalla työskentelyyn. Joissakin tilanteissa minun on pitänyt mm. takaisinmallintaa sovelluksia ymmärtääkseni miten jokin haittaohjelma toimii tarkalleen, eli työssä tulee osata myös haittaohjelma-analytiikan perusteita joissakin tapauksissa.

Ei-teknisiä, mutta olennaisia taitoja ovat asiakaspalvelutaidot, ja lähes natiivitason englanninkielen hallinta, sillä työtehtävien suurin sidosryhmä ovat asiakkaat, joita pitää joskus motivoida tekemään vaaditut muutokset, jotta heidän palveluiden käyttönsä olisi käyttöehtojen ja sovellettavan lainsäädännön mukaista. Sujuva taso kielestä tarvitaan taas siksi, että yritys on monikansallinen, ja englanti on myös yrityksen palvelukieli.

Osaaminen työtehtäviin on tullut pääasiassa harrastamisen kautta jo ennen kuin siirryin kokopäiväiseen työelämään eli noin 5 vuotta sitten. Ylläpidin silloin erilaisia Linux-pohjaisia virtuaalipalvelimia ja siirryin tekemään alan yritykseen teknistä asiakaspalvelua verkkosivujen alustojen ylläpitoa tuottavaan yritykseen, missä opin soveltamaan harrastusta käytäntöön, ja oleelliset asiakaspalvelun taidot. Työtehtävissä osaaminen kehittyi, ja pystyin ratkaisemaan vaativampia teknisiä haasteita, ja lisäksi kiinnostus, että osaamisen tietoturva-alaan kasvoi. Silloinen työpaikka ei kuitenkaan tarjonnut riittäviä kehitysmahdollisuuksia, joten vaihdoin työpaikkaa nykyiseeni, jossa tämä oli mahdollista tehokkaassa perehdytyksessä, ja sopivassa ilmapiirissä.

Työpaikkojen ja harrastuksien lisäksi osaamista tietoturva-alaan alaan ovat lisänneet erilaiset alan konferenssit, kuten CCC eli Chaos Communications Congress, jota järjestetään Saksassa. Kyseessä on eräs maailman suurimmista hakkeritapahtumista, jossa järjestetään alan työpajoja, ja luentoja.

Tällä hetkellä katson olevani kokenut asiantuntija alalla, enkä tarvitse juuri muita työpaikalla ohjaamaan toimintaa, vaan kehitän autonomisesti omaan työtehtävääni kuuluvaa prosessia eteenpäin, kuten minua on ohjeistettu tekemään, ja kysyn välillä palautetta tehdyistä muutoksista ja esittelen prosessien tuloksia raporttimuodossa, tai tarjoan muille tarvittavan rajapinnan hankkia tietoa omiin raportteihinsa. Tämä johtuu siitä, että olen ollut

nykyisessä työtehtävässäni jo yli 2 vuotta kokopäiväisesti työntekijänä, joten suurin osa prosesseista tuntuvat minulle pelkiltä normaaleilta rutiineilta.

Näin ei tietenkään ollut heti, kun aloitin noin 2 vuotta sitten nykyisissä työtehtävissäni vaan tarvitsin paljon palautetta varmistuakseni, että teen työtehtäviäni oikein, jotka vaativat paljon huolellisuutta, ja tein näin käytännössä jokaisen uuden kohtaamani ongelman kohdalla, kunnes olin tarpeeksi itsevarma tehdäkseni töitä ilman erityistä ohjausta.

Yleisellä tasolla ja käytännössä minua on ohjeistettu ottamaan kaikkia mahdollisia loppuasiakkaiden tietoturvapoikkeamiin liittyviä prosesseja haltuun, jotta prosesseilla olisi varmasti ns. omistaja eli henkilö, jonka vastuulla niiden tekemiseen ja suorittamiseen liittyvät asiat näkyvästi ovat yrityksen omassa vastuunjaossa.

Minulla on työpaikalla myös avustamassa henkilö, joka pystyy hoitamaan työtehtäviäni, jos olen väliaikaisesti tekemässä muita tehtäviä, ja olen myös perehdyttänyt hänet alan tehtäviin. Koska olen ollut perehdyttämässä henkilöä työpaikalla katson, että minulla on tarvittavia valmiuksia hoitaa myös omaan alaani liittyviä esimiestaitojen perusteita, ja todennäköisesti oma esimieheni katsoo myös näin, koska perehdytys kyseiselle henkilölle järjestettiin näin kauttani SOC-taitojen osalta.

Minulla on myös aiempia korkeakouluopintoja henkilöstöhallinnosta, ja johtamisesta, joten niistä on varmasti ollut myös apua ns. hiljaisen tiedon muuttamisessa kirjalliseen, ja suulliseen muotoon koulutustilanteessa.

Katsoisin, että minun on mahdollista ryhtyä keskittymään joko enemmän esimiestehtäviin, tai sitten, että minun olisi mahdollista syventää osaamistani entisestään siirtymällä toisenlaisiin työtehtäviin, jos minua kiinnostaisi yhtäkkiä enemmän vaikkapa haittaohjelmien päivittäinen analysointi, tai muuten vaativammat työtehtävät. Toistaiseksi suunnittelen pysyväni nykyisessä yrityksessä töissä, jonka asiakasmäärä näyttää tasaisesti kasvavan eli yrityksen kasvu näyttää olevan vakaalla pohjalla, ja se on myös vakavarainen, eli työntekijälle yrityksessä on erittäin turvallista pysyä.

Asiakasmäärien kasvu indikoi sitä, että asiakkaiden aiheuttamien tietoturvapoikkeamien määrä tulee varmasti entisestään kasvamaan, koska on ennakoivasti mahdotonta estää kaikkien ei-toivottujen asiakkuuksien syntymistä pitkällä aikatähtäimellä, joka tulee varmasti aiheuttamaan sen, että kaksi kokopäiväistä henkilöä, ja muina vuorokaudenaikoina toimivat päivystäjät eri mantereilla eivät yksinkertaisesti riitä vastaamaan kasvaviin tapausmääriin.

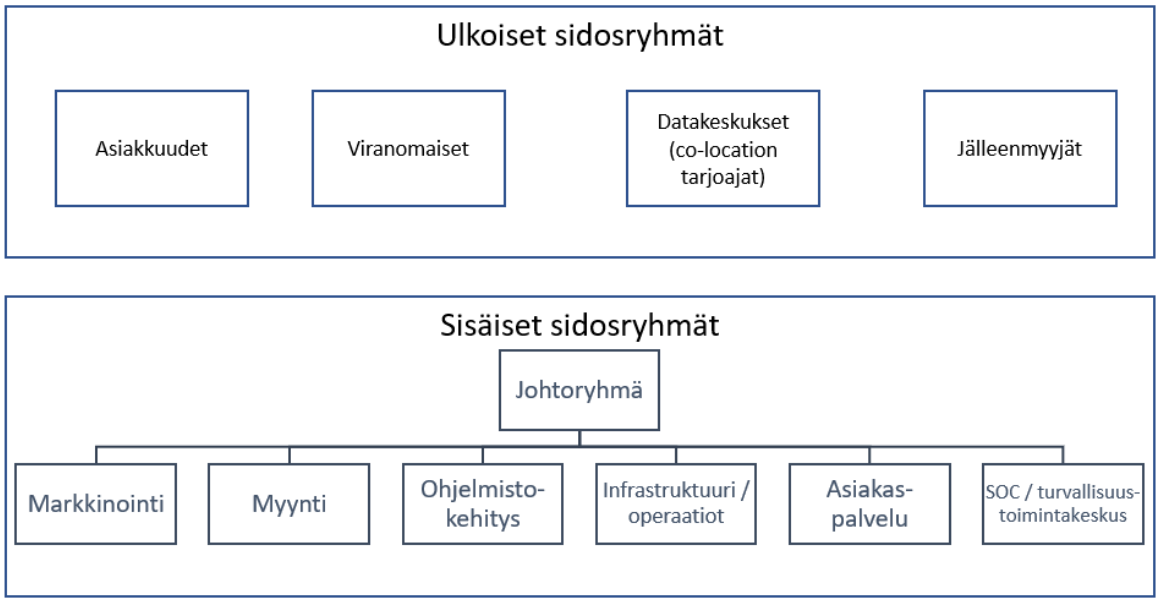
Aliresursointi johtaa useimmiten siihen, että joko käsiteltäviä asioita ratkaistaan yrityksessä liian hitaasti, tai jos halutaan pysyä jossakin sovitussa aikataulussa niin se voi johtaa siihen, että tärkeitä asioita käsitellään liian huolimattomasti, mikä ei ole toivottava tilanne. Tällä hetkellä aliresursointia ei onneksi ole, ja ilmoittaisin siitä lähiesimiehelleni, mikäli kyseinen tilanne tulisi jostain syystä vastaan.

Ennustaisin enemmänkin, että kasvu johtaa siihen, että tiimiini tulee lisää rekrytointeja perehdytettäväksi jossakin vaiheessa, ja että oma toimintani alkaa muuttua esimiestehtävien yksinkertaisista harjoituksista varsinaiseksi esimieslähtöiseksi työksi. Näkemykseni perustuu myös siihen, että ollessani aiemmin töissä eräällä teleoperaattorilla huomasin, että monen henkilön urakehitys oli vastaavanlainen työkokemuksen kasvaessa, ja rekrytointien määrän kasvaessa samalla, kun yritys kehittyy eteenpäin. Rekrytointitarpeet voivat myös muuttua, jos esimerkiksi saan tehostettua jotain henkilöriippuvaista prosessia omilla tietojärjestelmilläni niin, että rekrytoiminen ei ole tarpeellista.

Työelämässä moni asia ei tule tapahtumaan ilman erityistä ilmoitusta, tai vaatimusta, joten olisi hyödyllistä näitä asioida pohtiessa varautua pyytämään omilta esimiehiltä vaativampia, ja esimieslähtöisempiä työtehtäviä, jotka antavat mahdollisuuden soveltaa jo aiemmin opittua johtamisen teoriaa käytäntöön, eli minulta puuttuu pääosin osaamista esimiestyön harjoittamisesta käytännössä, mikäli yrityksen kehitys jatkuu nykyisellä tahdilla. Tarvetta esimiesosaamiselle on toisaalta vain, jos haluan siirtyä senkaltaisiin työtehtäviin.

2.2 Sidosryhmät työpaikalla

Sidosryhmät työpaikalla voi luokitella seuraavasti:



Kuva 1. Kaavio sidosryhmistä työpaikalla

Kaavion mukaisesti yrityksessä on tyypillisen IT-organisaation mukainen rakenne, jossa jokaisella tiimillä on oma esimiehensä. Sisäisistä sidosryhmistä tärkeimpiä ovat johtoryhmältä tulevat mielipiteet, jotka päättävät lähtökohtaisesti palvelun hyväksyttävän käytön, josta tulee ohjenuora käyttöehtojen muodossa loppuasiakkaille, jonka noudattamisen valvominen on myös omaa työtäni.

Tämä johtaa siihen, että oma toimintani koskettaa erityisesti myyntiä, ja markkinointia, joilla on tavoitteena saada yrityksen kokoa kasvatettua, kun taas itse pyrin valvomaan, että asiakkaiksi ei päätyisi ei-toivottuja asiakkuuksia. Asioista keskustellaan näiden sidosryhmien kanssa. Olen myös vuorovaikutuksessa asiakaspalvelu-tiimin kanssa, jonka kanssa olemme sopineet työnjaosta niin, että esimerkiksi kaikki asiakkaita huolestuttavat tietoturvakysymykset tai vaikkapa kyselyt luvallisesta tunkeutumistestauksesta tulevat käsitellyiksi kauttani.

Ulkoisista tärkein sidosryhmä on minusta asiakkaat, ja toiseksi viranomaiset. Asiakkaat ovat niitä, joihin yrityksen liiketoiminta perustuu, ja tietysti laukaiseva tekijä päivittäisille työtehtävilleni.

Viranomaiset taas ovat taho, joiden kanssa voin joutua tekemisiin, mikäli jollakin sopivalla viranomaisella on annettavana minulle määräys, jonka puitteissa toimin. Viranomaiset ovat myös siinä mielessä tärkeitä, että ne edustavat laissa säädettyä minimitasoa vastuullisuudesta, jonka lomassa pitää vähintään toimia. Olenkin ensimmäisen linjan kontakti usein viranomaisille.

2.3 Vuorovaikutus työpaikalla

Tärkeimmät vuorovaikutustilanteet työpaikalla työntekijöiden kanssa ovat työni raportointi suullisessa ja, tai kirjallisessa muodossa esimiehelleni, ja uusien työntekijöiden perehdyttämiseen osallistuminen muiden perehdyttäjien kanssa, jotta jokaiselle aloittavalle työntekijälle tulee selväksi, että mitä työtä teen. Vastaavia esittelytapahtumia järjestävät muidenkin tiimien työntekijät aloittaville rekrytoinneille. Asiakkaiden kanssa vuorovaikutustilanteet ovat pääasiassa minun, ja loppuasiakkaan välistä keskustelua, joka koskee jonkin asiakkaan tahallaan tai tahattomasti aiheuttaman tietoturvatapahtuman käsittelyä.

Asiakkaiden kanssa vuorovaikutustilanteet ovat käytännössä normaalitilanteessa kahdenvälistä sähköpostivaihtoa, jossa asiakasta kehotetaan tekemään jotakin, mikä palauttaa heidän toimintansa sellaiseksi, joka ei riko sovellettavia lakeja, tai käyttöehtojamme.

Haasteita suorastaan omalle osaamiselleni ei synny tästä tilanteesta lainkaan, mutta joissakin tilanteissa vastaan saattaa tulla sellaisia asiakkaita, jotka eivät esimerkiksi ole tarpeeksi osaavia ymmärtämään ongelmaa, joka asiakkaan tulee ratkaista, jos he eivät ole esimerkiksi tekniseltä osaamiseltaan tarpeeksi osaavia. Vastaavanlaisia ongelmia aiheuttavat myös sellaiset asiakkaat, jotka eivät halua reagoida, tai väittävät reagoineensa kehotuksiin, joita väärinkäyttötilanteita selvittäessä lähetetään. Ongelmia ei synny, jos asiakkaalla on myös vastaavasti hyvät prosessit ratkaista heihin liittyviä tietoturvapoikkeamia ja riittävää ymmärrystä niiden hoitamisesta.

Ainoat lopputulokset, mitä tästä asiakkaan toiminnasta voi seurata on, että toiminnasta annetaan entistä vakavampia varoituksia, tai sitten asiakkaan palveluun kohdistuu joko väliaikaisia, tai pysyviä sulkutiloja, joilla asiakkaan haitallinen toiminta pakotetaan loppumaan, koska kehotuksien noudattamatta jättäminen rikkoo palvelumme käyttöehtoja. Myöskään selitykset, joissa asiakas kertoo, ettei ymmärrä rikkovansa käyttöehtoja tai lakia eivät ole hyväksyttäviä syitä rikkoa kyseisiä asioita.

3 Päiväkirjaraportointi

3.1 Seurantaviikko 1

Maanantai 03.01.2022

Asetin tälle päivälle käytännössä ainoaksi merkittäväksi tavoitteeksi rutiininomaisten työtehtävieni lisäksi yhden virtuaalikoneen tilan tarkistamisen, jotta selviäisi tarvitseeko sitä huoltaa mitenkään erityisesti. Virtuaalikone on vastuussa mm. SOC-tiimin tiketöintijärjestelmän saatavuudesta sisäiseen käyttöön.

Käyttöjärjestelmä, jota virtuaalikoneessa ajetaan, on suosittu Debian -distribuutio, joka on tällä hetkellä versiossa 10, mikä on kirjoitushetkellä kyseisen distribuution toiseksi uusin vakaa versio. Pakettienhallinnan mukaan tarkistuksessa ei löytynyt mitään päivitettävää, mikä johtuu käytännössä siitä, että palvelin on konfiguroitu asentamaan kriittisiä tietoturvapäivityksiä automaattisesti käyttämällä Debianin Unattended Upgrades -järjestelmää, joka osaa myös tiedottaa sähköpostitse tai muulla tavalla niistä päivityksistä, jotka vaativat uudelleenkäynnistystä. Jäljelle jäisi käytännössä siis sellaisia päivityksiä, joilla ei ole niin korkea prioriteetti asentua automaattisesti, jos niitä olisi tarjolla.

Levytilan käytöltä ei löytynyt myöskään mitään huomioitavaa, sillä siihen liittynyt aiempi ongelma oli jo ratkaistu. Ongelmana oli aiemmin, että kun palvelimen säännöllistä varmuuskopiointia konfigurointiin, oli palvelimessa liitettynä vain yksi tallennuslaite, joka oli liian pieni. Kun päivittäisiä paikallisia varmuuskopioita ryhdyttiin ottamaan kaikista palvelimeen liittyvistä tiedostoista, niin kumulatiivinen levytilan käyttö, joka varmuuskopioiden määrästä ja koosta syntyi, oli lähes täyttää palvelimen koko levyn

Automaattinen Zabbix-pohjainen valvonta kuitenkin huomasi, että tärkein levyosio palvelimella alkoi täyttymään. Ratkaisuna oli liittää palvelimeen silloin suuri, ja hitaampi eli täysin tiedostojen varastointiin sopiva tallennuslaite, johon kaikki varmuuskopiot yksinkertaisesti siirrettiin, ja siitä tehtiin varmuuskopioiden kohde. Lisäksi virtuaalikoneen fstab -asetustiedostoa muutettiin niin, että levyosio tulee käynnistyksen yhteydessä aina saataville.

Tarkistin varmuuden vuoksi SystemD järjestelmästä myös kaikkien käytössä olevien palveluiden tilan, ja ne näyttivät kaikki olevan toiminnassa normaalisti.

Palvelimen saatavilla olevan muistin määrästä ja suoritinajan riittävyydestä en tehnyt havaintoja, koska tilannetta valvoo jo aiemmin mainittu Zabbix-järjestelmä. Jos valvonnassa olisi häiriöitä, niin Zabbix-palvelin itsessään ilmoittaisi, ettei se ole saanut uusia tietoja.

Pidän tämän päivän tarkistusta onnistuneena, koska siinä ei löytynyt mitään erityisiä löydöksiä kohteena olleesta virtuaalikoneesta. Reflektointi aiempiin huoltotoimenpiteisiin ja niiden seurauksien tarkistaminen käsin myös varmistaa sen, että automatisoitu valvonta näyttää luotettavaa tietoa erityisesti palvelimen levyosioista. Oppimiskokemuksen osalta päivä ei ollut niin merkittävä, paitsi sen osalta, että palvelimen tila varmistaa lähinnä itselleni, että toimin oikein.

Tiistai 04.01.2022

Päätin omistaa tämän päivän sisäiselle uudelleenbrändäyksen loppuunsaattamiselle. Yrityksen työkieli on englanti, ja sanatarkasti tiimistäni on käytetty aiemmin muotoa ’abuse team’, mikä saattoi joskus olla kuvaava nimi, ja myöhemmin siitä tuli SOC (Security Operations Center) eli turvallisuustoimintakeskus, tai turvallisuusoperaatiokeskus. Uudelleenbrändäyksen tarkoituksena on muuttaa tiimin nimi vielä niistä yrityksen sisäisistä kanavista uuteen muotoon, missä vanhaa nimeä vielä käytetään.

Ensimmäisenä huomioni kiinnittyi viestintäkanavien osalta Slack-pikaviestimen kanavien nimiin, sillä niistä löytyi jäänteitä vanhan nimen käytöstä. Jotkin viestintäkanavista olivat minun luomiani, joten niiden nimien muuttaminen onnistui helposti, ja loput ratkesivat yksinkertaisesti niin, että pyysin esimiestäni muuttamaan käytössä olevat nimet toivotuiksi.

Sisäisistä ohjeartikkeleista löytyi hieman jäänteitä vanhojen kanavanimien käytöstä, mutta ratkaisin ongelman hakemalla artikkelien listasta kaikki ne sivut, joilla esiintyi viittauksia vanhoihin nimiin, ja kävin muokkaamassa artikkeleihin uudet nimet paikalleen. Ohjeartikkeleiden päivittäminen on tärkeää, koska niitä lukevat erityisesti uudet työntekijät osana heidän omaa perehdytystään, kun he tutustuvat muihin tiimeihin.

Lopuksi tein sisäisiin kanaviimme tiedotteen nimien muuttamisesta, ja siitä, että projekti on saatu nyt päätökseen. Koska tärkein viestintäkanava on Slack, niin asiasta ei olisi välttämättä tarvinnut tehdä erityistä tiedotetta, koska kanavien nimien muuttaminen toimii reaaliajassa niin, että kanavalle jo liittyneet henkilöt huomaavat olevansa uudella kanavalla, ja lisäksi Slack muistaa kanavan aiemman nimen, ja huomauttaa siitä, jos jokin henkilö yrittää liittyä vanhan kanavanimen omaavalle kanavalle. Uudet työntekijät näkisivät taas ohjeartikkeleista, että mikä kanava on oleellinen SOC:ista keskusteltaessa.

Ongelmilta ei kuitenkaan vältytty. Slackiin oli integroitu muutamia Bash-skriptejä, joita käytettiin sisäiseen tiedottamiseen kanavilla, ja huomasin niiden ylläpitäjänä, että ne ovat ohjelmoitu edelleen aiemman tekijänsä toimesta tiedottamaan viestejään kanaville, joita ei ole enää ollenkaan olemassa, koska kanavien nimi on vaihtunut. Koska näiden skriptien

luonne on sellainen, että ne ajetaan säännöllisesti ajastettuna muutamia kertoja päivässä, niin ratkaisu oli vain vaihtaa skriptien sisällä viestien kohdekanavien nimet uusiin, jolloin ne toimivat oikein. Huomasin tämän onnekseni vielä ennen skriptien seuraavaa ajokertaa.

Päivän projektin tarkoituksena oli, että erityisesti uudet työntekijät ymmärtävät tiimin tarkoituksen joko välittömästi tai ainakin saavat idean siitä, että mitä tiimi tekee yrityksessä jo näkemällä tiimin nimen, sillä IT-alalla SOC itse terminä on yleisesti käytössä kaikista tiimeistä, jotka tekevät vastaavia töitä kuin minä. Opettavana kokemuksena voisi pitää sitä, että kovakoodausta tietyissä skripteissä kannattaisi välttää, että esimerkiksi kanavanimien muutoksia ei tarvitsisi toteuttaa käsin.

Keskiviikko 05.01.2022

Tänään päätin, että perehdyn päivän osalta lähes täysin vaihtoehtoisien kirjautumistapojen selvittämiseen tiketöintijärjestelmän osalta, mikä on Freescout-pohjainen. Vaihtoehtoisia kirjautumistapoja perinteisen käyttäjätunnuksella ja salasanalla kirjautumisen lisäksi ovat mm. LDAP-pohjaiset ratkaisut, SAML ja oAuth -ratkaisut. LDAP-tapa kirjautua järjestelmään olisi järjestelmän toteuttajien osalta virallisesti tuettu, mikä olisi yrityksen jatkuvuussuunnittelun osalta hyvä asia, kun taas SAML ja oAuth -ratkaisut ovat epävirallisesti tuettuja, ja Freescoutin käyttäjien tekemiä laajennuksia.

Saatavilla olevien ratkaisujen lisäksi tulee miettiä myös ratkaisuja, jotka ovat työnantajan osalta käytettävissä. Kyseessä on sisäinen järjestelmä, joten tässä tapauksessa ei tarvitse miettiä muiden, kuin sisäisten sidosryhmien teknisiä vaatimuksia. Tämä rajaa jäljelle LDAP ja oAuth -kirjautumistavat perinteisen metodin lisäksi, mutta koska vain LDAP on Freescout -projektin tekijöiden virallisesti tukema moduuli, päätin ottaa sen tarkasteltavaksi ainoana potentiaalisena vaihtoehtona.

Hyöty moduulin mahdollisesta käyttöönotosta olisi esimerkiksi se, että yritykseen tulevat työntekijät pääsisivät jo automaattisesti LDAP-tunnuksellaan kirjautumaan järjestelmään ilman, että järjestelmän pääkäyttäjistä jonkun täytyy tehdä hänelle käyttäjä, kuten minun. Vastaavasti jos työntekijä poistuu, niin ei tarvitsisi huolehtia siitä, että käyttäjätunnus varmasti poistuu. Moduuli tukee mm. LDAP-järjestelmästä käyttäjien automaattista synkronointia taustalla, joten käyttöönoton jälkeen normaalitilanteessa moduulia tarvitsisi vain päivittää. Kaiken tämän lisäksi moduuli tukee LDAP-ryhmien perusteella toimivaa pääsynhallintaa eli moduuli antaa vain yrityksessä määrätyssä roolissa toimivalle henkilölle juuri oikeat pääsyoikeudet.

Halusin, että vain tiettyyn LDAP-ryhmään kuuluvat työntekijät pääsevät järjestelmään käsiksi, joten pyysin sisäisistä sidosryhmistä infrastruktuuritiimiä luomaan minulle uuden LDAP-ryhmän ja siihen kaksi testitiliä, joista toinen kuuluu ryhmään ja toinen ei. Tunnuksien saapuessa olisi tarkoituksenmukaista, että niillä testataan miten LDAP-toimii syvemmällä tasolla, koska minulla ei ole kovin riittäviä taitoja vielä sen käyttämiseksi, vaikka olen testimielessä joskus rakentanutkin OpenLDAP -serverin.

Tämä päivä liittyi suurempaan projektiini, jonka tarkoituksena on varmistaa, että ylläpitämissäni järjestelmissäni noudatetaan AAA-periaatteiden lisäksi tietoturvan osalta myös sellaisia käytäntöjä, että prosesseissa olisi mahdollisimman vähän välivaiheita, mikä tässä tapauksessa helpottaa erityisesti prosesseja, joissa uusia työntekijöitä liittyy tai poistuu yrityksestä. Osaamiseni kehittyi erityisesti LDAP-järjestelmien perusteiden ymmärtämisen osalta.

Torstai 06.01.2022

Tämä päivä on vapaapäivä Suomessa.

Perjantai 07.01.2022

Tavoitteeni perjantaille oli siistiä ja uudelleenorganisoida käyttämäni projektinhallintatyökalu Asanan työlistoja niin, että selkeästi omat kehityshankeet liittyen SOC-tiimissä erotellaan omaksi lohkoksi päivittäisten rutiinitoimenpiteiden sivussa. Lisäksi sain tietää, että yritykseen on tulossa uusia työntekijöitä, joille tulisi järjestää perehdytystä omista työtehtävistäni, joten tarkoitukseni oli perjantaina myös varmistaa, että käyttämäni materiaalin Google Slide -diat olisivat varmasti ajantasaisia.

Projektihallinnan siistiminen on verrattain helppoa työtä, sillä työssä tarvitsi ainoastaan joko merkitä valmiiksi jokin työ sille varatusta osiosta, poistaa se tai työn osatehtävä. Keskiviikkoiseen työpyyntööni liittyneet testitunnukset olivat jo saapuneet keskiviikkona, joten suunnittelemani LDAP:iin siirtyminen tiketöintijärjestelmässä kirjautumisien osalta voisi liikkua eteenpäin, joten päätin lisätä projektiin kommentiksi, että LDAP:ia voisi testata viikolla 2 tai 3 vuonna 2022 ilman integrointia tiketöintijärjestelmään.

Kehityshankkeille perustin oman osion, jonka nimeksi tuli SOC R&D missä R ja D tarkoittavat tutkimusta ja kehitystä, ja siirsin sinne kaikki työkalujeni parannusehdotukset ja toiveet niihin kohdistuvista muutoksista.

Perehdytysmateriaalin osalta tein muutoksia ainoastaan käytössä olleeseen kuvaani itsestäni ja lisäsin materiaalin tietoihin, että opiskelen nykyään Haaga-Heliassa. Tarkastelun perusteella materiaali olisi esityskunnossa ensi viikon tiistaille.

Tavoitteet tälle päivälle toteutuivat täysin kuten olin suunnitellutkin, ja olen myös suunnitellut osittain tässä tapauksessa ensi viikon työtehtäviä. Projektinhallinnan järjestäminen on järkevää toimintaa ja eriyttää tiimin käynnissä olevia projekteja selkeällä tavalla, joten oppimani projektinhallintataidot ovat päässeet tässä tapauksessa käyttöön.

Viikkoanalyysi

Viikon aikana käytin eniten aikaa rutiinitehtävieni tekemiseen, joista saatan ottaa joitakin toimenpiteitä näytteeksi tulevissa päiväkirjamerkinnöissä sen lisäksi, että niitä olisi kuvattu vain osiossa, jossa kerron työni kuvauksesta. Toteutin myös erilaisia ylläpitotehtäviä ja tein sidosryhmäviestintää päiväkirjamerkintöjen mukaisesti.

Huoltotehtävä oli itsessään melko tavanomainen tarkistus palvelimelle, joka on jo perustamisvaiheessa rakennettu niin, että sen huoltamiseen tarvitsee käyttää mahdollisimman vähän aikaa, mutta toisaalta huollon tekeminen päivätöiden ohessa varmistaa, että näin on.

Sidosryhmäviestinnän osalta pidän tärkeänä, että yrityksessä käytetään samoja yhteisiä termejä kaikkialla, ja että ne vakiinnutetaan sekä vanhoille työntekijöille tiedottamisen kautta, ja uudet asiat tuodaan taas ilmi perehdytysmateriaalin kautta.

Eniten osaamiseni kehittyi työn osalta LDAP-järjestelmien tuntemisen osalta, sillä The Internet Societyn (2006) dokumentin perusteella oli mahdollista saada selkeä ymmärrys siitä, että miten LDAP protokolla on tarkalleen määritetty toimimaan, sillä kyseinen dokumentti kuvaa protokollan toimintatavan ja kapseloinnin suhteessa OSI malliin.

3.2 Seurantaviikko 2

Maanantai 10.01.2022

Suunnittelin tekeväni maanantaina pelkästään tavanomaisia tehtäviäni, mutta muuten erittäin rauhallinen työvuoro keskeytyi.

Valvontatyön ohessa sain yhtäkkiä useilta eri tahoilta ilmoituksia sähköpostilla tapahtuneesta roskapostituksesta. Ilmoituksissa tuli joko mukana täydet otsaketiedot eli headerit, joilla sähköpostiviestin reittiä pystyisi jäljittämään eri palvelimien välillä tai sitten osa ilmoituksista oli puutteellisia. Oleellisimpana asiana pidän sitä, että ilmoituksia tuli useita, ja että niissä oli myös viestien sisällöistä kopioita. Viestien sisällön tarkoituksena oli ilmeisimmin houkutella lukijaa siirtymään sivustolle, jossa kerrottaisiin vastaanottajalle saatavasta lähetyksestä lisätietoja. Viesteissä viitattiin myös erääseen suureen lähettiyritykseen harhaanjohtavasti.

Viestin SPF-tiedoista pystyi päättelemään, että viesti oli lähetetty väärennetyllä osoitteella. SPF-tietojen tarkoituksena on tarjota alkeellinen DNS-perusteinen järjestelmä, jolla voidaan määritellä sallitut IP-osoitteet lähettämään tietyn verkkotunnuksen nimissä sähköpostia. Tämän lisäksi sähköpostissahan ei ole muita tapoja varmistaa lähettäjän aitoutta, kuin DKIM-avaimet, ja nämä molemmat tavat ovat epävarmoja tapoja aina varmistaa lähettäjän aitous.

Tapaus näytti selvältä, mutta tein vielä yhden varmistuksen tutkittuani kaikki ilmoitukset, että ne koskivat varmasti samaa lähettävää asiakkaan virtuaalikonetta, jonka IP-osoitteen sai tietoon lukemalla valituksissa olevia otsaketietoja ensimmäisten kauttakulkupalvelimien kohdalta otsaketietojen lopusta.

Tämän lisäksi havaitsin, että roskapostittava lähdekone oli jo tuhottu, mutta viestien otsaketietojen aikaleimojen perusteella, ja sisällytetyn aikavyöhyketiedon avulla pystyi nopeasti päättelemään, että mistä asiakkuudesta oli kyse. Asiakkuus tunnistettiin tunnukseksi, joka oli ollut pitkään käyttämättömänä, ja se päätettiin lukita, kunnes asiakas ottaisi yhteyttä aiheesta, ja tulevat väärinkäyttöyritykset estettäisiin myös näin ainakin samalta tililtä. Tapauksesta jätettiin merkintä, jonka perusteella on mahdollista viitata menneisyydessä tapahtuneeseen lukitukseen, jos asiakas ottaa yhteyttä, niin tilannetta voidaan selvittää paremmin ja nopeammin yhteisymmärryksessä.

Muusta päivän rauhallisuudesta huolimatta sain reagoitua ja rauhoitettua syntyneen tilanteen nopeasti tavanomaisen SOC-prosessin avulla, jossa saatua uhkatietoa analysoidaan, ja tilannetta jäädään lopuksi monitoroimaan mitigointitoimenpiteiden jälkeen.

Tiistai 11.01.2022

Tiistainen päivä tulisi alkamaan tekemällä mahdollisia rutiinitehtäviä, mutta aiemmalla seurantaviikolla jo suunniteltu ja mainittu perehdytyspäivä tulisi viemään loppupäivän kalenteristani, mihin olinkin jo varautunut hyvin päivittämällä itseäni koskevia dioja esityksen osalta.

Perehdytyksen tarkoituksena oli tarjota yleiskatsaus uusille eli aloittaville työntekijöille oman tiimin, että asiakastukitoimintoja tuottavan tiimin tarkoituksesta, ja suhteesta muihin yrityksessä toimiviin yksiköihin. Oman esittelyni lisäksi toin ilmi erityisesti erilaisia tietoturvauhkia, joihin tiimini reagoi, ja miten asiakkaiden kanssa toteutetaan sidosryhmäviestintää.

Sovin päivän lopuksi, että pidän 21.01.22 päivän lomaa, ja mahdollisesti myös 24.01.22 päivän. Näinä päivinä minulla on sijainen, joka hoitaa tehtäviäni ollessani poissa. Päivän kohokohta eli perehdytys omalta osalta vaikutti onnistuneen, koska keneltäkään perehdytettävältä ei jäänyt mitään kysymyksiä avoimeksi, ja lopuksi esitysmateriaali lähetettiin mukana olleille vielä kopiona kertaalleen tutustumista varten.

Keskiviikko 12.01.2022

Keskiviikon, ja loppuviikon ajattelin tekeväni lyhyitä ja pieniä selvityksiä siitä, että miten esimerkiksi jo olemassa olevia tietojärjestelmiä voisi laajentaa, tai miten avoimien lähteiden tiedustelua, eli OSINT:ia voisi hyödyntää paremmin SOC-tehtävissä.

Työn kohteeksi tälle päivälle tulikin tiketöintijärjestelmä, jota ylläpidän. Järjestelmän käyttöliittymässä on pieniä puutteita, jotka eivät haittaa työskentelyä, mutta tuottavat tällä hetkellä ylimääräisiä välivaiheita työntekoon. Yksi harvoista, mutta ylivoimaisesti toistuvimmista tehtävistä on asiakkuuden selvittäminen käsin. Tämä tapahtuu käytännössä niin, että avatessa jonkin tiketöintijärjestelmän viestiketjuista; kopioidaan selainikkunan oikealla puolella olevasta tietolaatikosta sähköpostiosoite, jonka avulla voidaan selvittää asiakkuus.

Asiaa tutkiessa selvisi, että en ollut selkeästi ainoa henkilö, joka oli pohtinut asiaa ja löysin järjestelmään sopivan lisämoduulin, johon tulee kuitenkin kehittää oma sovellus, joka voi

auttaa halutussa tehtävässä. Moduuli toimisi niin, että se injektoi JavaScript-komentosarjan avulla sivupalkkiin HTML-muodossa viestiketjun perustietojen alle haluttuja tietoja asiakkuudesta. Injektointi aloitetaan niin, että viestiketjua avatessa JavaScript-lisäosa tekee HTTP POST pyynnön halutulle palvelimelle, jonka sisältönä on viestiketjun lähettäjän sähköpostiosoite.

Oman sovelluksen kehitys koskisi nimenomaan siis sitä, että mitä sovellus palauttaa sivupalkkiin HTTP POST pyynnön vastauksessa. HTML-muotoisessa palautuvassa dokumentissa pitäisi olla tässä tapauksessa haluttuja tietoja asiakkuudesta.

Pidän löydettyä moduulia erittäin tärkeänä ajansäästäjänä, ja se tulee käyttöönottovaiheessa vähentämään myös inhimillisiä virheitä, jotka voivat sattua tietoja haettaessa. Tein lopuksi käyttöönotosta kehityshankkeen Asana-projektinhallintaan kolmannelle seurantaviikolle. Oma sovellus voisi olla esimerkiksi pieni, ja erillinen mikropalvelu tai se voidaan lisätä nykyiseen itse tuottamaani rajapintaan, joka on toteutettu Pythonin tunnetulla Flask-ohjelmointikehikolla. Suunnittelin myös torstaille tutkivani mahdollisuuksia validoida palvelun loppukäyttäjien osoitteita.

Torstai 13.01.2022

Keskiviikon suunnitelmien mukaisesti päätin, että käytän tämän päivän tutkiessa mahdollisia tapoja toteuttaa järjestelmää, jolla on mahdollista tarkistaa, onko jotain osoitetta tietyssä maassa olemassa.

Tutkin tähän potentiaalisia datalähteitä ja toteutuskeinoja, joista yksi olisi mm. Google Maps -karttarajapinnan käyttäminen, missä on erittäin kattava lista erilaisia paikkatietoja, kuten osoitteita sisältäen myös talojen numerot, ja postinumeroalueet ympäri maapalloa, mutta kiinnostuin enemmän avoimesta OpenStreetMap -projektista.

OpenStreetMap on kuin yleisesti tunnettu Google Maps -karttapalvelu, mutta se perustuu avoimeen dataan ja valtiollisiin tietolähteisiin, jotka ovat avoimia, sekä tuhansiin vapaaehtoisiin kehittäjiin ja alustavien päätelmien mukaan siinä näyttää olevan tarpeeksi salliva lisenssi CC-BY-SA 2.0 yrityskäyttöön (OpenStreetMap s.a.).

OpenStreetMapin kartta-aineisto on ladattavissa, ja projektin omilla kehitystyökaluilla siitä on mahdollista suodattaa dataa niin, että jäljelle jää vain haluttua aineistoa toivotussa muodossa. Tässä tapauksessa halusin vain listan uniikeista osoitteista tietyssä valtiossa.

Projektin dokumentaation perusteella syntyi toimiva prototyyppi datasta, jossa minulla oli hallussa kaikki Suomen kadunnimet tekstimuotoisena listana.

Hyöty prototyypistä olisi se, että dataa voisi käyttää mahdollisesti asiakastietojen validointiin tarvittaessa. Tarkoitus on kehittää prototyyppi myöhemmin toimivaksi kokonaisuudeksi. Suunnittelin tekeväni lisää tutkimusluonteista työtä perjantaille, jonka aiheena on sähköpostien validointimahdollisuuksien selvittäminen pelkillä SMTP komennoilla. Tämän päivän osalta tavoitteet täyttyivät. Lisäselvitettävää projektista olisi datan käyttökelpoisuus muihin valtioihin, kuin Suomeen liittyen ja sen validointi. Oppimiskokemuksena tästä jäi mieleen erityisesti vieraaseen dataformaattiin tutustuminen, ja suurien datamäärien käsittely onnistuneena kokemuksena.

Perjantai 14.01.2022

Torstain tavoitteiden mukaisesti tämä päivä tulisi kulumaan toisen kehitysprojektin kanssa, minkä ideana olisi selvittää vaihtoehtoisia tapoja jonkin asiakkaan sähköpostin aitouden varmistamiselle.

Kirjallisena materiaalina luin RFC 5321 dokumenttia, joka kuvaa hyvin SMTP protokollan käyttämistä, mitä sähköposteja siirtävät palvelinsovellukset käyttävät kommunikaatioissa. Oleellisena löydöksenä dokumentissa oli kolmivaiheinen SMTP sähköpostitransaktio, jossa asiakkaan ja palvelimen tervehdyssanoman eli HELO-viestin jälkeen kerrotaan sähköpostipalvelimelle MAIL FROM -komennolla, että keneltä sähköpostiviestiä halutaan toimittaa, mutta vielä tärkeämpänä keinona on vastaanottajan kohdistus RCPT TO -komennolla, jota itsessään voidaan käyttää selvittämään onko jokin sähköpostiosoite ylipäätään olemassa. Mikäli on, niin palvelin vahvistaa asian, ja jos ei niin palvelin antaa aiheesta virheilmoituksen. (Klensin 2008.)

Edellisen dokumentaation perusteella näyttää siltä, että olisi mahdollista tehdä sovellus, joka käyttää joko matalan tason ohjelmointikirjastoa kommunikaatiossa palvelinten kanssa TCP/IP -pinossa tai esimerkiksi käyttää valmista SMTP-kirjastoa jostakin ohjelmointikielestä, ja varmistaa onko jokin tietty sähköpostilaatikko olemassa tietyllä sähköpostipalvelimella.

Tätä ennen olisi tietenkin syytä varmistaa, että onko tutkittavan sähköpostiosoitteen verkkotunnuksessa ylipäätään MX-tietuetta, mikä viittaa usein vahvasti siihen, että verkkotunnuksella on mahdollista vastaanottaa sähköpostia. Jos näin ei ole, niin verkkotunnuksella ei tietenkään voi vastaanottaa sähköpostia ja sähköpostiosoite on siten virheellinen toiminnaltaan, vaikkakaan ei muodoltaan.

Tämänkaltainen ratkaisu voisi tuotantoversiona esimerkiksi varmistaa juuri asiakkaan sähköpostin aitouden niin, että erillistä sähköpostiviestiä rekisteröityneelle asiakkaalle ei tarvitsisi lähettää, vaan olisi mahdollista varmistaa kuin näkymättömästi tietojen oikeellisuus.

Päivän osalta yllätyin eniten siitä, että käyttämäni dokumentaatio SMTP protokollasta vastasi niin suoraan omaan kysymykseeni sähköpostiosoitteen varmennuksesta. Olin joskus tehnyt sähköpostipalvelimien vianselvitystä telnet-sovelluksella vastaavalla tavalla, mutta tämä varmisti sen, että samoja SMTP komentoja voi käyttää aivan hyvin myös osoitteen olemassaolon varmistamiseen. Tutkimustyön perusteella ratkaisu olisi erittäin helppo saattaa tuotantovalmiiksi, jos niin haluaisi tehdä.

Viikkoanalyysi

Viikon aikana tapahtui selkeästi pääosin kolmentyyppisiä asioita. Omistin joidenkin päivien ajankäyttöä enemmän pienkehityshankkeille, kuten loppuviikosta kävi. Osa päivistä oli pääsääntöisesti rutiinien suorittamista, ja osa ajasta kului taas henkilöstöhallinnon prosessien mukana toiminnasta.

Kehittymistä tapahtui eniten tiketöintijärjestelmän käyttöönoton yhteydessä, jonka laajennustyöt jatkuvat edelleen vakaasti. Onnistuin suunnittelemaan erittäin selkeän pienprojektin seuraavalle seurantaviikolle, josta tulee olemaan hyötyä työn stressinhallinnan osalta ja muun psykologisen kuormituksen vähentämisessä toistuvien tehtävien eliminoinnissa. Perehdytysten esittelyosuuden osalta sain, kuten muutkin mukana olleet positiivista palautetta, eli voisin päätellä tästä esiintymiskokemukseni, ja kielitaitojeni myös kehittyneen sillä jouduin esittämään perehdytyksen esittelyn osaltani englanniksi, mihin en ollut oikeastaan varautunut.

Perehdytyksistä voisi ehkä vielä erikseen kerätä palautetta, vaikka sitä tulikin tällä kertaa pyytämättä, ja vaikuttaa siltä, että perehdytyksistä palautteen kerääminen on sellainen asia, joka unohtuu erittäin monelta yritykseltä. (HR Partner, s.a.)

Etätöissä aloittavien henkilöiden osalta voisi ajatella kuitenkin, että perehdytys oli ennalta suunniteltu hyvin, päivitetty ajan tasalle ja aikataulutettu hyvin, mikä on erittäin tärkeää etätyöntekijöille. Tiimien roolit ja tarkoitukset esiteltiin myös muiden hyvien käytänteiden mukaisesti reflektoiden HR Partnerin (s.a.) muihin hyviin perehdytyskäytänteisiin.

Rutiinitehtävät ovat taas juuri rutinoitumisen takia oppimiskokemuksiltaan vähäisempiä, mutta pyrin ottamaan niitä lisää käsittelyyn tilaisuuden tullen eri seurantaviikoilla. Tällä viikolla aloitin esittelemällä sähköpostilla tapahtuvan roskapostituksen pysäyttämistä.

Roskapostituksen torjunnan osalta on tärkeää, että myös valittaja osaa valittaa oikein. Tässä tapauksessa puutteellisia tietoja ei ollut kuin muutamissa valituksissa, ja oikein tehtyjen raporttien määrän ollessa suuri, oli helppoa rajata epäiltävä taho yhteen asiakkuuteen täsmällisesti. Lifewiressä on kirjoitettu (Tschabitscher, 2021) toimesta aiheesta artikkeli, jota voisi mahdollisesti käyttää pohjana niissä tapauksissa, joissa valittajaa pitäisi pyytää toimittamaan oikeita tietoja hyvän valituksen tekemiseksi. Artikkeli näyttää keskittyvän samoihin oleellisiin otsaketietoihin mitä itse käytän roskapostituksen tutkimisessa.

Operaattorin osalta, jonka verkoissa roskapostitus tapahtuu, on tärkeää puolestaan taas osata ymmärtää viestin sisällön konteksti suhteessa otsaketietoihin (Pair 2020).

Selvitettäviä asioita olivat erityisesti sellaiset asiat, joissa oli jokin erittäin helposti ymmärrettävä tutkimuskysymys tai tutkimusongelma, kuten se, että onko mahdollista löytää niin avointa datalähdettä, jossa olisi mahdollisesti kartoitettuna kaikkien maailman maiden osoitteet, tai piti varmistaa miten SMTP protokolla tarkalleen toimii tietyissä tilanteissa.

Ongelmat, jotka tulivat esiin tutkimuskysymyksiä ratkaistaessa, olivat luonteeltaan sellaisia, että niiden ratkaiseminen tapahtui käyttämällä tiedonhankintakeinoja ja lähdekritiikkiä. Ensi viikon seurantaviikon osalta ongelmaksi tulee selkeästi se, että minun pitää auditoida itse kolmannen osapuolen lisäosan tietoturvallisuus ja soveltuvuus yrityskäyttöön, ja tehdä sille hallittu käyttöönotto tuotannossa.

Pidän tätä toista seurantaviikkoa huomattavasti merkittävämpänä kuin ensimmäistä, sillä onnistuin tekemään raportoinnista jo rutinoituneempaa, kuin aiempana seurantaviikkona. Koen suurimpana hyötynä viikoittaisista raporteista sen, että voin katsoa paremmin menneeseen, ja työpäivän aikana syntyneet merkittävämmät ideat, joita mahdollisesti haluan edistää löytyvät varmasti muualtakin kirjallisessa muodossa, jos ne eivät ole jo projektinhallinnassa.

3.3 Seurantaviikko 3

Maanantai 17.01.2022

Päivästä oli aiemman seurantaviikon mukaisesti tarkoitus tulla sovelluskehityspäivä. Tarkoitus oli nimenomaan selvittää Freescout-tiketöintijärjestelmään asennettavan lisäosan yhteensopivuus perinpohjin, ja ottaa se käyttöön hallitusti.

Koska kyseessä oli 3. osapuolen tekemä lisäosa, päätin että teen sille yksinkertaisen tietoturvan auditoinnin ennen asennusvaihetta. Lisensoinnin osalta näytti siltä, että lisäosan käyttämisessä ei ole minkäänlaista ongelmaa, sillä MIT-lisenssi sallii myös kaupallisen käytön, vaikkakaan ei anna minkäänlaista takuuta ohjelman toimivuudesta.

Auditointi olisi tarkoitus toteuttaa niin, että käyn läpi koko lisäosan lähdekoodin, ja sitä ei ollut paljoa luettavaksi eli vain noin 80 riviä PHP-kieltä. Tarkemmin sanottuna se oli kirjoitettu yhteensopivaksi PHP:llä tehdyn Laravel sovelluskehitys-paketin kanssa, jonka avulla myös lisäosan kohteena oleva Freescout tiketöintijärjestelmä on kirjoitettu. Lyhyydestä huolimatta hyöty on merkittävä, kun ottaa huomioon, että se tulisi poistamaan jatkuvasti toistuvan välivaiheen erilaisten työpyyntöjen käsittelyssä liittyen asiakkuuksien tunnistuksiin.

Selvisi, että ohjelma koostuu kahdesta komponentista pääosin. Osasta, joka rekisteröi lisäosan palvelut Freescout-järjestelmään, ja osasta, jonka perusteella muodostetaan jokaiseen viestiketjuun oma Javascript-komentosarja, joka lähettää HTTP POST pyynnön omalle sovellukselle. Toinen osa taas on pohja Javascript-komentosarjalle, joka muokkautuu siis aina ensimmäisen osan sille lähettämien muuttujien perusteella.

Oleellinen ongelma tietoturvan kannalta tässä onkin se, että lisäosa perustuu sellaiseen ajatukseen, että yrityksen sisäverkkoon voitaisiin avata sellainen palvelu, joka paljastaa ilman kirjautumista tiettyjä käyttäjätietoja. Vaikka tämä tapahtuisikin sisäverkossa, niin olisi äärimmäisen hankalaa noudattaa AAA-periaatteita lisäosan kannalta tässä tapauksessa.

Käyttöönoton sijasta päätinkin tehdä oman Laravel-yhteensopivan lisäosan ja hyödyntää siinä lisäosakandidaatin lähdekoodia. Lisäosa toimisi niin, että se noutaakin tekemästäni sovelluksesta PHP-kielen cURL -kirjaston avulla halutut käyttäjätiedot HTML-dokumenttina, joka renderöidään jokaisen viestiketjun sivupalkkiin. Käyttäjätietojen nouto tapahtuu rajapintatunnuksen avulla suojatusti siis taustalla sen sijaan, että rajapintakutsu menisi avoimesti selainliikenteen ohessa lävitse, ja Freescout-järjestelmä saa huolehtia lisäosaa käyttävien käyttäjien tunnistuksesta itsessään.

Minulla ei ollut maanantaina kokemusta Laravel-lisäosakehityksestä, mutta pääsin nopeasti alkuun lukemalla Freescoutin omaa dokumentaatiota lisäosien kehittämisestä, ja vertaamalla aiemman lisäosakandidaatin lähdekoodia muihin lisäosiin. Freescoutissa oli myös mukana ominaisuus, joka generoi automaattisesti halutun nimisen pohjan omalle lisäosalle.

Kehitys alkoi tosin HTML-dokumentin palauttavasta ohjelmakokonaisuudesta, joka ottaa vastaan HTTP POST-pyyntöjä. Minulla oli jo valmiiksi oma Flask-pohjainen rajapintapalvelin ja sovellus, johon lisäsin URL reitin päätepisteen, johon POST-pyyntö kohdistuu, ja siihen liittyvän Python-luokan, joka käsittelee POST-pyynnön.

PHP-lähdekoodin luonnostelin taas käyttämällä Insomniaksi nimettyä web-kehitystyökalua, jolla on mahdollista kehittää omia rajapintoja, eli testasin ohjelmalla, toimiiko tekemäni uusi luokka tuotantoympäristössä, ja kun se osuus oli testattu toimivaksi, niin käytin ohjelman työkaluja generoidakseni itselleni valmiin PHP cURL-yhteensopivan ohjelmaosuuden, jonka liitin mukaan tekemääni lisäosaan. Insomnia osaa vastaavat generointitoiminnot muillekin ohjelmointikielille, mikä nopeuttaa kehitystyötä.

Tekemäni lisäosa sai yhteyden muodostettua hallitusti rajapintaani, ja käytin alkuperäisen lisäosakandidaatin lähdekoodia hyödyksi sen PHP-osuuden viimeistelyssä. Aiemmin suunnitellen ja sen sijasta, että renderöisin JavaScript-koodia muuttujien avulla alkuperäisen idean mukaan; päätin lähettää jo valmiiksi käsitellyn HTML-dokumentin näytettäväksi suoraan jokaisen viestiketjun sivupalkkiin vastauksena POST pyyntöön käyttämällä palvelimen puolella tapahtuvaa renderöintiä.

Integraatio paljastui tuotantokelpoiseksi ja sen käytössä ei ilmennyt virheitä ja julkaisin kehitystyön tulokset kollegoideni kansa, ja esittelin päivän päätteeksi sen toiminnan tarkoituksen, sekä merkitsin tekemäni projektin valmiiksi projektinhallinnassa. Päivä oli varsin opettavainen Laravel-kehitysympäristön perusteiden osalta, jonka ymmärrys on selkeästi pakollista, vaikka osaankin PHP:tä.

Tiistai 18.01.2022

Tiistaina tarkoituksenani oli selvittää miten SOC-tiimin uusia ja tulevia työkaluja, että Freescout-tiketöintijärjestelmä saataisiin integroitua LDAP-järjestelmään kiinni. Ensimmäisen seurantaviikon mukaisesti selvisi, että jälkimmäinen projekti olisi täysin mahdollinen, ja sain hankittua tarkoitukseen sopivat käyttäjätunnukset testaukseen. Nyt olisi tarkoitus selvittää kuinka tunnuksia voi käyttää ylipäätään.

LDAP protokollassa käytetään sellaisia termejä, jotka eivät välttämättä avaudu suoraan sellaisenaan protokollaa käyttöönottavalle tai siitä kiinnostuneelle esimerkiksi, sillä jopa täysin tavallisista sisäänkirjautumisoperaatioista puhutaan ’bind operaatioina’. Varmin tapa saada hyvä yleiskäsitys protokollasta on lukea siihen liittyvää dokumentaatiota.

Koska tarkoituksenani oli myös tutkia, että millä tavalla tätä protokollaa voisi oikein hyödyntää sitten muissakin työkaluissa, kuin mainitsemassani Freescoutissa, ja koska oma pääohjelmointikieleni on Python, niin päätin että selvitän ensimmäisenä, että miten LDAP ja Python tulisivat toimeen keskenään.

Tutkiessani asiaa huomasin, että ylivoimaisesti suosituin ja tuetuin ohjelmointikirjasto Pythonille LDAP-protokollan kanssa toimimiseen olisi nimeltään ldap3, joka oman dokumentaationsa mukaan sitoutuu noudattamaan tiukasti RFC4510 dokumenttia, jossa kuvataan LDAP-protokollan tarkka toiminta. (Cannata, 2020.)

Lisäosan dokumentaatiossa (Cannata, 2020) oli erittäin paljon esimerkkejä ja ohjeita yhteyksien muodostamiseksi, ja se näytti olevan niin laaja, että sen kehittäjä näytti haluavan ilmeisesti myös lukijan ymmärtävän mitä LDAP-protokolla tekee, ja miten se tekee protokollassa mainitut asiat eikä keskittynyt pelkästään kuvaamaan ohjelmointikirjaston kaikkia kykyjä. Tästä oli tietysti erittäin paljon apua myöhemmässä sovelluskehityksestä. Otin siis tavoitteekseni saada kirjaston toimimaan perusmuodossa. Käytin muitakin lähteitä, kuten erityisesti Red Hatin (s.a.) LDAP-dokumentaatiota toimintaperiaatteiden ymmärtämiseksi.

Ensimmäisenä tehtävänäni tosin tein niin, että en halunnut muodostaa minkäänlaisia yhteyksiä yrityksemme LDAP-verkkoon, vaan halusin tehdä kehitystyötä kirjaston parissa paikallisesti. Kehitystyötä varten hankin itselleni Docker-kontin, joka sisälsi itsestään konfiguroituvan OpenLDAP-serverin asennuksen, joka paljastaisi itsensä Docker-isäntäkoneen verkkoon paikallisesti.

Docker-kontti pystytti itsensä virheettömästi ja sen mukana tullut dokumentaatio antoi ohjeita tehdä komentorivityökaluilla kyselyitä OpenLDAP-palvelimelle kirjautumisen kautta. Oletussalasanat olivat heikkoja, mikä oli ymmärrettävää koska kyseessä oli vain kehitysversio palvelimesta. Komentorivityökalut varmistivat lähinnä, että yhteys toimii, joten pystyisin aloittamaan kehitystyön Python-kielellä varmasti.

Koska ldap3-kirjaston kehittäjä (Cannata, 2020) oli niin perusteellinen, niin pystyin ottamaan hänen dokumentaatiostaan suoria esimerkkejä kehitystyöni tueksi. Ensimmäinen tavoite oli saada yhteys muodostettua kirjastolla palvelimeen, mihin tarvitaan protokollan mukaisesti base dn, palvelimen yhteysosoite, käytetty salaus, käyttäjätunnus ja salasana.

Base dn tarkoittaa käytännössä kaikille tietyllä LDAP-palvelimella sijaitseville käyttäjille yhteistä polkua, jonka eteen lisätään salasanalla tunnistautuvan käyttäjätunnuksen tiedot, kuten organisaatio ja käyttäjätunnus.

Sain yhteyden muodostettua Docker-kontin OpenLDAP-palvelimeen kyseisen kirjaston avulla, ja tehtyä yksinkertaisen ohjelman, joka validoi kirjautuvan käyttäjän salasanoja LDAP bind-operaation yrityksen jälkeen, ja tämän jälkeen tehtyä myös palvelimen hakemistossa hakukomentoja, joilla pyrin selvittämään mm. mihin ryhmiin aiemmin pyytämäni testikäyttäjä kuuluu oikein. Päivän lopuksi muokkasin tekemääni Python-funktiota niin, että se pystyi kirjautumaan yrityksemme LDAP-järjestelmiin, milloin lisäsin myös samalla tuen salatuille LDAP-yhteyksille.

Kehityspäivä oli varsin onnistunut, mutta siitä jää myös mietittävää tulevaisuuteen, että miten tekemäni perusmuotoinen funktio voidaan integroida johonkin tulevaan työkaluun mukaan. Yksi potentiaalinen käyttötapa olisi ehkä se, että järjestelmä validoi kirjautumisia taustajärjestelmänä LDAP:in avulla ja ldap3-kirjastolla, ja myöntää web-sovellusta käyttävälle SOC-työntekijälle vaikkapa JWT-tokenin osoituksena onnistuneesta kirjautumisesta, joka on voimassa tietyn aika käyttäjän istuntovarastossa selaimen muistissa. Oppimiskokemuksena LDAP-protokollan peruskäyttö, haut ja suodattaminen olivat hyödyllisiä, ja välttämättömiä tuleville projekteille. Freescoutin osalta integraatioiden hyödyt ovat jo aiempien seurantaviikkojen mukaisesti selviä, kunhan se työ tullaan saattamaan loppuun asti.

Keskiviikko 19.01.2022

Keskiviikkona tarkoituksenani oli keskittyä vain normaaleihin työtehtäviin, sillä olin tehnyt suhteellisen paljon töitä verrattuna omiin tavoitteisiini jo tällä viikolla.

Päivän mittaan kuitenkin kohtasin muutamia kysymyksiä koskien uusien asiakkaiden hankintaa, ja annoin selvityksiä siitä, että kuinka tarkalleen asiakkaiden rekisteröitymisprosessi toimii infrastruktuurissamme, sillä asiaa hoitavaan tiimiin oli tullut henkilöstövaihdoksia, jonka vuoksi asiaan piti antaa selvyys.

Käytin saatavilla ollutta aikaa aiemman päivän loppuosan aiheena olleiden JWT-tokenien myöntämisen perusteiden hallitsemiseen, ja opettelin generoimaan näitä tokeneita salaisella avainarvolla ja antamaan niille erilaisia voimassaoloaikoja, ja lopuksi yritin validoida tokeneitani onnistuneesti.

Keskiviikosta tuli odotetusti rauhallinen päivä, sillä se on sitä tilastollisesti yleensä käytännössä aina saapuvien työpyyntöjen määrällä mitattuna.

Torstai 20.01.2022

Tarkoituksenani oli enimmäkseen, että koska kyseessä on päivä ennen vapaapäivän viettämistä, niin pyrin tekemään työtilanteen sellaiseksi sijaisenani toimivalle kollegalleni niin, että hän pystyy mahdollisimman vaivattomasti siirtymään tehtäviini väliaikaisesti.

Käytännössä tämä tarkoitti sitä, että raportoin kollegalleni päivän lomassa huomionarvioisia asioita siitä, että täytyykö työssä tällä hetkellä kiinnittää erityisesti joihinkin asioihin huomioita. Tämänkaltaisia tapauksia ovat erityisesti sellaiset, joita ei ole kokonaan ratkaistu eli ne odottavat esimerkiksi asiakkaan vastausta. Erityisesti vielä varmistin, että tekemäni integraatio tiketöintijärjestelmään on tullut tutuksi, ja että sen tarkoitus on selvä.

Päivän lomassa muita rutiinitöitä tehdessäni törmäsin mielenkiintoiseen tapaukseen, jossa asiakas oli ajanut virtuaalikoneellaan verkkosivustoa, jota epäiltiin tekijänoikeusloukkauksista ulkopuolisilta tulleissa ilmoituksissa, ja tapaus paljastui selväksi, sillä se oli vietnaminkielinen verkkosivuportaali, jossa oli listattuna useita hyvin tunnettuja elokuvia, joita pystyi katsomaan mainoksien kera ilmaiseksi suoraan sivustosta. Tapaus rikkoo selkeästi ja räikeästi käyttöehtoja, joten asiakkaalle lähetettiin ystävällinen pyyntö poistaa sisältö saatavilta mahdollisimman nopeasti, ja asiasta tehtiin muistuttava merkintä, että asiaan tullaan lähipäivinä palaamaan eli kollegani palaa tutkimaan ollessaan sijaisena, että onko sivusto edelleen pystyssä.

Löydöstä oli myös helppo varmistaa, että se kuuluu meidän verkkoihimme tekemällä DNS-kyselyn ilmoituksen kohteena olleeseen verkkotunnukseen, jonka A-tietue ilmoittaa suoraan palvelimen IP-osoitteen, joka oli asiakkaan käytössä, minkä avulla asiakkaan tunnistus onnistui hyvin.

Päivän osalta vaikuttaa siltä, että työtehtävien väliaikainen siirto eli lomasijaisuusprosessi toimii hyvin, ja odotan saavani tietoja lyhyen loman aikana ilmenneistä asioista ja mahdollisista ongelmista tiistaina, kun palaan normaaleihin työtehtäviin. Kuvatun kaltaiset tekijänoikeusloukkaukset ovat helppoja ymmärtää, koska ne ovat räikeitä, mutta haastavampia ovat sellaiset, jotka koskevat vaikkapa jotain rekisteröityä tavaramerkkiä.

Perjantai 21.01.2022

Käytin mahdollisuuden vapaapäivään suunnitellusti. Ei merkintöjä.

Viikkoraportti

Aiemmasta viikosta poiketen tämän viikon hallitseva teema oli erityisesti sovelluskehitys, joka kuitenkin perustui aiempien viikkojen tutkimuksiin LDAP-protokollan perusteista suuremman verkkosovelluksen lisäosakehitykseen.

Sovelluskehitys Freescoutin osalta oli erittäin kehittävää, sillä vaikka tiesin jo valmiiksi, että kyseinen järjestelmä pohjautui Laravel-kehitysympäristöön, niin en ollut kuitenkaan tehnyt varsinaisia omia komponentteja Laravelilla, mutta käytännössä tekemässäni auditoinnissa lisäosakandidaattia vastaan huomasin, että minun on kuitenkin pakko perehtyä aiheeseen senkin lisäksi, kun huomasin, että kandidaatin tapa toimia on virheellinen suhteessa omiin tavoitteisiin.

Mielestäni se tapa, että tiedot haetaan sellaisella tavalla, jossa varmistetaan, että tiedot hakee sellainen henkilö, jolla on lupa hakea niitä on oikein ja tätä toteutustapaa noudattava ratkaisu on mielestäni toimiva siksi, että se heijastelee tietoturvan AAA-periaatteita sen sijaan, että hyvin toimivaan järjestelmään tulisi tehdä lisäosan toiminnan mahdollistamiseksi sellaisia muutoksia, jotka ovat tietoturvan osalta haitallisia.

Käyttämäni tapa toimia on lisäksi sellaista renderöintiä käytössä olevalle verkkosivulle, missä palvelinsovellukset tuottavat halutun sisällön sen sijaan, ja määräävät mitä sisältöä näytetään ja kenelle, mikä itsessään on tehokkaampi tietoturvatoimenpide sovellusasiakkaan puolella tapahtuvan renderöinnin sijasta. (McKenzien, 2020.)

McKenzien (2020) tekstien valossa näyttää siltä, että toimintatapani tämänkaltaisen sovelluksen parissa on oikea tai ainakin tietoturvallisempi.

Kehitystyö myös paljasti sen, että erittäin osaava Laravel-kehittäjä osaisi mahdollisesti tuottaa huomattavasti laajempia moduuleja SOC-tiimin käyttöön, jos tarve vaatisi sillä käytännössä moduulien tekeminen Laravel-pohjaisissa sovelluksissa selkeästi jo olemassa olevien PHP-luokkien toiminnan laajentamista. Tärkeää on myös, että minulla on itseni ylläpitämistä järjestelmistä entistä parempi käsitys, ja tietysti että projekti saatiin kehitysasteelta nopeasti käyttöön vajaassa viikossa.

LDAP:iin liittyvä projekti taas oli luonteeltaan sellainen, että sovelluskehitysympäristöön perehtymisen sijaan suurin kehittyminen tapahtui siinä, että ymmärtää kuinka protokolla toimii, vaikkakin aiheeseen liittyi erittäin vahvasti ldap3-nimisen Python ohjelmakirjaston toiminnan opettelu, että protokollaa pääsisi hyödyntämään myös käytännössä yksinkertaisiin kirjautumistehtäviin, jotka ovat roolipohjaisia. LDAP:in käyttö tässä mielessä on

erinomainen valinta, sillä kyseisellä protokollalla tapahtuvat sisäänkirjautumistoimenpiteet ovat sellaisia, että kirjautumisen yhteydessä voidaan tarkistaa mihin ryhmään jokin tietty käyttäjä kuuluu, ja nämä ryhmät ovat sellaisia, että ne voidaan suoraan yritystoiminnassa rinnastaa rooleihin.

LDAP on kuitenkin erittäin laaja tapa järjestää minkäänlainen hakemistopalvelu yritykseen, minkä ylläpito vaatii vastuuta, mutta tässä tapauksessa ei kuitenkaan ole tarkoituksenmukaista oppia ylläpitämään esimerkiksi OpenLDAP-serveriä, paitsi siinä mielessä tietysti, että protokollan perusteet oppii käytännönläheisesti, kuten Dockerilla pystyttämälläni harjoitusympäristöllä oli mahdollista. Dockerin käyttöä voi perustella hyvin harjoittelumielessä, sillä se on kuitenkin kehittäjien ylivoimaisesti yksi suosituimmista tavoista rakentaa kehitysympäristöjä.

Onneksi ja erityisesti ldap3 Python-kirjaston tekijä (Cannata, 2020) osoitti suurta perehtymistä protokollakuvaukseen käytännössä, sillä kirjaston olemassaolo mahdollistaa sen, ettei jonkun toisen tarvitse keksiä uudelleen tapoja käyttää LDAP-protokollaan eniten suosimallani ohjelmointikielellä. Kehittäjän sitoutuminen kuvaukseen auttaa seuraamaan parhaita periaatteita tulevien LDAP:iin liittyvien projektien osalta ja ratkaisi suurimman ongelman itseni kannalta, missä halusin selvittää kuinka kirjautuminen protokollalla tehdään turvallisesti, ja kuinka LDAP-hakuparametreilla tutkitaan halutun käyttäjän ryhmiä eli rooleja, kun yhteys onnistuneesti on muodostettu. Redhat (s.a.), joka on tunnettu yrityskäyttöön suunnatuista Linux-palveluistaan, ja käyttöjärjestelmästään oli kirjoittanut aiheeseen liittyvän hyvän tiivistelmän, joka auttoi pääsemään myös oikeaan lopputulokseen.

3.4 Seurantaviikko 4

Maanantai 31.01.2021

Lomaltapaluuni viivästyi entisestään suunnitellusta, sillä sairastuin flunssaan lähes välittömästi lomani jälkeen, ja tämä johti sitten siihen, että suunnitellulta seurantaviikolta ei tullut merkintöjä. Tämän takia sijaiseni jatkoi tehtäviäni suunniteltua pidempään, joten tämä päivä tulisi kulumaan osittainkin siinä, että saatan itseni ajan tasalle kuluneista tapahtumista.

Tarkistin siis päivän aluksi kaikki sähköpostini, joissa ei ollut mitään erityistä, sillä sähköposti itsessään ei ole juuri minulle merkittävä viestintäkanava, mutta sinne tulee kootusti ilmoituksia organisaatioiden yksiköiden jakamia viestejä omasta toiminnastaan, joten luin läpi kaikki tiedotteet, joista olin tilannut muistutuksi.

Tämän jälkeen oli vuoro tarkistaa sisäisistä viestimistä sijaiseni tekemät työt, ja näyttää siltä, että kaikki näytti sujuneen hyvin sisäisen dokumentaation ja kommentoinnin perusteella, joten siirryin tekemään töitä tavallisesti, ja ilmoitin olevani tulossa toimistolle tiistaiksi, sillä kuulin että toimistolle on saapumassa vierailijoita silloin. Kertasin myös lyhyesti viikon tapahtumia sijaiseni kanssa.

Päivän merkittävintä antia kuluneista tapahtumista oli suuren luokan huijaussivusto-kampanjan hävittäminen. Ulkopuoliselta taholta oli saapunut ilmoituksia 20 kappaletta erilaisesta huijaussivustosta, joille jokaiselle oli yhteistä se, että ne kohdistuivat ilmoittajan omaan asiakkaaseen, joka oli pankki, eli ilmoittaja oli tietoturvayritys.

Tapaus ei meinannut selvitä aluksi, sillä tapausta käsiteltiin vääränä positiivisena eli false positive -tapahtumana, sillä eristetyssä ympäristössä sivustojen tutkiminen ei tuottanut tuloksia, eli ilmoituksen sisällön sanomaa ei pystytty varmistamaan ja toteamaan todeksi ml. puutteellisten tietojen vuoksi.

Päätin ottaa yhteyttä ilmoittajaan ja jaoin kuvakaappauksia siitä, että ilmoitus ei pidä paikkaansa nykyisten tietojeni perusteella, mutta aiemmista vastaavista tapauksista kokeneena päättelin, että huijaussivustot saattoivat olla tehty niin, että huijaussisältö näytetään vain tietyn valtion alueelta vierailevalle selaimelle ja tiettyä päätelaitetta käyttävälle taholle. Tämän vuoksi pyysin ilmoittajaa näyttämään mahdollisimman pian omaa todistusaineistoa kuvakaappauksien kera pelkän tekstimuotoisen ilmoituksen sijasta, ja kertomaan että mistä valtiosta ilmoittaja selaa verkkosivua, ja mikä hänen user-agent -otsakkeensa on HTTP GET -pyynnöissä. User-agent on käytännössä WWW-selaimen lähettämä otsake HTTP -pyynnössä, jolla voidaan ilmaista pyyntöön vastaavalle palvelimelle, että millaista päätelaitetta selaaja käyttää.

Asiassa kului muutama tovi, ja valittaja ilmoitti käyttävänsä Safari-selaimen mobiilikäyttöön tarkoitettua user-agenttia, ja että hänen sijaintinsa on Venäjällä. Valittajan kuvakaappauksissa ilmeni pian, että ainakin valittaja näkee huijaussivuja omasta näkökulmastaan, ja hänen antamiensa tietojen perusteella oli mahdollista konfiguroida huijaussivustoja tarkastelevaa testiympäristöäni niin, että se esittää iPhonen Safari-selainta, joka yhdistää ilmoitettuun huijaussivustoon Venäjältä. Testiympäristöllä oli mahdollista todeta, että valitus todella piti paikkaansa.

Tilanteen havainnollistamiseksi itselleni tein taulukkolaskentasovelluksessa väliaikaisen listan jokaisesta valituksen kohteena olleesta verkkosivustosta, johon kokosin niistä tietoja. Otsikkotasolla kerätyt tiedot olivat valituksen kohteena oleva domain, aikaleima

havainnosta, IP-osoite nimipalvelimelta, ja käyttäjätunnus, jota valitus koskee. Ongelmaksi tuli nimenomaan kaksi jälkimmäistä kohtaa, jossa selvisi vielä, että vaikka muuten oli varmaa, että kyseessä oli varmoja huijaussivuja, niin asiakkuuksien tunnistaminen ei olisi mahdollista, sillä IP-osoitteet eivät osoittaneet meidän verkkoomme.

IP-osoite kuului jokaisella valituksen kohteena olleella verkkotunnuksella Cloudflaren käänteisvälityspalveluverkkoon, joka normaalissa tapauksessa on tarkoitettu estämään palvelunestohyökkäyksiä niin, että mahdollinen hyökkääjä, joka haluaa esimerkiksi kaataa jonkin verkkosivun ei saisi kohdennettua hyökkäystä oikein. Cloudflaren palvelu siis kuin piilottaa todellisen WWW-palvelimen IP-osoitteen taakseen, joten ratkaisu tässä tapauksessa oli kysyä, että onko valittaja jo pyytänyt Cloudflaren SOC-tiimiä käsittelemään tapausta, ja kävi ilmi, että valittaja oli vähän aikaa sitten tehnytkin näin, ja ilmoitin, että tämän perusteella jään odottamaan Cloudflaren vastausta. Cloudflare on erittäin tyypillinen uhkatiedon toimittaja siinä mielessä, että he tarjoavat standardimuotoisen lomakkeen, jolla tämänkaltaiset valittajat voivat ottaa yhteyttä Cloudflareen. Cloudflare taas automatisoidusti lähettää käänteisvälityspalvelun takana olevalle palvelimen IPv4 tai IPv6 osoitteen verkkoblokin tekniselle tai SOC-kontaktille lomakkeen kautta saadut valituksen tiedot.

Noin puoli tuntia tästä viestinvaihdosta pääsin täydentämään vihdoin taulukkoani haluamillani tiedoilla, kun ne saapuivat, ja selvisi, että joka ikinen mainituista sivustoista kuului samalle asiakkuudelle, joka oli erittäin tuore, ja jonka osoitetiedoissa oli ristiriitoja, kuten ristiriidassa oleva maatieto jonkin toisen maan kaupungin kanssa. Koska asiakkuus oli erittäin tuore, ja sivuja oli yli 20, ja tämän lisäksi sivustot oli yritetty piilottaa äärimmäisen tarkasti näiden ristiriitaisten asiakastietojen lisäksi, niin päätin jäädyttää asiakkuuden käyttöehtorikkomusten mukaisesti, ja kirjasin asiasta merkinnän, että asiakkuuden mahdollista avaamista voidaan käydä läpi asiakkaan kanssa, jos hän ottaa yhteyttä.

Olin omien muistikuvieni mukaisesti ensimmäistä kertaa sairauslomalla tässä yrityksessä, joten pääsin ensimmäistä kertaa näkemään, että kuinka sijaisuuksien järjestäminen oikein onnistuu tämänkaltaisessa tilanteessa, jossa jokin työntekijä sairastuu, missä oli erittäin hyvä havaita, että prosessi toimii odotetusti.

Tiistai 01.02.2022

Varauduin tälle päivälle tapaamaan suunnitellusti toimistolle saapuvaa väkeä, ja tarkoituksenani oli myös aloittaa LDAP:in käyttöönotto Freescoutissa niin, että protokolla saadaan viimeistään tällä viikolla lopullisesti käyttöön tässä järjestelmässä.

Minulle kuitenkin selvisi, että erehdyin päivästä, ja toimistolle saapuvia vierailijoita olisikin odotettavissa vasta keskiviikoksi, joten päivä kului pääasiassa aluksi LDAP-protokollan käytön aloittamisen selvittämiseen Freescoutissa.

Käyttöönottoa varten tarkistin virallisen LDAP-lisäosan dokumentaation sivuista tarvittavat tiedot käyttöönottoa varten, ja huomasin että siitä tärkein osuus koskisi oikeastaan sopivan suodattimen luomista, millä varmistetaan, että vain tietyn LDAP-ryhmän jäsenet pääsevät käsiksi järjestelmään. Jos suodatinta ei luoda, niin käykin niin, että kaikki organisaation alle luodut käyttäjät pääsevät kirjautumaan työkaluun, mikä ei ole yhtään toivottavaa.

Tästä syystä kävinkin sisäistä keskustelua esimieheni kanssa siitä, että keille kaikille tulisi antaa pääsyoikeuksia järjestelmään, ja olimme samaa mieltä siitä, että henkilöt, jotka nytkin käyttävät järjestelmää saavat käyttää sitä jatkossakin.

Tämä oli käytännössä päivän aikana vaatimusten määrittelyä suodattimelle, joka käytännössä vain tarkistaa tietystä ryhmästä, että kirjautuva käyttäjä kuuluu varmasti ryhmään. Suodatin tuli valmiiksi jo päivän aikana, ja se testattiin ldapsearch-nimisellä binäärillä toimivaksi, mutta seuraavalle päivälle minun pitää tehdä työpyyntö, jossa pyydän LDAP-järjestelmän ylläpidosta vastaavaa tahoa sisäisesti lisäämään kaikki toivomani henkilöt haluttuun ryhmään, että käyttöönottoa voidaan jatkaa.

Muut käyttöönottoon tarvittavat tiedot sain yleisestä LDAP-protokollaa koskevasta sisäisestä dokumentaatiostamme, mutta varmistin vielä, että artikkeli on ajan tasalla ja käyttökelpoinen, ja selvisi, että näin oli, joten käyttöönoton viimeisten vaiheiden pitäisi olla erittäin helppoja.

Onnistuneena asiana pidän tässä päivässä sitä, että sain suunniteltua hyvin ennakoiden tulevia töitä loppuviikolla ja asetettua selkeän tavoitteen, jonka suunnittelen saattavani loppuun. Pidän myös tiettyjen asioiden testausta hyvänä ideana, ennen kuin ne kytketään tuotantovalmiiseen ympäristöön täydellisesti.

Keskiviikko 02.02.2022

Otin päivän tehtäväksi edistää aiempaa LDAP-integraatiota lisää, ja tehdä muutoksia tiketöintijärjestelmän sähköpostien tietojen retentio-ominaisuuksiin.

Kävin keskustelua toimistolle saapuneen vierailijan kanssa, ja muidenkin Freescoutia käyttävien tahojen kanssa siitä, että miten LDAP-käyttöönotto tässä kontekstissa tulee

vaikuttamaan sisäänkirjautumisiin, eli käyttöönoton jälkeen kirjautumismuoto on oma LDAP-salasana, ja käyttäjätunnus tulisi pysymään samana kuin ennenkin.

Tärkein valmisteleva toimenpide oli tietysti kerätä lista kaikista nykyisistä käyttäjistä, ja tunnistaa heidän sähköpostiosoitteensa avulla heidän LDAP-tunnuksensa, jotka listaamalla kerroin järjestelmän ylläpidosta vastaavalle tiimille, että ketkä kaikki tulee liittää tähän LDAP-ryhmään mukaan.

Ryhmään jäsenien lisääminen ei ole mikään erityisen korkean prioriteetin työ, ja oletin, että työpyyntö valmistuu viimeistään torstaina.

Tämän lisäksi tein sähköpostijärjestelmäämme muutoksia, joilla varmistetaan, että liian vanhat sähköpostiviestit tuhotaan sähköpostipalvelimilta varmasti asianmukaisesti automaattisesti.

Näistä kahdesta muutoksesta päätin tehdä tiedotteen sisäiseen tiedotuskanavaan, ja lisäksi tein pieniä parannuksia sisäiseen tiedottamiseemme, eli tein integraation tiedotuskanavatyökalusta liitoksen Slack-pikaviestimeen niin, että joka ikinen tiedotuskanavalle tekemäni viesti tullaan lähettämään oman tiimini Slack-kanavalle niin, ettei minun tarvitse lähettää näistä viesteistä kopioita erikseen käsin.

Tämän päivän lopussa ymmärrykseni LDAP-kansioiden rakenteista kasvoi hyvin, sillä pystyin määrittelemään tarkasti järjestelmän ylläpitäjälle mitä mihinkin ryhmään piti lisätä, ja katsoin myös, että olin erittäin lähellä siinä, että saan pian tavoitteeni toimivasta integraatiosta valmiiksi tuotantoon.

Torstai 03.02.2022

Torstaina oli tarkoitus jatkaa edelleen LDAP-projektin käyttöönottoa suunnitellusti ja viimeistellä työ.

Huomasin aamulla, että Freescout-järjestelmään oli ilmestynyt uusia päivityksiä sekä järjestelmään itseensä, että järjestelmän moduuleihin, joten ensimmäisinä töinä aamulla päätin ilmoittaa käyttäjille lyhyestä huoltokatkosta, sillä tämän järjestelmän päivitykset ovat tyypillisesti sellaisia, että niiden ajaminen läpi kestää alle minuutin, ja näin tapahtui.

Tämän jälkeen oli vuoro konfiguroida LDAP-moduuli itsessään käyttökelpoiseksi. Pääpiirteissään konfigurointi oli helppoa, sillä minulla oli jo valmiiksi kaikki kirjautumistiedot, ja keskiviikkoisen työpyyntöni ansiosta olin saanut kuittauksen siihen, että kaikki toivotut

henkilöt olisivat nyt jäseniä oikeassa LDAP-ryhmässä, joten käyttöönottoa voitaisiin edistää. Ensimmäinen vaihe käyttöönotossa oli vain tarkistaa, että bind-operaatiot toimivat oikein.

Ainoastaan edistyneemmässä konfiguroinnissa tuli ongelmia, kun halusin suodattaa juuri vain tiettyyn ryhmään kuuluvat mukaan Freescoutin käyttäjiin aiemmin keksimälläni suodattimella. Seuraamalla lisäosan dokumentaatiota tämä ongelma tosin poistui, ja käyttäjiä tuodessani järjestelmään sain eteeni listan suodatinta vastaavista käyttäjistä, joiden attribuutit piti vielä määrittää esim. sähköposti, etunimi ja sukunimi tuli kiinnittää tiettyihin attribuutteihin LDAP-käyttäjissä. En tiedä vielä, että miksi aiemmin tekemäni suodatin käyttäjille ei toiminut kuin ldapsearch-binäärissä, mutta ainakin sain ongelman ratkaistua.

Käyttäjien tuonti onnistui lopulta täysin, joten oli turvallista ottaa kirjautuminen LDAP:illa käyttöön, ja aktivoida taustalla toimiva automaattinen käyttäjien tuonti, ja käyttöoikeuksien monitorointi. Kirjautumiset alkoivat toimia normaalisti LDAP:ia vasten testeissä, mutta aiemmin aamulla tehdyt järjestelmäpäivitykset olivat aiheuttaneet ongelmia 2FA-kirjautumisten aktivoinneissa, mitä päätin selvittää myöhemmin perjantaina.

Suunniteltu työ tuli siis saatettua loppuun ja merkittävä muutos kirjautumislogiikassa tuli otettua käyttöön, kuten pitikin, mutta pidän 2FA-kirjautumisten aktivoinnissa ilmennyttä ongelmaa huolestuttavana, mutta en kriittisenä koska ongelma koskee vain uusia käyttäjiä, joita ei ollut odotettavissa.

Perjantai 04.02.2022

2FA-kirjautumisten aktivointien ongelmasta seurasi suurta päänvaivaa, sillä vikatilanne tulisi kuitenkin ratkaista. Ongelmia ei ollut vanhoilla käyttäjillä olleenkaan vaan ongelma koski ainoastaan uusia 2FA-kirjautumisten aktivointeja.

Lähdin selvittämään tilannetta niin, että tutkin selaimen kehitystyökaluilla sivua, jolla 2FA-kirjautuminen aktivoidaan. Huomasin, että selainkonsoliin ilmestyi ilmoitus virheellisestä Javascript-komentosarjasta sivun sisään upotetusta Javascript-koodista. Koodia tutkimalla katsoin, että sivulla yritetään ajaa funktiota, mutta niin ettei funktio saa toista tarvitsemaansa muuttujaa, jolla kerrotaan, että onko jokin käyttäjä jo aktivoinut 2FA:n käyttäjätunnukselleen, ja jos ei niin käyttäjä ohjataan sellaiselle sivulle, missä aktivoinnin voi tehdä.

Tämän perusteella selvitin, että mihin Javascript-funktio haluaisi ohjata tämänkaltaisen käyttäjän, jolla ei ole 2FA-kirjautumista aktivoituna, mikä paljastui normaaliksi PHP:n renderöimäksi dokumentiksi, jossa on mahdollista skannata 2FA:n aktivointiin tarvittava QR-

koodi johonkin toivottuun autentikointisovellukseen, ja että tämä aktivointisivu toimi muutenkin oikein, eli jossakin funktiossa oli vikaa niin, että tuo Javascript-koodi sivulla ei ollut upotettu oikein.

Lähdekoodia tarkasteltassa selvisi, että Javascript-upotus perustuu Laravel-sovelluskehitystyökalun Blade PHP-pohjiin, eli jostain syystä PHP-koodi, joka vastaa upotettavan Javascript-koodin generoinnista ei toimi oikein, ja lisäksi huomasin kyseisen tiedoston muokkausajasta, että sitä oli muokattu torstaina sellaiseen kellonaikaan, kun Freescoutiin oli ajettu päivityksiä sisään, joten pidin äärimmäisen todennäköisenä, että virhetilanne ei liity LDAP-protokollan käyttöönottoon vaan järjestelmäpäivitykseen.

Lounastauolta palatessani takaisin tarkistelemaan asiaa huomasin, että Freescoutin kehittäjältä oli julkaistu tuoreita päivityksiä sovellukseen, joissa kerrottiin useista virhekorjauksista sovelluksen logiikan toimintaan, ja nämä päivitykset asentaessani 2FA-aktivoinnit alkoivat toimivaan normaalisti eli Javascript-upote alkoi generoitumaan oikein, ja varmistin asian eli lähdekoodiin muodostui haluttua Javascriptia oikein.

Päivän aikana tuli siis syvennyttyä Laravel-pohjaisen PHP-sovelluksen logiikkaan syvemmin, ja siinä auttoi paljon se, että minulla oli aiemmalta seurantaviikolta jo kokemusta Laravel-sovelluskehitystä. Olisin ollut valmis ottamaan sovelluksen kehittäjään yhteyttä, mutta ilmeisimmin kehittäjä huomasi virheen muutenkin ja korjasi sen lyhyessä ajassa onnekseni.

Viikkoraportti

Tämän viikon merkinnät kietoutuivat pääsääntöisesti yhden pääteeman ympärille, eli tietojärjestelmään merkittävien muutoksien tekeminen LDAP-protokollan käyttöönotossa ja pienempään rooliin jäi sitten itsessään suuremman huijaussivustokampanjan käsittely, vaikkakin kyseessä oli tavallista edistyneempi kampanja.

Eniten viikon ajalta pohdin, että oliko LDAP-protokollan käyttöönottoprosessiin lähteminen järkevää ajankäytöllisesti. Tämänkaltaista kirjautumistapaa, joka otetaan käyttöön johonkin sovellukseen, kutsutaan ulkoiseksi autentikointipalveluksi, sillä sen sijasta, että käytettäisiin jotain sovelluksen itsensä hallinnoimaa kirjautumispalvelua kirjautumisien ja käyttäjien hallintaan, niin tämä prosessi ulkoisestaan sovelluksen ulkopuolelle.

Pidän tätä lähestymistapaa toimivana omasta mielestä jo siksi, että kyseessä on kuitenkin yrityksen osalta sellainen kirjautumistapa, jota käytetään jo nykyisessä muodossa moniin sisäisiin palveluihin, joten silläkin perusteella lisäosan käyttö on perusteltua. Aiemmilla

seurantaviikoilla tuli myös selväksi, että käyttöönotettava LDAP-lisäosa on myös sellainen asia, mitä Freescout-järjestelmän tekijä tukee suoraan. Tämänkaltaisten autentikointiratkaisuiden käyttö on perusteltua siksi, että se säästää käyttäjät erillisien salasanojen, ja käyttäjätunnuksien luomiselta palveluun (Microsoft, 2020). Lisäksi tämä taas parantaa käyttäjätunnuksien hallitsemista, ja keskitetty käyttäjätietokanta keskittää kontrollia yhteen tietokantaan (Oracle, s.a.).

Näiden perusteella on mahdollista havaita, että ulkoisen autentikointipalvelun käyttöönotto on järkevää ajankäyttöä. Tämä vähentää myös juuri keskitetyn käyttäjätietokannan kautta hallinnollisia tehtäviä esimerkiksi uusien työntekijöiden liittyessä yritykseen, sillä ryhmään rajatun pääsyn avulla uusi käyttäjä saa automaattisesti oikeat pääsyoikeudet.

Eräs vaihtoehto kyseiselle LDAP-ratkaisulle olisi voinut olla myös oAuth-tyyppinen ratkaisu, joka kuuluu myös ulkoisiin autentikointipalveluihin, sillä sitä tuetaan myös yrityksessä, mutta ongelmana tässä asiassa pidin juuri sitä, että kyseinen ratkaisu ei ole Freescoutin kehittäjän toimesta virallisesti tuettu lisäosa.

Kolmannen osapuolen ratkaisu saattaisi olla vaikkapa perusteltu sellaisessa tilanteessa, missä halutaan juuri parantaa sovelluksen käyttökokemusta, alentaa kehityskustannuksia, ja tehostaa sovelluskehitysprosessia muutenkin, mutta täysin sellaisen ratkaisun käyttöönotossa saattaa olla suurena ongelmana se, että tämänkaltaiset integraatiot eivät useimmiten käytä kehittäjän virallisesti tai suoraan tukemia metodeja, ja käyttöönotto saattaa aiheuttaa muutenkin sellaisen tilanteen, että mahdollisessa vikatilanteessa kolmannen osapuolen ratkaisu aiheuttaa riippuvuuden yrityksessä lisäosan tekijään. (Janus, J., Chojnacki, M., Szwajkiewicz, W. 2020.)

Koska käytössä oleva oAuth -ratkaisu on riippuvainen tässä tapauksessa LDAP-autentikaatiosta muutenkin, ja ulkoisten riippuvuuksien välttämiseksi, ja muutenkin tietojärjestelmän elinkaaren hallittavuuden kannalta pidän järkevänä tapaani toimia siten, että vältytään kolmansien osapuolien lisäosien mahdollisilta, ja tyypillisimmiltä todennäköisiltä ongelmilta mitä käyttöönotto voisi aiheuttaa.

Edellä mainitun mukaisesti valinta oli myös perusteltu, ja itsessään käyttöönotossa ei ilmennyt erityisempiä ongelmia, mutta koska itselle kyseessä oli tuntemattomampi protokolla aiempien seurantaviikkojen mukaisesti, niin ajankäyttö protokollan perusteiden selvittäminen oli hyvä tehdä mahdollisten ongelmatilanteiden selvittämiseksi, ja todennäköisesti tämä myös vähensi käyttöönotolta tarvittavaa aikaa.

Ohjelmoitaessa on esimerkiksi usein kiistaa siitä, että tuleeko opettaa jollekin taholle ensin teoriaa, ja sitten käytäntöä vai toisinpäin. Teoriaa ensimmäisenä opeteltaessa on useimmiten kriteerinä se, että näin tekevä taho on motivoitunut, ja suoraan käytännönläheisesti jonkin asian oppiminen voi aiheuttaa sen, että näin asiaa lähestyvä taho kohtaa sellaisia haasteita, jotka ovat lamaannuttavia, ja käytännössä joitakin abstrakteja aiheita kuten matematiikkaa ei ole mahdollista opetella tietämättä perusteita. (Young, 2016.)

Tämän perusteella voisin pitää mielessä, että teorian opiskelu juuri abstrakteja käsitteitä, kuten protokollakuvauksia opiskellessa on jatkossakin hyödyllistä.

Huijaussivustokampanjoiden torjunta taas on esimerkki eräästä yleisimmästä työtehtävästäni, johon on selvä peruste sekä Suomen laissa, että omissa käyttöehdoissamme, ja torjunta on muutenkin hyvä lisä yritysvastuullisuuteen.

Nämä kampanjat ovat useimmiten helposti tunnistettavissa, sillä ne ovat mitä tahansa hyökkääjien toimesta rakentamia kokonaisuuksia, joilla hyökkääjä pyrkii saamaan pääsyn uhrin käyttämiin järjestelmiin (MITRE s.a.).

Tunnistamisessa helpottaa paljon se, että tunnistamisessa voidaan käyttää usein tuntomerkkejä, jotka ovat erityisesti luonteeltaan sellaisia, missä pankki, luottokorttiyritys tai sosiaalisen median sivustona esiintyvä verkkosivu pyytää tunnistautumaan tai maksamaan jotain, jolla potentiaalista uhria houkutellaan antamaan arkaluontoista tietoa, millä voidaan kirjautua, vaikka juuri uhrin käyttämiin järjestelmiin. (Federal Trade Commission, s.a.)

Tunnistaminen oli aluksi tämän viikon suurimmassa tapauksessa ongelmallista siinä käytetyn sijaintiin rajatun toiminnallisuuden vuoksi, ja valittajan puutteellisten tietojen vuoksi, mutta varmentaminen tämänkaltaisessa tapauksessa on lähes pakollista siksi, että jonkin työpaikan SOC-prosessia ei voisi häiritä tekaistuilla ilmoituksilla huijaussivustoista.

Pohdin, että voisin kehittää mahdollisia vastaavia geolokaation avulla rajattua kampanjoita varten paremman tutkimusympäristön, joka mahdollistaa nopean maantieteellisen sijainnin vaihtamisen tutkimuksia varten, ja muutenkin selvittää, että kuinka tämänkaltaisen järjestelyn toteuttaminen olisi mahdollista.

3.5 Seurantaviikko 5

Maanantai 14.02.2022

Tämän seurantaviikon ja aiemman seurantaviikon välissä on viikon aukko, koska henkilökohtaisen ajan tasapainottaminen työelämän kanssa vaati tauon pitämisen opinnäytetyön kirjoittamisesta. Työtehtäviäni hoiti suunnitellusti tämän ajan sijainen, jonka raportoinnin perusteella pääsin nopeasti aamulla tilanteen tasalle.

Aiemman seurantaviikon mukaisesti päätin, että tekisin kehitystyötä geolokaation perusteella rajattujen huijaussivustokampanjoiden tutkimuksessa avustavaa työkalua.

Ongelma geolokaation perusteella rajatussa huijaussivustokampanjassa onkin siis tutkimuksen osalta pääosin se, että mistä löytyy sellainen välityspalvelin, jolla on sellainen IP-osoite, joka assosioidaan riittävän monessa IP-osoitteita koskevassa geolokaatiotietokannassa toivottuun maantieteelliseen sijaintiin maapallolla, missä tällä tavalla suojeltua huijauskampanjaa ajetaan.

IP-osoitteeseen perustuva maantieteellisen sijainnin selvitys perustuu maatasolla pääsin verkkoblokkien whois-tietoihin, ja näistä tiedoista koostettuihin tietoihin. (ARIN, 2018.) Tästä voi päätellä, että kampanjoita ajavissa palvelimissa ajetaan jotain sellaista lähdekoodia, mikä selvittää vierailijan maantieteellisen sijainnin maatasolla.

Pohdin erilaisia tapoja, joiden avulla saisi tilanteen tullen mahdollisimman nopeasti välityspalvelimen, tai vastaavan ratkaisun tutkimuskäyttöön, ja sain selville, että TOR-verkkoa voi hyödyntää erittäin helposti tähän tarkoitukseen.

Monelle henkilölle TOR-verkko on tutumpi TOR-selaimen kautta, jonka avulla on mahdollista selata verkkosivuja, joiden todellinen ylläpitävä palvelin kätketään usean välityspalvelimen muodostaman ketjun päähän pelkistetysti ilmaistuna. Selaimessa on kuitenkin tarjolla asetustiedosto nimeltään torrc, jonka avulla on mahdollista pakottaa oman selaimen liikenne poistumaan toivotusta maantieteellisestä sijainnista määrittämällä Exit-palvelimen toivottu maakoodi ISO 3166 ALPHA 2 eli kahden kirjaimen muodostavalla maakoodilla. (Accessnow, 2019.)

Kokeilin tätä metodia käytännössä tutkimusympäristössäni ja totesin sen toimivaksi geolokaation näyttävällä verkkosivustolla useaan kertaan. Työpäivän tavoite tuli siis täytettyä, mutta menetelmää pitäisi testata vielä oikean huijaussivuston kanssa käytännössä.

Päivän aikana opin soveltamaan TOR-selainta luovalla tavalla. Menetelmän saa tätä kautta käyttöön myös nopeasti komentorivityökaluille, kuten curlille käyttämällä selaimen yhteydessä käynnistettyä tor-välityspalvelinta, jota myös TOR-selain käyttää.

Tiistai 15.02.2022

Päivän tarkoituksenani oli aloittaa suunnittelu SOC-työskentelyssä käytettävien järjestelmien kehittämiseksi.

Huomioni oli pitkään kiinnittynyt aiemmin luomaani yksisuuntaisesti toimivaan muistutusjärjestelmääni, joka muistuttaa työpyynnöistä, jotka kaipaavat huomiota. Järjestelmässä ei ole erityisesti mitään vikaa, mutta elinkaarenhallinnan osalta olen todennut, että se on elinkaarensa päässä. Muistutukset toimivat yhteen suuntaan vain niin, että niitä ei voi muuttaa helposti jälkikäteen, ja ne voi kuitata tehdyiksi tai tekemättömiksi. Järjestelmä on perustettu aiemmin mainitsemani ja valmistamani Flask-rajapinnan päälle.

Rajapinnassa on myös tiettyjä puutteita, kuten siitä puuttuu versiointi ja tietyt vastaukset rajapinnasta eivät ole omasta mielestäni tarpeeksi RESTful -periaatteen mukaisia. Puutteet ovat olleet valitettavasti sellaisia, että ne on huomannut omien ohjelmointitaitojen kehittyessä ja työtehtävien kehittyessä.

Ongelmien ja puutteiden korjaamiseksi päätin, että suunnittelen korvaavan rajapinnan nykyisten käyttötarpeiden mukaisesti ja sellaiseksi, että sitä on helppo laajentaa niin, että sillä on mahdollista syrjäyttää vanha rajapintajärjestelmä kokonaan.

Suunnittelutyön avuksi otin tutun Insomnia-työkalun, jota käytetään rajapintojen suunnittelussa. Työkalu oli pääosin aluksi tarkoitettu pelkkien rajapintojen testaamista varten, mutta nykyään sen käyttötarkoitus on laajentunut varsin laajaksi ja siinä on mukana myös suunnittelutyökalu, jossa voidaan YAML-syntaksia käyttäen kuvata kokonainen rajapinta.

Syntaksia minun ei tarvinnut opetella, koska Python-kehitystyössä YAML-syntaksi tulee väistämättä jossain vaiheessa vastaan. Tämän sijasta minun piti opetella tavat, millä rajapintoja kuvataan tässä tapauksessa YAML-syntaksilla, ja ohessa selvisi, että käyttämäni syntaksi eli OpenAPI onkin Swaggerin tarjoamien rajapintakehitystyökalujen jatkoa, mikä oli erittäin miellyttävä yllätys minulle. Tämä tarkoittaa käytännössä sitä, että samalla kun kuvailen YAML-syntaksilla jonkin rajapinnan, niin se tulee työn ohessa myös dokumentoitua. YAML-dokumentista taas on jälkikäteen mahdollista luoda staattinen dokumentti yleisempään jakoon.

Onnistuin toteuttamaan CRUD-operaatioita vastaavat päätepisteet rajapinnan pohjalle, jolla käsitellään kehityskohteena olevia muistutuksia, ja nämä näkyvät käytännössä valituilla HTTP metodeilla. Kriteereinä oli, että GET-metodilla on mahdollista hankkia sekä lista kaikista halutuista muistutuksista, tai parametrin avulla vain yhden muistutuksen tiedot. POST-pyynnöllä taas on mahdollista luoda uusi muistutus, ja tarkoitus on, että siihen tulee vastauksena luodun muistutuksen tiedot. Muita metodeja olivat mm. PUT, jolla päivitetään muistutusta, ja DELETE HTTP -metodilla tuhotaan muistutus kokonaan.

Päivän aikana pääsin opettamaan itselleni uusia ominaisuuksia Insomniasta, minkä käyttämistä olin harkinnut pitkään, mutta syystä tai toisesta en ollut käyttänyt niitä. Mielikuva uudesta työkalun ominaisuudesta on toistaiseksi hyvä, ja se generoi selkeästi HTTP metodeita vastaavat pyyntöjen aihiot testausta varten, vaikka rajapintaa itsessään ei olekaan vielä luotu. Päätin, että keskiviikkona voisin keskittyä rajapinnan prototyypitys-prosessiin.

Keskiviikko 16.02.2022

Koska Insomnia-työkalulla ei ole mahdollista luoda tietenkään rajapintaa itsessään, vaan pelkästään suunnitella tai testata sellaista, niin päätin, että keskiviikkona keskityn sovelluskehityksen osalta paikallisen kehitysympäristön luomiseen tätä tarkoitusta varten, ja mahdollisesti rajapinnan kehitys voisi alkaa alustavan suunnitelman mukaisesti.

Tietojen varastointipaikka pitäisi olla tässä tapauksessa MariaDB-tyyppisessä tietokannassa, mikä tarkoittaa sitä, että minun pitäisi suunnitella rajapintaan myös tietokantamalli tauluun, johon tietoa muistutuksista tallennetaan.

Pythonin osalta olin tottunut käyttämään virtuaaliympäristöjä, joiden tarkoituksena on eristää paikalliseksi Python-tulkki ja ympäristö käyttöjärjestelmään itseensä asennetuista Python-paketeista. Mielestäni tämä on hyvä tapa, sillä siinä ei ole riskiä, että käyttäisi jotain sellaista kirjastoa, joka pahimmassa tapauksessa aiheuttaisi vahinkoa käyttöjärjestelmän Python-pakettien riippuvuuksille. Virtuaaliympäristöt ovat myös helppo toteuttaa yhdellä komennolla ja ne voidaan aktivoida heti käyttöön toisella. Käyttämäni IDE osaa myös ajaa kehitysvaiheessa Python koodin tässä virtuaaliympäristössä käyttöjärjestelmän Python-ympäristön sijasta automaattisesti. Vaihtoehto olisi voinut esimerkiksi käyttää IDE:n uutta toimintoa, jossa koodi ajetaan Docker-kontissa, mitä pidin piian järeänä ratkaisuna tähän ongelmaan, joten en tehnyt niin.

Tämän sijasta alkuvaiheen kehitysympäristön toisen osan eli MariaDB-palvelimen paikallinen versio piti olla sellainen, että sen voi käynnistää ja tuhota nopeasti helposti, ja pystyttää vaikkapa toiselle tietokoneelle tarvittaessa paikallista kehitystyötä varten. Tästä syystä

valmistin Dockerin avulla MariaDB-kontin, jonka paljastin docker-virtuaaliadapterin kautta loopback-adapterille tarjolle.

Tietokantaa pitäisi myös pystyä työstämään jollain työkalulla, joten valitsin tehtävään DBeaver-työkalun, jossa on nykyään myös MariaDB-tuki MySQL-serverien lisäksi. Visuaalinen työkalu on erinomainen lisä pelkälle SQL-komentokehotteelle ja auttaa hahmottamaan tietokantaoperaatioita, ja kyselyiden muodostamista.

Muodostettuani tietokantaan yhteydet, luotuani tyhjän tietokannan ja taulun muistutuksille aloin mallintamaan muistutusdataa niin, että se olisi sekä yhteensopiva nykyisen prosessin kanssa, mutta myös sellainen että sitä voisi laajentaa tulevaisuudessa. Rakenne syntyi erittäin helposti, sillä se perustui nykyiseen järjestelmään.

Päivän lopuksi loin Flask-sovelluskehitysverkolla pohjat rajapinnan päätepisteille Insomnialla tekemäni dokumentaation mukaisesti, mutta en yhdistänyt Flaskia mitenkään tietokantaan vielä tässä vaiheessa, vaan pistin luodut reitit palauttamaan keksittyjä arvoja nähdäkseni, että Flask toimii Pythonin kanssa oikein virtuaaliympäristössä. Pääsin ylittämään siis päivän tavoitteet hyvin ja hyödyntämääni aiemmin oppimaani Insomnia-työkalun ominaisuutta, joka oli vielä eilen itselleni uusi.

Torstai 17.02.2022

Suunnitellessani työpäivää sain eteeni aamulla työpyynnön, joka vaatisi tiettyjen lokitiedostojen läpikäymistä, mikä johti uuteen pienkehitysprojektiin.

Työssäni ei ole yleensä minkäänlaista ongelmaa käydä läpi tietyissä tilanteissa lokitiedostoja, mutta ongelmaksi tulee esimerkiksi sellaiset tilanteet, missä tiedostoja on paljon käytäväksi, ja tässä tapauksessa erilaisissa formaateissa. Lokitiedostojen läpikäymisen perustavanlaatuiset työkalut ovat kaikille Linux-käyttäjille tuttuja komentoja, kuten grep, egrep, zegrep, awk, cut jne.

Koska lokien selvittämisessä tulisi selkeästi kulumaan runsaasti aikaa useitten määritellyitten aikaikkunojen takia; päätin, että valmistan sellaisen työkalun Python-kielellä, jolla tiedostoja voi tarkistella helposti antamalla hakuparametrin, ja toivotun aikavälin, jolla tietoja halutaan katsoa.

Lokitiedostoja oli siis kahdentyyppisiä eli sellaisia, jotka ovat selväkielisiä, ja sellaisia, jotka ovat lokirotaation mukaisesti pakattuja tiedostoja, mikä pitää ottaa huomioon. Tiedostojen nimeämisessä noudatetaan kaavaa, jossa lokitiedoston nimessä on mukana aina

päivämäärä tietyssä muodossa. Tämän mielessä pitäen tein funktion, joka generoi listan kaikista mahdollisista päivämääristä tietyllä aikavälillä, ja listan avulla suoritetaan lokikansioista hakuja tiedostoihin, joita halutaan tutkia. Tiedostojen pääte taas määrää, että mitä grep -komentoa kohdistetaan mihinkin tiedostotyyppiin, kun hakufunktio käy löydettyjä tiedostoja läpi, mitkä täsmäävät määriteltyyn aikaikkunaan. Hakujen täsmentämiseksi käytin apuna säännöllisiä lausekkeita eli regexiä, koska grep-tukee niitä.

Tein päivän lopussa työkalusta sellaisen, että sitä olisi mahdollisimman helppo päivittää, eli lisäsin sen melko varhaisessa vaiheessa versionhallintajärjestelmiimme, mistä se oli helppo siirtää tuotantoon, ja sain hankittua työkalulla tarvitsemani tiedot. Opin tässä tapauksessa tekemään parempia ns. wrapper -ratkaisuja Pythonilla, kun haluan suorittaa joitakin järjestelmäkomentoja subprocess-kirjastolla.

Perjantai 18.02.2022

Suunnittelin käyttäväni päivän käyttöjärjestelmäpäivitysten tekemiseen, ja työpaikkailmoituksen tarkistukseen, että esivalmisteluun yhdessä tiimini kanssa.

Saatuani työjonot tyhjäksi päätin aloittaa suunnitellun huolto-ikkunan käyttämisen käyttöjärjestelmäpäivityksen tekemiseen Debian 10 käyttöjärjestelmästä versioon Debian 11 virtuaalipalvelimella, jolla ajetaan ml. ylläpitämääni tiketöintijärjestelmää ja muita SOC-työssä tarvittavia sisäisiä mikropalveluita.

Käyttöjärjestelmää ei ollut erityisemmin mukautettu esim. pakettivarastojen osalta, joten päivitys pääsi alkamaan helposti niin, että määritin pakettivarastolistat käyttämään uuden Debian julkaisun lähteitä heti, kun olin tarkistanut, ettei järjestelmässä ollut muita päivitettäviä paketteja, mikä on tärkeää, kun tehdään Debian-päivityksiä.

Päivitys alkoi normaalisti, ja päättyi tilanteeseen, missä järjestelmä pitää käynnistää enää uudestaan uuden version käyttöönottamiseksi, mutta kohtasin virhetilanteen, missä en saanut enää SSH-yhteyttä muodostettua palvelimeen. Ongelma tosin selvisi melko nopeasti, ja sen syy saatiin rajattua omaan päätelaitteeseeni, sillä muita henkilöitä tiimistäni pystyi kirjautumaan palvelimelle, ja he olivat aiemmin törmänneet samaan ongelmaan. Syy oli se, että Debian 11 käyttöjärjestelmässä käytetään sellaista OpenSSH-palvelimen versiota, joka ei oletuksena enää luota tietynlaisiin RSA-avaimiin. Ongelma korjautui nopeasti sillä, että muutin käyttämääni avaintyyppiä. Pääsinkin siis kirjautumaan melko pian normaalisti palvelimelle, ja tarkistamaan, että se toimii normaalisti kaikkine palveluineen, ja julistin huoltoikkunan päättyneeksi.

Työpaikkailmoituksen osalta en tehnyt juurikaan merkittäviä toimenpiteitä, mutta varaudun siihen, että minun pitää olla läsnä tekemässä työpaikkaan liittyviä haastatteluita, joissa haastatellaan kandidaattia, joka tulisi mukaan tekemään SOC-töitä tiimiini.

Päivän osalta yllättävin asia koski juuri OpenSSH-version muutoksesta tapahtunutta asiaa, mutta haasteesta selvittiin onneksi nopeasti, ja tärkeä päivitys saatiin valmiiksi noin 6 kuukautta ennen Debian 10 -version EOL-aikaa.

Viikkoraportti

Seurantaviikko kului pitkälti SOC-tehtävien ratkaisumenetelmiin liittyvien työkalujen kanssa ja niiden pienkehitysprojekteissa, sekä tärkeissä ylläpitotehtävissä millä SOC-tehtävien infrastruktuuria pidetään yllä. Pienemmässä mittakaavassa aikaa meni myös rekrytointiprojektin aloituksissa, jossa tulee menemään tulevaisuudessa enemmänkin aikaa, kun työpaikkailmoitus julkaistaan.

Sovelluskehitystehtävien, ja erityisesti tiistain ja keskiviikon osalta voisin pohtia, että kuinka järkevää oli käyttää virtuaaliympäristöjä, verrattuna esimerkiksi konttiratkaisuihin. Mielestäni kehitysympäristöä rakennettaessa on kuitenkin jo alan standardi, että kehitystyötä sovelluskehityksessä tehdään johonkin versionhallintajärjestelmään niin, että sovelluskehittäjällä on käytössä jokin paikallinen kehitysympäristö, minkä minä tässä tapauksessa rakensin, joten tämän ratkaisun järkevyyttä ei ole todennäköisesti kannattavaa edes analysoida.

Ympäristön perusperiaatteena on kuitenkin käytännössä kaikissa projekteissani se, että kehitystä tapahtuu paikallisesti, koodi siirretään versionhallintajärjestelmään, mistä se ohjataan tuotantojärjestelmään, ja tuotantojärjestelmän olisi tarkoitus olla identtinen paikallisen kehitysympäristön kanssa.

Pythonin virtuaaliympäristön osalta tässä on pieni haaste siinä, että käyttöjärjestelmien Python-pakettien versioiden erojen vuoksi ympäristöt eivät ole esimerkiksi aivan täsmälleen identtisiä joko itse Python-tulkin osalta, jonka virtuaaliympäristö-moduulia käytetään, mutta ainakin muiden pakettien osalta versiot voivat olla identtisiä, jos käytetään Pythonin pakettienhallintajärjestelmää virtuaaliympäristössä keräämään kaikki virtuaaliympäristön käyttämät paketit listaan kaikkine versioineen.

Ympäristössä, jossa ajetaan vaikkapa sitten useita Python-pohjaisia projekteja voikin tulla ongelmia juuri näistä paketeista johtuvista riippuvuuksista, jos projektit käyttävät eri

versioita vaadituista paketeista. Tämän vuoksi on käytännössä pakollista tai ainakin erittäin suositeltavaa, että virtuaaliympäristöjä käytetään (Real Python, s.a.).

Tässä nouseekin toisaalta kysymys siitä, että voisiko esimerkiksi juuri Dockeria käyttää Pythonin paikallisessa kehityksessä, että myös tuotantoon asettamisessa? Tein aiemmin päätöksen siitä, etten tee näin, koska pidin kehityshetkellä Dockerin käyttämistä liian järeänä ratkaisuna.

Muiden kehittäjien mielestä vaikuttaa siltä, että käytännössä Dockerin tai Pythonin omien virtuaaliympäristöjen käyttäminen on lähinnä mielipidekysymys, sillä molemmilla ratkaisuilla saavutetaan samat lopputulokset eli tehdyn ratkaisun eristäminen enemmän tai vähemmän muusta käyttöjärjestelmästä (Lind, 2019). Tämän vuoksi aion todennäköisesti jatkaa projektin jatkokehityksessä ja tulevissakin Python-projekteissa jo hyväksi havaittua menetelmääni, ellen päätä siirtyä käyttämään esim. lähes automatisoitua CI/CD järjestelmää projektieni tuotantoon laittamiseksi.

Virtuaaliympäristöjen osalta löysin viikon lopuksi itselleni uuden työkalun, virtualenvwrapperin, jonka avulla menetelmäni käyttämisestä pitäisi tulla entistä helpompaa (Reiz, s.a.).

Itse rajapintakehitystyön jatkoa ajatellen reflektoin jo tehtyä työtä muiden kehittäjien hyväksi havaitsemiin periaatteisiin. Löytämässäni materiaalissa käytetään esimerkkinä Node-kieltä ja siihen liittyviä kirjastoja, mutta se on täysin sovellettavissa Flask-kirjastolle myös (Au-Yeung, J & Donovan, R, 2020).

Hyvänä muistutuksena lukiessa minulle tuli, että minun kannattaisi lisätä vielä erikseen versiointi osaksi rajapinnan päätepisteiden suunnittelua, niin samaa rajapinta-palvelua olisi helpompaa laajentaa tulevaisuutta ajatellen. Yleisesti materiaalista ei ilmennyt mitään erityistä tai uutta itselle, ja projekti ei ole edes vielä niin pitkällä, että esimerkiksi virheenhallintaa olisi täysin implementoitu, joten kaikkea siitä ei voi tässä tapauksessa edes soveltaa kunnolla.

Maanantain, ja torstain osalta sovelluskehitys oli luonteeltaan toisenlaista, sillä siinä oli ainoastaan tavoitteena luoda jokin menetelmä tai torstain osalta työkalu, jolla päästään tarvittaessa johonkin tavoitteeseen mahdollisimman nopeasti. Maanantaisen huijaussivujen geolokaatiosuojan ohittava työkalu pääsee vasta käytännössä myöhemmin testeihin, jolloin vasta silloin sen toimintakelpoisuutta voi arvioida kunnolla, mutta torstain lokityökalun luominen voidaan taas nähdä eräänlaisena sijoituksena tulevaisuuteen, sillä sen avulla varmistetaan, ettei jatkossa vastaavanlaiseen tehtävään tarvitse käyttää työaikaa, joten se on siinäkin mielessä jo perustellusti yleishyödyllinen työkalu.

Virtuaalipalvelimen päivityksen osalta perjantaina ei ilmennyt mitään erityistä, mutta jos näin olisi käynyt, niin olin jo varautunut pahimpaan mahdolliseen skenaarioon varmuuskopioilla ja siihen, että esimerkiksi normaali SSH-yhteys palvelimeen olisi menetetty syystä tai toisesta. Jälkimmäinen skenaario vaikutti aluksi tapahtuneen, mutta ongelma selvisi kollegoiden avustuksella helposti. Käyttöjärjestelmäpäivitys on minulle melko lailla rutiinitoimenpide, mutta tätä päiväkirjaa varten päätin lukea siihen liittyen Debianin virallisen päivitysoppaan versiosta 10 versioon 11, minkä perusteella näyttää siltä, etten tehnyt mitään erityisiä virheitä vaan keskityin ydinkomentoihin, joilla päivitys viedään maaliin alusta asti. (Debian s.a.). Hyvänä tapana Debianin materiaalissa on jopa esitelty ilmoitus loppukäyttäjille päivittämisestä, minkä itse ratkaisin käyttämällä huoltoikkunaa, mitä voi käyttää juuri tämänkaltaisten päivitysten tekemiseen.

3.6 Seurantaviikko 6

Maanantai 21.02.2022

Maanantaina tarkoitus oli tutkia erästä aiemmin työpaikallani tekemääni lupaustani SOC-työntekijän perehdytysoppaan päivitystöistä. Materiaalia perehdytykseen juuri SOC-töistä on tällä hetkellä hajautetusti, joten katsoin että olisi hyvä aloittaa suunnittelu siitä lähtökohdasta, että uusi opas on päivitetty, ja yhteen paikkaan koostettu, että päivitetty versio. Ennen uutta mahdollisesti aloittavaa työntekijää materiaalia on muutenkin syytä tarkastella ennen sen esittelemistä ja käyttämistä.

Tällä hetkellä perehdytysmateriaalin tila on sopeutettu toimimaan yhdessä tehtäviin liittyvän perehdytysprosessin parissa, joka on luonteeltaan enemmän käytännönläheinen, kuin teoreettinen, mikä johtuu yrityksessä olevista käytännöistä. Tässä kohtaa on hyvä korostaa, että kyseessä on juuri tähän SOC-tehtävään tarkoitettu materiaali eikä yrityksen toimintaan ja kulttuuriin perehdyttävä materiaali, mitä käytetään perehdytyksissä erikseen.

Kuitenkin mm. koronatilanteen lisäämän etätyön vuoksi voisi olla kannattavampaa, että materiaalia olisi enemmänkin saatavilla jatkossa kirjallisessa ja paremmin organisoidussa muodossa. Tällä hetkellä materiaali on myös erittäin käytännönläheistä, ja siihen liittyy kirjallisen materiaalin ulkopuolelle mm. tarkistuslistoja, joiden avulla voi katsoa osaako perehdytettävä vaadittavia toimenpiteitä perehdytyksen jälkeen vaikkapa tietojärjestelmistä, ja tarkkoja prosessikuvauksia, joiden avulla voidaan kirjallisessa muodossa selventää ydinasiat, joita tehdään vaikkapa tietyntyyppisten tietoturvapoikkeamien ratkomisessa.

Aiemmin luotua kirjallista materiaalia olisi tarkoitus tietysti uudelleen käyttää uuden oppaan kirjoittamisen pohjana, mutta uusia näkemyksiä oppaan kirjoittamisen tueksi voisin katsoa jostain ulkopuolisesta kirjallisesta lähteestä, kun itse kirjoitusprosessi alkaa.

Materiaalin toimivuudesta ja soveltuvuudesta voisi olla myös mahdollista kerätä palautetta, jotta opas pysyisi ajan tasalla. Käytännössä itselläni on kaikki mahdollisuudet päättää siitä, että missä laajuudessa oppaassa mitäkin asiaa käsittelen, ja tulen kirjoittamaan oppaan todennäköisesti pääosin itse pyytäen siitä kuitenkin muilta tiimin jäseniltä palautetta.

Maanantai oli siis tämän asian osalta pääosin alustavaa suunnittelutyötä, jossa määrättiin suuntaviivat perehdytysoppaan luomiseksi.

Tiistai 22.02.2022

Tarkoituksenani oli siirtyä SOC-työntekijän perehdytysoppaan valmistuksessa sellaiseen vaiheeseen, että sen kirjoittaminen voidaan aloittaa.

Pohdin useita erilaisia tapoja toteuttaa opas jo sovellustasolta alkaen, mutta päädyin esim. Googlen toimistosovelluksien sijasta siihen, että kirjoitan oppaan kokonaan markdown-muodossa, mikä tarkoittaa sitä, että raakamateriaali koko oppaaseen tulee olemaan tekstimuotoisena, mistä se voidaan muuntaa ihmisten helpommin ymmärtämiin dokumenttimuotoihin, tai vaikka yrityksen varsinaisiin tiedonhallinta-järjestelmiin, mistä tietoja jaetaan keskitetysti työntekijöille.

Tiedoston säilytyspaikkaa piti toisaalta pohtia, mutta päätin, että se voi sijaita yrityksen versionhallintajärjestelmässä. Tämä varmistaa käytännössä sen, että muokkaukset pysyvät varmistetusti tallessa, muokkaushistoria näkyy kaikille yrityksessä ja lisäksi muut työntekijät voivat ottaa osaa dokumentin parannukseen tekemällä git-pohjaisessa versionhallinnassa muutospyyntöjä.

Kävin repositorion eli tiedoston säilytyspaikan perustamisen jälkeen käymässä lävitse jo olemassa olevaa perehdytysmateriaalia, ja ryhdyin siirtämään sitä markdown-tiedostoon sellaisenaan, jos materiaali oli ajan tasalla, ja jos ei niin tein siihen tarvittavat muutokset. Samalla, kun tietoa tuli koottua loin myös tyhjiä otsikoita oppaaseen, jotka olisi tarkoitus kirjoittaa myöhemmin auki. Kokonaisuudessaan oppaan pohja käsittelisi noin 60 erilaista asiaa nykyisessä muodossaan ja valmiina otsikkomäärän perusteella.

Perustustyöt tulivat siis kokonaisuudessaan valmiiksi, ja pääsin työstämään opasta joiltakin osilta laajemmin, ja osaltaan opas on kokonaan kesken juuri auki jätettyjen väliotsikoiden osalta. Koska nämä avoimet otsikot ovat kokonaan uusia oppaan kannalta otin tavoitteeksi, että nämä otsikot siirretään repositoriossa päähaarasta omaan haaraansa, missä niitä voi työstää, kunnes ne ovat valmiita. Samalla voi keskittyä päivittämään oppaan valmiilta osilta ajan tasalle, jos niin ei jo ole tehty.

Keskiviikko 23.02.2022

Päätin, että laajennan keskiviikkona aiemmalla viikolla työstämääni uutta Flask-rajapintaa.

Aiemman raportoinnin mukaisesti työ jäi viimeksi kesken siitä kohdasta, missä sitä olisi voinut yhdistää testiympäristön tietokantapalveluun. Yhdistäminen tapahtui kuitenkin melko nopeasti käyttämällä MariaDB:n tarjoamaa Python-kirjastoa, ja asentamalla siihen liittyvät paketit Linux-distribuutiolleni, eli kirjasto sai ns. kursoriyhteyden tietokantaan, eli sinne olisi mahdollista syöttää normaaleja SQL-lausekkeita Flask-ohjelman kautta.

Jo tutuksi tulleen Insomnia-työkalun avulla testasin HTTP-pyyntöjä sitä mukaa, kun sain tehtyä Flask-ohjelmaa valmiimmaksi eli korvattua testiarvoja palauttavia reittejä sellaisiksi, että ne palauttaisivat tai syöttäisivät dataa MariaDB-tietokantaan.

Kaiken tämän keskellä jäi tietysti mieleen se, että Flask-sovelluksen autentikointijärjestely tulisi toteuttaa niin, että se tukee useaa autentikointimenetelmää, mikä pitää ottaa huomioon, jos sovellusta käyttää esim. React-pohjaisen käyttöliittymän kautta ihmiset ja rajapinnan kautta palvelinsovellukset. Päätin tässä vaiheessa, että bearer-tokenit saavat toimia rajapintasovelluksien pääsyoikeuksien varmennuksena, ja ihmiskäyttäjät saavat JWT tokenin taas kirjautuessaan onnistuneesti LDAP-tunnuksilla, mitä aiemmissa vaiheissa raportointia testailin prototyypillä onnistuneesti.

Koska ajattelin, että rajapintaa kehitetään ensimmäisenä, niin saatoin bearer-tokenit toimintaan ensimmäisenä ja asetin kaikki tekemäni HTTP-reititykset kirjautumisen taakse. Itse tokenit päätin varastoida suolattuna, että tarkistussummattuna sovelluksen tietokantaan nyt kun siihen oli saatu muodostettua toimiva yhteys. Tuki usealle kirjautumistavalle lisättiin myös käyttämällä Flaskin kanssa yhteensopivia kirjastoja, vaikkei mitään JWT tokenia ei ollut otettukaan vielä käyttöön.

Käytännössä työpäivän aikana valmistui sellainen pohja rajapinnalle, jota on helppo kehittää useaan suuntaan, eli siinä on jo reittejä, joita voi käyttää, ja alan standardien mukaisesti suojattu käyttäjienhallinta ainakin rajapintatunnuksien osalta jo kehitysversiossa,

vaikka rajapinnasta jääkin vielä paljon kehitettävää. Seuraavana työnä voisi selvittää erityisesti tähän liittyen, että millä keinoilla on järkevintä tehdä versiointi, koska siihen Flask ei ainakaan nykyisen oman käsitykseni mukaan tarjoa mitään valmista ratkaisua vaan oma menetelmä pitää päättää itse.

Torstai 24.02.2022

Torstain työpäivälle en suunnitellut mitään erityistä toimintaa, mutta työpäivän aikana törmäsin erääseen toimenpiteeseen, joka toistuu aina välillä, ja joka sisältää ylimääräisiä välivaiheita.

Joissakin tilanteissa, kun käyttäjien tileillä sijaitsevilla palvelimella tapahtuu sellaista toimintaa, mistä ilmoitetaan käyttäjälle, ja mihin käyttäjät eivät reagoi tulee tarpeelliseksi usean ilmoituksen jälkeen useimmiten joko keskeyttää tämänkaltaisen asiakkuuden toiminta pysyvästi tai väliaikaisesti, tai sitten lievemmissä tapauksissa esimerkiksi sammuttaa kyseisen käyttäjän virtuaalipalvelininstanssi, ja ilmoittaa käyttäjälle miksi näin tapahtui käyttäjän laiminlyönnin seurauksena. Yleensä tässä tilanteessa käyttäjä viimeistään reagoi korjaamaan ongelmaa. Tässä tapauksessa asiakas ei ollut reagoinut toistuviin ilmoituksiin SSH-protokollalla tehtävästä haittaliikenteestä.

Tämänkaltaisista toimista pidetään lokia yllä, mutta ongelmana on aina se, että näitä kolmea toimenpidettä ei ole yhdistetty toisiinsa automatisoidusti, eli sammutus, lokimerkintä ja tiedottaminen pitää tehdä erikseen.

Päätin, että korjaan prosessia tekemällä tähän sopivan yksinkertaisen Python-skriptin, jota testaan omilla virtuaalipalvelimillani.

Melko nopeasti sain valmistettua sellainen työkalun, joka pystyy sammuttamaan määrättyjä instansseja, tunnistaa instanssin perusteella käyttäjätunnuksen, jota sammutus koskee ja tekee lokimerkinnän asiasta, mutta siitä puuttuu enää vain yksi oleellinen toiminto, joka on käyttäjän tiedottaminen aiheesta tiketöintijärjestelmässä, minkä rakentamisen jätän myöhemmälle ajalle.

Perjantai 25.02.2022

Suunnittelin, että viimeistelen torstaina aloittamani skriptin valmiiksi perjantaina, ja teen pientä uutiskatsausta kyberturvallisuuden osalta liittyen Venäjän julistamaan sotaan Ukrainaa vastaan, millä voi olla vaikutuksia omaan työhöni.

Skriptin viimeistely onnistui lopulta erittäin nopeasti, sillä minulla oli jo aiemmasta rajapintapalvelustani tarjolla sellainen päätepiste, joka pystyy ottamaan vastaan vastaanottajan sähköpostiosoitteen, viestin otsikon ja viestin itsessään. Tarkoituksena on siis, että palvelimen sammutuksen, ja lokimerkinnän jälkeen asiakasta vielä tiedotettaisiin automaattisesti toimenpiteestä, joten suurin työ oli oikeastaan kirjoittaa sopiva viestipohja tätä varten.

Uutiskatsauksen perusteella puolestaan oli jo selvää, että vaikutukset oman työni kenttään voivat olla mahdollisia, sillä Ukrainaa vastaan tarkoitettua haittaohjelmaa oli levinnyt hallitsemattomasti ml. Latviaan, että Liettuaan (Catalin, 2022). Tämä tarkoittaa käytännössä töiden osalta sitä, että haittaohjelmista kertovat raportit voivat olla kasvussa, jos leviäminen on hallitsematonta, mikä lisää käyttäjiin kohdistuvia toimia, jos toimintaa havaitaan verkoissamme tavalla tai toisella.

Palvelunestohyökkäyksiä liittyen finanssisektoriin on lisäksi liikkeellä Ukrainaa vastaan (State Service of Special Communication and Information Protection of Ukraine, 2022). Palvelunestohyökkäyksissä käytetään yleensä murrettuja palvelimia, joten esimerkiksi verkon valvonnassa on syytä olla tarkkana mahdollisiin piikkeihin liikenteestä, joka poistuu asiakkaiden palvelimilta. Nykypäivänä liikennemäärät, mitä vaaditaan jonkin palvelun kaatamiseen ovat melko suuria, joten piikit erottuvat varmasti selkeästi.

Viikkoraportti

Viikon tapahtumat ovat selkeästi jaettavissa pääosin kahteen eri kategoriaan, eli sovelluskehitystehtäviin, jotka tukevat tulevaisuudessa omaa ja muiden tiimissä työskentelevien ihmisten arkea ja SOC-tehtävien kuvailuun perehdytysmielessä. Vähäisemmälle tarkastelulle jää uutiskatsaus, josta tein jo johtopäätöksiä, ja yleishyödyllinen aikaa säästävä skpritin valmistus taas on yleishyödyllistä työtä, joka selkeyttää prosesseja samalla, ja vähentää virheiden mahdollisuutta.

Perehdytysoppaasta täytyy tehdä laajempi huomio, että sen tarkoitus koskee tässä tapauksessa pelkästään työpaikalla SOC-roolia. Työpaikalla on erikseen olemassa työntekijöille luettavaksi tarkoitettu manuaali, joka kertoo työpaikasta ja sen kulttuurista yleisesti, joten oppaan ei ole tarkoitus vastata täysin yleisiin kysymyksiin ollenkaan. Lisäksi niitä käydään muutenkin läpi omissa yleisissä perehdytystilaisuuksissa, ja tämän lisäksi on käytäntönä, että uusien työntekijöiden aloittaessa jokainen yrityksen tiimi esittelee itsensä. Useimmiten uusi työntekijä saa myös jonkin vertaisen auttamaan ensimmäisten päivien aikana esimiehensä lisäksi tehtävien aloittamisessa, ja ulkomaiset työntekijät pyritään kutsumaan tutustuttavaksi pääkonttorille noin viikoksi koronatilanteen mukaan.

Koska opas on vasta kehitysvaiheessa, niin päätin että ennen sen edistämistä pidemmälle etsin ja luen HR-ammattilaisten vinkkejä, ja ohjeita materiaalin kirjoittamiseksi yllämainitun rajauksen mukaisesti. Society for Human Resource Management (s.a.) on esimerkiksi tiivistänyt hyvin vaihe vaiheelta itse tuottamisen jatkuvaksi 9 vaiheen prosessiksi, joista viimeinen vaihe kehottaa erityisesti päivittämään opasta tarpeen tullen.

Päivittämistarpeet, ja potentiaalinen uusi rekrytointi ovat laukaisseet tämän prosessin työpaikallani, joten päivittämällä tiimiä koskeva materiaali varmistetaan, että työntekijää perehdytetään erikoistehtäviinsä oikein ja ilman ristiriitaista tietoa.

SHRM:n materiaaliin viitaten itse kirjoittamisesta ovat valmiina jo kaksi ensimmäistä vaihetta eli aiemmista materiaaleista on käyty lävitse jo aiemmat voimassaolevat politiikat, jotka ohjaavat SOC-työtä. Politiikkoja ovat esimerkiksi suorat käskyt, ja ohjeet siitä miten tietynlaiseen käyttäjäaktiviteettiin suhtaudutaan yrityksessä, ja muut vastaavat poikkeusohjeet. Otsikkotasolla SHRM:n toisen vaiheen mukaisesti olen jo käytännössä tehnyt päätökset siitä, että mitä oppaaseen tulee, ja tämän nojalla jo kolmas vaihe, eli erilaisten prosessien kuvaukset ja vaiheet ovat osasta otsikkoja jo kuvattuina. Seuraavaksi askeleeksi oppaassa jää siis vaiheen kolme viimeistely.

Aiemmassa materiaalissa ei ole juurikaan perusteltu tiimin olemassaolon tarkoitusta tai esimerkiksi missiota, visiota tai arvoja pääosin siksi, että niistä asioista kerrotaan juuri muussa yrityksen materiaalissa, mutta tarkemmin tarkasteltuna nämä asiat voisi olla hyvä määritellä oppaaseen erikseen tiimin näkökulmasta niin, että ne ovat linjassa myös yrityksen pääarvojen kanssa. Karkealla tasolla tiimin toiminta perustuu matalimmalla yritysvastuun tasolla käyttöehtosopimuksen noudattamisen valvontaan, mikä taas perustuu Suomen lakien noudattamiseen, ja korkeammalla tasolla yritysvastuussa turvallisen internetin ylläpitämiseen omien IP-verkkojen osalta. Eettisiin asioihin otetaan myös kantaa eli esimerkiksi laittoman materiaalin näyttäminen on kiellettyä, kuten myös materiaali, jossa yllytetään väkivaltaan tai ylipäätään äärimmäisen väkivaltaisen materiaalin näyttäminen on kiellettyä. Missio, visiot ja arvot tiimin sisäisesti voisivat näin olla siis jo käyttöehtosopimuksen pohjalta määriteltävissä.

Miksi nämä asiat tulisi siis myös määritellä? On itsestään selvää kenelle tahansa henkilölle, joka on ollut työelämässä, että näin tapahtuu, mutta syitä sille on useita. Rekrytointitoimisto HAYS:in mukaan (s.a.) kyse on erityisesti siitä, että yrityksen sisäiselle yksilölle muodostuu jonkinlainen identiteetti, joka rakennetaan ydinarvojen ympärille, missä visio määrittää yksikön roolin yrityksen tulevaisuudessa. Ydinarvoilla on myös mahdollista ohjata työntekijöiden toimintaa, sillä jos niitä korostetaan, niin niistä tulee keskeisiä työkaluja

päätöksenteossa. Itselleni esimerkiksi käyttöehtosopimus on eräistä tärkeimmistä ohjenuorista, sillä se kattaa käyttäjien oikeuksien lisäksi myös yleiset vastatoimet, mihin voidaan ryhtyä sopimusrikkomuksia havaittaessa.

Kirjoitustyyliä jatkoa ajatellen ajattelin ottaa esimerkkiä peliyhtiö Valven tekemästä työntekijäoppaasta. Koska suurin osa materiaalista koskee juuri prosessien kuvausta, ja on luonteeltaan vakavaa, on riskinä, että materiaali luettavuus kärsii. Valven oppaasta on tehty havaintoja, että tärkeä teksti voidaan kirjoittaa sellaiseksi, että se puhuttelee lukijaa käyttämällä sinuttelevia pronomineja, ja on myös erittäin yksityiskohtainen. (Kearl, 2016.)

Itse Valven (2012) kirjaa luettaessa voi yleisesti havaita, että siinä on joitakin muitakin aiheita, jotka voisi olla hyvä käsitellä oppaassa, kuten tiimin historian esittely aiemmin mainittujen mission, vision ja arvojen sisällyttämisen lisäksi. Valitsin juuri tämän kirjan siksi, että yrityksen käyttämä matala organisaatiomalli on melko lähellä oman työpaikkani tapaa organisoitua.

Sovelluskehityksen osalta taas kehitystyö keskittyi juuri REST API autentikoinnin implementointiin juuri rajapintaa käyttävien sovelluksien osalta, ja esimerkiksi JWT-autentikoinnin toteuttaminen jäi tekemättä tarkoituksella. Rajapintoihin näyttää olevan neljä suosittua tapaa toteuttaa itse autentikointiprosessi, joista yksi oli juuri valitsemani bearer-token, jonka esittämällä HTTP pyynnössä saa tokenin haltija itselleen pääsyn rajapintaan (Levin 2019) HTTP Basic -autentikointimalli, ja API-avaimet ovat tähän rinnastettavia, koska loppupeleissä niiden autentikointidata kulkee tavalla tai toisessa HTTP pyyntöjen otsaketiedoissa mukana.

Poikkeavia, mutta yleisiä tapoja näistä ovat OAuth ja OpenID. Käytännössä OAuth olisi bearer tokenia parempi, jos rajapinnalla olisi useita käyttäjiä, ja jos useat rajapinnan käyttäjät tulisi päästää sisään eri rooleilla. Toisin sanoen OAuth sopii enemmän tyypilliselle julkiseen käyttöön tarkoitetulle verkkosovellukselle, kun käyttäjiä on paljon. Tästä syystä katson, että jatkan bearer-tokenien käyttöä. Tietysti, jos tokeneille vaaditaan joskus rajattuja pääsyjä tiettyihin uuden rajapinnan osiin, olisi mahdollista tehdä vaikkapa käyttäjätietokantaan oma MariaDB taulu, joka määrittää kunkin bearer-token käyttäjän ID:llä saatavilla olevat rajapinnan HTTP REST päätepisteet. Tämän perusteella pidän bearer-tokenin käyttämistä toimivana valintana edelleen varsinkin, kun tokenit varastoidaan ensin suolattuna, ja toiseksi myös tarkistussummattuna.

Suolauksen käyttö standarditavan eli pelkän tarkistussummauksen käytön sijasta varmistaa sen, että salauksessa on varauduttu salasanatietojen murtoja vastaan myös sellaisessa tilanteessa, että vihamielinen taho, jolla on pääsy salasanatietokantaan ei voi

murtaa tallennettuja salasanoja pelkällä väsytyshyökkäyksellä, vaan tarvitsee myös avuksi rainbow-taulun, mikä kuluttaa hyökkääjän resursseja merkittävästi. (Arias, 2021.)

Ihmiskäyttäjille, jotka käyttäisivät rajapintaa jonkin selainkäyttöliittymän kautta esim. React-kirjastolla toteutettuna voisi olla taas tarjolla aiemmin pohdittu JWT tokenin myöntö. Tarkemmin tutkittuna JWT käyttö sisältää paljon hyödyllisiä ominaisuuksia, että riskejä.

Ensinnäkin JWT puolesta voi sanoa, että tokeniin luottaminen on helppoa, kun se myönnettäessä oikein onnistuneen kirjautumisen kanssa allekirjoitetaan palvelimen puolella. Tämä tarkoittaa sitä, että kun asiakas esittää palvelimelle HTTP pyynnössä JWT tokeninsa, palvelin voi purkaa sen avaimellaan. Jos purku onnistuu, niin tokeniin voi pelkistetysti sanottuna luottaa. Yleensä tokeniin sisällytetään myös jokin voimassaoloaika, minkä ajan se on voimassa. Käyttö myös varmistaa sen, ettei ole tarvetta kuin tunnistaa palvelimen puolella tokenista, että kenelle token kuuluu, jotta sen käyttäjälle voidaan näyttää juuri haltijan mukaisia tietoja.

Vasta-aiheet käytölle liittyvät pääosin siihen, että tokenia voidaan käyttää vahingossa väärin joko konfiguraatiovirheillä tai olemalla piittaamatta hyvistä käyttöperiaatteista. Konfiguraatiovirheet ovat yksinkertaisia, mutta helppoja myös välttää esim. tokenia ei tulisi lähettää verkossa selväkielisenä MITM hyökkäyksen minimoimiseksi.

Yksi riski liittyy myös potentiaaliseen skenaarioon, jossa React-pohjaiseen sovellukseen kohdistuu XSS hyökkäys, ja jos token on tallennettu käyttäjän selaimen paikalliseen varastoon evästesäiliön sijasta, se on minkä tahansa vihamielisen sivulle injektoidun skriptin luettavissa. (Copes, 2021.)

Koska kyseessä on pääosin sisäisen työkalun luominen, jolla ei ole mitään näkyvyyttä julkiseen internetiin, ja koska XSS hyökkäys on pääasiassa käyttäjälle renderöityvän sisällön sovelluskehitykseen liittyvä uhka, niin se voidaan ottaa huomioon tutkimalla millä keinoilla XSS hyökkäyksiä mitigoidaan sekä React-sovelluksessa, että Flask-kirjastossa.

Tämän perusteella jatkan JWT-ratkaisuni tukemista ja totean, että se on käyttökelpoinen, joten sen implementointi saattaa olla lähiviikkojen projekti siitäkin huolimatta, että mitään React-käyttöliittymää ei kerkeäisi rakentamaan opinnäytetyön aikana.

3.7 Seurantaviikko 7

Maanantai 28.02.2022

Viikon alkajaisiksi ja kuukauden loppuessa ajattelin ajaa päivityksiä Freescout-tiketöintijärjestelmään rutiinitehtävien ohella.

Päivityksien asennuksissa ei ilmennyt mitään tavallisuudesta poikkeavia ongelmia, ja ne olivat tyypilliseen tapaan ohitse minuuteissa. Tämän sijasta täysin eri asiaan liittyen erästä epäiltyä tekijäinoikeusrikkomusta käsitellessäni törmäsin tahoon, joka väitti edustavansa erästä brändiä, ja suojelevansa brändin oikeuksia. Henkilö ei kuitenkaan suostunut yhteistyöhön tapauksen varmennusvaiheessa, sillä henkilö ei toimittanut brändin omistajan allekirjoittamaa dokumenttia, josta selviää edustajan tarkat oikeudet toimia brändin nimissä, ja tarkemmin tarkasteltuna valituksessa mainittu domain osoitti A tietueiltaan aiemmin esitellyn ja tunnetun Cloudflaren verkkoon. Valittajaa kehotettiin toimittamaan pyydetyt tiedot ja suuntaamaan valitus oikean verkon omistajalle, että tapaus selviäisi.

Päivän aikana selvisi myös, että eräs asiakaspalvelutiimille ja osittain SOC-tiimille oleelliseen palvelimeen tulisi päivitystarpeita, sillä Debian 9 käyttöjärjestelmän LTS tuki loppuu kesäkuun lopussa 2022, joten sen päivittäminen on erittäin ajanmukaista. Tämän lisäksi eräs työkalu, jota ajetaan palvelimella, tarvitsee Debianin pakettivarastoista uudemman version Python-kielen tulkista, joka on saatavilla vain uudemmissa versioissa. En ole itse tekemässä päivitystä tällä kertaa, kuten viikolla viisi, mutta ajattelin että tarkistan tiistaina 8.3, että palvelimella ajettavat omalle työlle tärkeät työkalut toimivat edelleen huoltoikkunan jälkeen. Yhteensopivuusongelmia ei ainakaan olisi odotettavissa, joten asia ei edellytä erityisempiä toimenpiteitä tämän asian osalta.

Tulevana työtehtävänä tein itselleni päivityksestä vastaavan tahon tekemään projektiin oman aliprojektin yhteensopivuusongelmien tarkastuksesta niin, että saan itselleni ilmoituksen päivityksen valmistuttua, ja rutiineista poikkeavan valittajan osalta jään odottamaan valittajan vastausta. Aiempien kokemuksien perusteella vastausta ei tule tapahtumaan tuntemattomista syistä.

Tiistai 01.03.2022

Tiistaina suunnittelin, että selvitän lyhyehkösti erilaisia mahdollisuuksia uusien uhkatietosyötteiden käyttöönottamiseksi.

Yrityksessä käytetään jo tällä hetkellä useita erilaisia syötteitä, joista osa toimittaa tietoja pyytämättä, ja osa taas toimittaa tietoja käyttäjien haitallisesta toiminnasta vain, jos tietoa joko hakee käyttämällä palveluntarjoajan rajapintaa, tai jos raportit tilaa syötteen luojalta. Tarjoajat edustavat usein CERT-toimintoja eri maissa ja yksityisiä henkilöitä, tai tietoturva-alan yrityksiä.

SOCRadar (2021) oli tuottanut esimerkiksi kattavan listan erilaisista uhkatietosyötteistä. Kuitenkin listaa läpi käydessäni havaitsin, että osa listatuista syötteistä oli joko jo valmiiksi meillä käytössä, puutteellisia tai ei sovellettavissa yrityksen toimintaan. Esimerkiksi huijaussivuja torjuva Google Safe Browsing olisi yleisellä tasolla erittäin hyödyllinen, ja sen tarjoama rajapinta Web Risk myös, mutta käytännössä se ei vain sovi yrityksen toimintamalliin. Tämä johtuu juuri tässä tapauksessa esimerkiksi siitä, että Googlen tarjoama rajapinta ottaa vastaan URL osoitteita, jotka Google raportoi vastauksissaan turvallisiksi tai epäturvallisiksi. IaaS-mallissa, missä loppuasiakas ajaa ilman palveluntarjoajan hallinnointia palveluitaan tätä ei ole mahdollista käyttää. Osissa syötteistä taas on pitkät hakuprosessit, joten täytyy pohtia sitäkin, että kuinka paljon aikaa haluaa käyttää ylipäätään käyttöönottoon varsinkaan, kun syötteen hyödystä ei ole tarjolla minkäänlaista näyttöä.

Jotkin syötteet taas esimerkiksi, kuten Phishtank vaikuttivat ulkoiselta olemukseltaan toimivilta, ja niissä oli erittäin hyödyllisiä toimintoja kuten IP-verkkojen AS-numerolla tehtävä haku, joka listaa kaikki palveluun raportoidut aktiiviset ja epäaktiiviset uhkat tietyn ISP:n alla. Tekniseltä tasoltaan tämä oli käytännössä ideaaliratkaisu, mutta ongelmana oli se, että siinä oli liian vähän tietoa meidän verkoistamme eikä Phistankista löytynyt mitään sellaista uhkaa, mikä ei olisi tullut muilta syötteiltä tietoon.

Katsaus oli päivän osalta sikäli hyödyllinen, sillä siinä kävi ilmi, että tietysti vaikka muita uhkatietosyötteitä, kuin ne mitä jo nyt yrityksessä on käytössä on olemassa, niin välttämättä erityisesti pienempien ja epätunnettujen syötteiden laadussa on paljon puutteita.

Keskiviikko 02.03.2022

Keskiviikon osalta huomasin, että minulla oli kalenterissani ennakkoon sovittu palaveri esimiehen ja kollegoiden parissa. Aiheena olisi lähinnä, että kukin tiimissä kertoo viimeaikojen kuulumisia, ja palaverin vetäjä kirjoittaa lyhyet muistiinpanot aiheesta.

Aiemmista rekrytointisuunnitelmista tuli hivenen keskustelua siinä määrin, että aiemmin luonnosteltu työpaikkailmoitus oli siirretty yrityksen sisäiselle ammattirekrytoijalle, jonka tehtävä on etsiä sopivia kandidaatteja SOC-työtehtäviin ja lisäksi tietysti viimeistellä työpaikkailmoitus sellaiseksi, että se on julkaisukelpoinen. Viime viikon teemaan liittyen

arvoista käytiin myös keskustelua siitä, että mitä yrityksen omat pääarvot merkitsevät kenellekin. Tästä on hyötyä aiemmin mainitun SOC-perehdytysoppaan kehityksessä, mihin suunniteltiin sisällytettävän tiimin sisäisiä omia arvoja, jotka ovat yrityksen pääarvojen kanssa linjassa.

Etätyöskentelyn ergonomiaa ajatellen tilasin myös päivän päätteeksi itselleni uuden toimistotuolin.

Torstai 03.03.2022

Suunnittelin, että torstaina voisin valmistella Pythonilla luokkaa, jolla olisi mahdollista toteuttaa automaattista päätöksentekoa SOC-tapahtumien käsittelyssä

Tämä luokka toimisi käytännössä yksinkertaistettuna niin, että siihen syötetään JSON-dataa, joka sisältää SOC-tiimille tulevia valituksia ulkoisista lähteistä, ja sitten käsittelyn jälkeen luokka palauttaa joko virheilmoituksen siitä, että viesti ei ole automaattisesti tulkittavissa tai sitten onnistuessaan luokka palauttaisi kokonaisen valmiiksi muotoillun ja viestipohjan ympärille rakennetun ehdotuksen siitä, että miltä loppukäyttäjälle menevän viestin kannattaisi näyttää. Vaatimuksenmäärittely on siis agile-tapaa mukaillen täsmennettynä sellainen, että luokka ratkaisee toistuvaa kaavaa noudattavia ilmoituksia, ja sellaisia ilmoituksia, joissa on selkeä koneluettava formaatti esim. X-ARF on hyvä esimerkki koneluettavasta SOC-tiimille tarkoitetusta ilmoituksesta. Kokemuksesta tiedän, että toistuvaa kaavaa noudattavia ilmoituksia on erittäin paljon, ja koska näiden tehtävien kuvaaminen tekstipohjaisesti on mahdollista niin se tarkoittaa käytännössä lähes aina sitä, että tehtävä on erittäin helposti automatisoitavissa ohjelmoinnin kautta.

Suunnitteluvaiheessa agilea mukaillen pitäisi sitten miettiä vielä tärkeitä ominaisuuksia tähän luokkaan, ja keinoja millä ominaisuuksia toteutetaan. Yhteistä kaikille ilmoituksille saapuessaan tietenkin se, että niiden mukana on tekstiä, joista osa on erittäin hyödyllistä tietoa loppukäyttäjälle, ja osa tiedosta saattaa olla täysin turhaa tai emotionaalisesti latautunutta. Emotionaalisesti latautunut sisältö on esimerkiksi roskapostivalituksiin liittyvä lisätieto valittajalta, joka sisältää usein ylimääräisiä valitukseen kuulumattomia kirosanoja, jotka liittyvät usein vaatimukseen roskapostituksen lopettamiseksi. SOC-tehtävien kannalta juuri tämänkaltainen sisältö on sellaista, jonka esimerkiksi voi jättää kokonaan suoraan pois. Säännölliset lausekkeet ovatkin sellaisia, että niillä voi poimia leipätekstistä tärkeitä tietoja, ja avainsanoja. Yleisluontoisesti kuvattuna valituksen leipätekstistä tärkeimmät tiedot ovat lokitiedot aikaleimoineen, domain-nimet ja IP-osoitteet. Pääsääntöisesti näitä kaikkia tarvitaan yhdessä lähes aina tapauksen käsittelyyn, mutta ei aina. Tämän takia päätin, että ennen tietojen keruuta tulisi tehdä tapauksien luokittelutyökalu, joka

määrittää, että mitkä tiedot ovat oleellisia automaattisen päätöksenteon tueksi. Luokittelu perustuisi kahteen tapaan, joista ensisijainen perustuu siihen, että luokittelujärjestelmä yrittää tunnistaa viestin lähettäjän sähköpostiosoitteen perusteella, ja leipätekstin avainsanojen perusteella, että mistä asiasta on kyse, ja toissijaisesti arvaa pelkillä avainsanoilla mistä asioista on kyse. Virhetilanteessa, jossa useita avainsanoja löytyy sekalaisesti, määritetään avainsanojen määriin perustuen mistä asiasta on kyse. Avainsanoja etsittäisiin ensisijaisesti otsikosta ja toissijaisesti leipätekstistä.

Ajatustasolla pidän tätä projektia luonteeltaan sellaisena, että sen ideaa voi jatkokehittää suunnitellusti, ja pidän sitä teknisesti erittäin mahdollisena toteuttaa, ja toteutuessaan se poistaisi monta toistuvaa välivaihetta tiimini työskentelystä.

Perjantai 04.03.2022

Suunnittelin, että voisin jatkaa perjantaina aloittamaani projektia SOC-tiimille tulevien toistuvan tyyppisten ilmoituksien automaattisen käsittelyn kehittämiseksi.

Leipätekstin analytiikasta tuli mieleen itselle uusia ajatuksia, sillä domain-nimet, ja IP-osoitteet ovat sellaisia asioita, joita käytetään erityisesti jonkin käyttäjän tunnistamiseen. Julkisessa pilvipalvelussa on tietenkin ongelmana vielä se, että IP-osoitteet saattavat vaihtaa omistajia, joten tämä täytyy pitää automaattisessa päätöksenteossa mielessä. Tämän takia torstaina ajattelin jo, että luokka antaakin vain ehdotuksensa ratkaisusta johonkin ongelmaan, jota ei ole vielä validoitu.

Säännölliset lausekkeet ovat kuitenkin sellainen asia, jolla näitä avaintietoja, kuten domain-nimiä ja IP-osoitteita pystyy etsimään. Kohdennusta varten siis tarvitsee suunnitella tai käyttää jonkun tahon valmiiksi luomaa säännöllistä lauseketta, joka etsii leipätekstistä kaikki domain-nimet, ja IP-osoitteet, joita pikaisilla hakukonehauilla löytyy paljon, ja olen käyttänyt vastaavia säännöllisiä lausekkeita aiemminkin. IP-osoite on luonteeltaan sellaista tietoa, jonka voi yhdistää aikaleimatiedoilla melko luotettavasti asiakkuuteen, mutta esimerkiksi domain-nimissä olisi kannattavaa tehdä niin, että domain-nimelle tehdään A ja AAAA-tietueista kysely domain-nimen nimipalvelimelta, jotta siihen liittyvät IP-osoitteet selviävät. Luokan pitäisi olla myös tietoinen turhan työn välttämiseksi siitä, että mitkä ovat ylipäätään yrityksen omistamia IP-osoitteita. Pythonin ipaddress-kirjastolla on mahdollista esimerkiksi tarkistaa helposti, että kuuluuko jokin IP-osoite tiettyyn verkkoblokkiin, joten luomalla listan hallinnoiduista verkoistamme saan järjestelmän tietämään heti liittyykö valitus varmasti meihin vai ei. Aikaleimat ovat lokitiedostoissa myös melko usein sellaisia, että ne noudattavat jotain tiettyä formaattia, joten niitäkin voisi etsiä säännöllisillä lausekkeilla, ja saatua aikaleimaa voisi käyttää valistuneen arvauksen tekemiseen siitä, että ketä

käyttäjää jokin valitus koskee. X-ARF-tyyppisten tietojen koneluku tulee varmasti olemaan helppoa.

Kun kohdennus, valituksen oleellisten osien valinta, viestin käärintä oikean viestipohjan ympärille on suunniteltu, niin käytännössä koko ohjelmalogiikka on samalla tullut suunniteltua tässä vaiheessa ajatustasolla, ja käytännön kehitystyö agile-prosessin mukaisesti puuttuu, ja teoreettisesti tämä kokonaisuus on omasta mielestäni täysin toimiva.

Viikkoraportti

Kuluneen viikon teemojen mukaisesti ajattelin, että tällä kertaa olisi hyvä käsitellä yleisesti automaattista päätöksentekoa turvallisuustoimintakeskuksessa, sivuta nopeasti agile-prosessia sovelluskehityksessä ja tehdä katsaus uhkatietosyötteiden löytämisessä törmättyihin ongelmiin.

Automatisoidun päätöksenteon osalta Trull (2017) esittelee blogikirjoituksessaan erilaisia parhaiksi katsottuja periaatteita, joilla SOC-työtä voidaan automatisoida. Ennen varsinaista kehitystyötä nämä periaatteet on varmasti hyvä katsoa lävitse. Tiivistetysti periaatteet ovat luonteeltaan sellaisia, että niissä viitataan pelkistetyllä tavalla yleiseen tietoturvapoikkeaman reaktioon havainnosta mitigointiin asti. Oleellista antia blogipostauksesta ovat tarkat toimenpiteet siitä, että mihin jonkin tapauksen käsittely pitäisi perustua tarkalleen eli kyse on ehdoista muodostettavasta logiikasta, joka määrittää toimenpiteet, mitkä seuraavat lopulta uhkatiedosta. Toimet ovat esimerkiksi tässä tapauksessa Microsoftilla palomuurisääntöjen luomisia, tikettien luomista ja sähköpostien lähettämistä.

Verrattuna omaan toimintaani ja viitaten aiemmin suunnittelutasolla olevaan Python luokan logiikkaan, joka itsessään vastaisi pelkästään tapauksen käsiteltävyyden päättämisestä, ja ratkaisuehdotuksista näen tässä hyvin paljon samankaltaisuuksia, mutta esimerkiksi Microsoftin tapa vaikuttaa olevan sellainen, että se keskittyy vahvasti automaatioon ilman ihmisten väliintuloa tapauksesta riippuen. Trull (2017) viittaakin siihen, että Microsoftilla on käytössään viisiportainen malli, millä arvioidaan automaatiotason kykyä käsitellä tapauksia, missä ensimmäisellä tasolla automaatio keskittyy hallinnollisten tehtävien, kuten juuri tikettien avauksiin tai sulkuihin, ja sähköpostin lähetykseen, sekä toisella tasolla ylimääräisen hälyn vähentämiseen toistuvista ilmoituksista. Kolmas taso on oleellinen siinä mielessä, siinä tapahtuu suunnittelemaani korrelaatiota ja saatujen tietojen yhdistämistä johonkin asiakkaaseen. Viimeisellä eli viidennellä tasolla puhutaan jo koneoppimisen hyödyntämisestä tapauksien ratkaisemiseksi.

Tämän perusteella pidän ainakin suunnitelmaani järkevänä ottamatta huomioon tulevaa kehitystyötä siitä, että mitä automaattinen käsittelyjärjestelmä tulee lopullisessa muodossaan pitämään sisällään. Kolmas taso on ainakin itseni kehittämisen ja tiimin rasituksen vähentämisen kannalta ainakin suhteessa omiin kykyihin saavutettavissa, ja blogitekstin perusteella kehitystyö on sellaista, että se on menossa oikeaan suuntaan.

Tarkoitus olisi ainakin, että tapauksien käsittely ei olisi täysin automatisoitua, vaan juuri projektin seurauksena syntyy järjestelmä, joka tarjoaa automatisoituja ratkaisuehdotuksia, jotka on helppo hyväksyä, hylätä tai muokata ennen välitystä asiakkaille. Luokka itsessään ei pysty tätä tehtävää hoitamaan vaan tarvitsee mm. integraatiota tiketöintijärjestelmään, mutta koska luokan tarkoitus onkin olla komponentti, jota on helppo käyttää uudelleen, niin tätä ei tarvitse aivan vielä miettiä varsinkaan, kun luokkaa ei ole vielä kehitetty.

Kehitystapa on tässä tapauksessa nyt tyypiltään hyvin agile-tyyppinen, minkä olen valinnut vähän aikaa sitten sovellettavaksi omaan työskentelyyni sellaisilla muutoksilla, että pelkistän mallia ja jätän esimerkiksi sprintit pois kokonaan mallista. Tyypillisesti tämä on se toinen ja alalla eräs tyypillisimmistä tavoista kehittää sovelluksia vesiputousmallin sijasta, joten en aio käsitellä kehitystapaa ainakaan enempää tässä yhteydessä.

Viikon aikana tuli myös tutkittua kevyessä määrin mahdollisia uusia uhkatietojen toimittajia, mutta ongelmaksi itselle tuli tässä tapauksessa erityisesti se, että esitellyt ja tutkimani toimittajat eivät vastanneet tarpeisiini. Syynä olivat juuri esimerkiksi tietojen vähyys, näyttö toimittajan tiedon luotettavuudesta tai hankaluudet päästä järkeväksi määriteltävässä ajassa tiedon saajaksi. Kasarev (2019) kirjoittaa ratkaisuja tähän ongelmaan blogissaan, ja on törmännyt oikeastaan samoihin havaintoihin. Löytämäni lähteet olivat ilmaisia ja avoimia, mutta Kasarev pitää näitä lähteitä yleensä luonteeltaan sellaisina, että niistä tulee usein paljon vääriä positiivisia, tai että löydökset ovat poissa konteksteistaan. Mielestäni tämä havainto on hyvin sidoksissa omiin havaintoihini, että löydetyissä lähteissä oli paljon epäluotettavuutta erityisesti kattavuuden osalta.

Hyvinä lähteinä Kasarev (2019) pitää puolestaan maksullisia lähteitä, mutta niiden haittapuolena on ilmeisesti kallis hinta. Maksullisen lähteen hankinta, joka maksaa merkittävästi on todennäköisesti sellainen asia, joka vaatisi yrityksen sisällä paljon keskustelua, joten jos tämänkaltaista parempaa, mutta maksullista lähdettä haluaisi käyttää niin se pitäisi olla luonteeltaan sellainen, että lähde Kasarevin ehdotuksien mukaisesti räätälöidään tarpeisiin sopivaksi. Yhtenä vinkkinä kirjoituksessa mainitaan myös, että FS-ISAC (Financial Services Information Sharing and Analysis Center) -lähde voisi olla potentiaalinen uhkatiedon lähde, mutta pääsy tietoon vaatisi jäsenyyttä. Käytännössä FS-ISAC on globaali toimija, jolla on rahoitusalaan liittyen jatkuvaa tietoa alaan liittyvistä uhkista.

Pidän FS-ISAC:in tutkimista potentiaalisena lähteenä hyvänä ideana, sillä monissa väärinkäyttötilanteissa on usein rahaan liittyviä motiiveja.

3.8 Seurantaviikko 8

Maanantai 07.03.2022

Heti aamulla sain useita ilmoituksia haavoittuvaisista Gitlab-instansseista verkossamme, joten päätin selvittää tapahtunutta, eli päivän aikataulu määräytyi melko vauhdikkaasti korkeamman prioriteetin tehtävän saapuessa.

Ilmoitukset olivat saapuneet verkon tietoturvaa vapaaehtoisesti edistävän yrityksen toimesta. Ilmoituksissa oli kuitenkin muutamia puutteita. Ilmoituksissa mainittiin haavoittuvat instanssit pelkillä IP-osoitteilla, mikä itsessään olisi normaalitapauksessa ongelmallista, jos kyse olisi vaikkapa haavoittuvasta julkaisujärjestelmästä, sillä usein kyseessä ei ole ainoa julkaisujärjestelmä samalla palvelimella. Tässä tapauksessa pitää kuitenkin pitää mielessä, että todennäköisimmin Gitlab instanssin ylläpitäjä toimii niin, samalla palvelimella ajetaan vain yhtä Gitlab-instanssia, koska Gitlab on kuitenkin yrityskäytössä erittäin merkittävä versionhallintajärjestelmä, jota voi myös ylläpitää itse.

Ainoaksi ihmetyttäväksi asiaksi tästä jäi kuitenkin vain se, että ilmoituksissa ei mainittu mitään erityistä tapaa, että millä haavoittuvuus on tunnistettu tarkalleen. Tätä asiaa kyseltäessä selvisi, että taho identifioi instanssit tutkimalla Gitlabin versionumeroita JavaScript-komentosarjoista suoraan, mitkä ovat saatavilla julkisesti internettiin, ja versionumeroista taho oli päätellyt, että onko instanssi haavoittuvainen vai ei.

Yleisellä tasolla taho vaikutti luotettavalta, ja tietoisena yhtenä Gitlabin parhaista tietoturvaperiaatteista päätin lähettää instanssien omistajille viestejä siitä, että he päivittävät instanssinsa uusimpaan versioon mahdollisimman pian mahdollisen haavoittuvuuden vuoksi, ja että he tarkistavat virtuaalipalvelinten tilan muutenkin mahdollisten hyväksikäyttötilanteiden varalta. Lisäksi kehotin asiakkaita toimimaan niin, että jatkossa he pitäisivät muutenkin instanssit poissa näkyvistä internetistä, eli laittaisivat Gitlab instanssinsa VPN ratkaisun taakse, mikä on normaali ja turvallinen tapa toimia, sillä Gitlab-instanssit ovat yleensä yrityksissä sisäisiä palveluita aina.

Nähtäväksi jää, että kuinka asiakkaat reagoivat tähän kehotukseen, mutta muilta osin työpäivä eteni melko rutiininomaisesti ilman erityisempiä yllätyksiä, ja voisin jatkaa tiistaina aiemmin sovitusta palvelimen ylläpitotehtävästä, ja ajattelin SSO-palvelun käyttöönottoa

eräässä yrityksen tietojärjestelmässä. Tyypillisen IaaS-vastuujaon mukaisesti asiakkaat tulivat kuitenkin informoiduksi mikä on tärkeintä.

Tiistai 08.03.2022

Suunnitelman mukaisesti aloitin tutkimaan dokumentaatiota 3. osapuolen palvelusta SSO-palvelun käyttöönottoon liittyen, jotta selviäisi miten SSO voidaan ottaa käyttöön. Myöhemmälle iltapäivälle olin varannut aikaa aiemmin mainitulle palvelimen päivitystyön tarkistukselle, mikä minut oli pyydetty tekemään aiemmalla seurantaviikolla.

SSO toiminnot ovat, kuten aiemmin sanottu ulkoistettuja ja keskitettyjä kirjautumismenetelmiä, joilla voidaan korvata yksittäisiä ja pirstaleisia käyttäjätietoja eri tietokannoissa, millä kirjaudutaan esimerkiksi yrityksen sisäisisiin ja ulkoisiin palveluihin.

Havaitsin, että erääseen yrityksen käyttämistä tietojärjestelmistä oli tullut tähän liittyen tuki, jonka käyttöönottoa ryhdyin selvittämään henkilöiden kanssa, jotka vastaavat esimerkiksi näiden järjestelmien integroinneista. Selvisi, että integraatio olisi erittäin helppo toteuttaa, sillä kyseessä oli ilmeisesti asiaa kanssa hoitaneiden henkilöiden mukaan ensimmäistä kertaa 3. osapuolen toimesta toimitetut ja erittäin selkeät, ja vaihe vaiheelta kuvatut ohjeet SSO palvelun käyttöönottamiseksi.

Kuten Freescoutin LDAP-kytkennässä, tätä asiaa päätetään myöhemmin edistää, kun SSO:n käyttöönottoon saadaan täytettyä sitä edeltävät vaatimukset, mutta hyödyt ovat vastaavia kuin LDAP:in käytössä, sillä tällä tavalla voidaan esimerkiksi jättää väliin työpaikasta poistuvien työntekijöiden kohdalta käyttäjätunnuksien manuaalinen poistaminen, sillä jos SSO:ssa oleva käyttäjätunnus tuhoutuu, niin sitä ei voi käyttää enää 3. osapuolen järjestelmässä. Suurin vaatimus, mikä pitää täyttää tässä tapauksessa on se, että SSO järjestelmän pääkäyttäjän tulisi luoda varmenteita, mihin tarvitaan yrityksestä henkilöä, jolla on korkeammat käyttöoikeudet kyseisiin järjestelmiin.

Loppuiltapäivästä vastaan tuli sopiva hetki, jolloin minun piti tarkistaa infrastruktuurin ylläpidosta vastaavan tiimin päivitystyön vaikutukset virtuaalipalvelimeen, mihin liittyen en odottanut liittyvän ongelmia. Testauksessa tarvitsi ainoastaan tarkistaa, että tietyt työkalut tulostavat oletettuja arvoja, ja että cron-järjestelmässä toimivat skriptit ovat toimivia. Asiassa ei ilmennyt ongelmia.

Keskiviikko 09.03.2022

Keskiviikkona suunnittelin keskittyväni rutiinitehtäviin, enkä ollut suunnitellut tekeväni mitään erityistä päivän aikana.

Huomiota päivän aikana kiinnitti kuitenkin havaitsemani kuvio, että joku asiakas esiintyy useilla identiteeteillä, ja päättelin henkilön olevan sama henkilö siitä, että palveluumme ilmestyi jatkuvasti kokeilutarjoustamme hyväksikäyttäviä tunnuksia, mitkä loivat huijaussivustoja näyttäviä virtuaalipalvelimia aina tietystä maasta, ja kaikki käyttäjätunnukset muistuttivat toisiaan.

Tarjouksen avulla on mahdollista käyttää hetken aikaa palveluamme ilmaiseksi tietyillä rajoituksilla, mutta henkilö oli keksinyt vihamielisen käyttötavan palveluun, mikä tuli ilmi ulkoisista lähteistä saaduilla ilmoituksilla ja omatoimisella seurannalla. Kun yhden tunnuksen poisti käytöstä, vastaava sivusto aktivoitui jollakin toisella tunnuksella, ja nämä havainnot tehtiin ulkoisista lähteistä saaduilla ilmoituksilla, jotka varmennettiin.

Ratkaisuna tässä tapauksessa oli käyttäjämassasta kuvioon täsmäävien tuoreiden, ja tietystä maasta saapuneiden uusien asiakkaiden manuaalinen validointi, ja tarkkaavaisuuden lisääminen kyseisestä maasta saapuvien asiakkaiden kanssa. Tunnistuksessa auttoi se, että maa ei ole mitenkään merkittävä yrityksen liikevaihdon kannalta, joten etsittäviä tahoja oli vähän.

Tapaus tuli ratkaistua toistaiseksi, ja todennäköisesti taho ymmärtää tulleensa havaituksi sillä taho ei pääse enää tietenkään kirjautumaan palveluun, joten asiaan pitää suhtautua sellaisella varauksella, että taho luovuttaa haitallisen toimintansa harjoittamisen tai muuttaa sitä. Pidän todennäköisempänä, että taho luovuttaa ja siirtyy yrittämään huijauksia jollakin toisella palveluntarjoajalla, missä yksinkertainen huijaus menee helpommin läpi.

Torstai 10.03.2022

Torstaina suunnittelin muiden työtehtävieni ohessa tarkistavani ensimmäisiä SOC-keskeisiin tehtäviin hakeneiden henkilöiden taustoja, sillä keskiviikon ja torstain välillä henkilöstöhallinto oli saanut viimeisteltyä, ja julkaistua SOC-tehtäviin liittyvän työpaikkailmoituksen.

Keskiviikon mukaisesti olin jo saanut koulutusta rekrytointijärjestelmän käyttämiseksi ja kandidaattien kelpoisuuksien selvittämiseksi, joten omalta osaltani sovitun mukaisesti keskityin ainoastaan äänestämään ja kommentoimaan henkilöiden sopivuutta mahdolliseen tulevaan haastatteluun liittyen.

Suoraan hylkäävillä kommenteilla äänestin esimerkiksi sellaisia tahoja, jotka eivät olleet lukeneet työpaikkailmoituksen sisältöä eli tässä esimerkissä tahot olivat lähettäneet sellaisia hakemuksia, jotka olivat ristiriidassa työpaikkailmoituksen kanssa. Joillakin hakijoilla löytyi selvää kiinnostusta alaa kohtaa, ja sopivaa työkokemusta tai työkokemusta, joten äänestin heitä jatkoon. Henkilöt, joilla taas oli epäselvästi tuotuna kokemusta esille hakemuksissa ovat taas sellaisia, joiden osalta tarvitsee kysyä jatkokysymyksiä siitä, että miten juuri heidän osaamisensa on relevanttia työpaikan kannalta.

Rutiinitehtävien osalta aiemmin viikolla mainitut haavoittuvaiset Gitlab instanssit saivat tapauksena loppupäivästä jatkoa, sillä niistä raportoinut taho suostui jakamaan kanssamme heidän käyttämänsä Python-skriptin, ja havainnointimenetelmän haavoittuvaisten instanssien varmentamiseksi, minkä otin säilytykseen, mikäli sitä tarvitsee käyttää jonkin instanssin tilan tarkastamiseksi jälkikäteen. Päivän osalta tavoitteet tulivat täytetyiksi ja jään odottamaan lisää kandidaatteja erityisesti rekrytointiprosessiin liittyen hakuajan ollessa auki.

Perjantai 11.03.2022

Viimeisenä päivänä viikkoraportoinnissa kalenterimerkintöjen perusteella aikaa päivän aikana tulisi viemään tiimien välisen toiminnan koordinointi asiakkaiden rekisteröintiprosessin helpottamiseksi kaikkien asiaan liittyvien sisäisten sidosryhmien osalta, ja ulkoisesti asiakkaiden osalta.

Asiakkaat tulevat siis väistämättä myyntikanavasta riippumatta sisään palvelun asiakkaiksi käyttämällä yrityksen julkisia internet-sivuja, mutta tällä hetkellä käytetyissä tiedonkeruulomakkeissa oli joitakin puutteita. Puutteista oli kerätty yrityksen sisällä palautetta eri tiimeiltä, jotta mahdollisimman monen tiimin näkökanta tulee huomioiduksi, jotta lomakkeita ei kehitettäisi esimerkiksi vain harvojen sisäisten tiimien liiketoimintatavoitteiden näkökulmista.

Palautteen perusteella oli syntynyt projekti, johon oli tehty ns. mockup eli kuvilla havainnollistettu prosessi tavoista, miten asiakkaaksi ryhtyminen jatkossa toimisi vaihe vaiheelta. Käytännössä itseni kohdalta huomasin vain, että prosessissa oli unohdettu kokonaan yritysasiakkaiden näkökulma, eli yritysasiakkailla ei ollut keinoja syöttää yritystietojaan kuin manuaalisesti vasta jälkikäteen asiakkaaksi tulemisen jälkeen, mikä voi aiheuttaa hämmennystä ja maksuvirheitä, jos asiakkaalla on yrityksen nimen sisältävä yrityskortti. Tein asiasta kehitysehdotuksen ennen kuin suunnitteluvaiheista edetään kehitysvaiheisiin.

Päivän lopuksi tarkastin vielä muutaman SOC-tiimiin hakeneen henkilön taustoja rekrytointiportaalista, ja olin valmis lopettamaan tämän viimeisen seurantaviikon kirjoittamisen,

kun sain varmistettua, että SOC tiimin tavoitteet on huomioitu asiakashankintaprosesseissa.

Viikkoraportti

Viimeisen seurantaviikon vaihtelevat tehtävät koskivat pääosin erilaisten tietoturvatapahtumien käsittelyä, eli esimerkiksi maanantain ja keskiviikon tapauksissa korostui erilaisten uhkien torjunnat melko lailla samankaltaisella mitigointiprosessilla. Tiistain osalta SSO:n käyttöönoton tarkastelu muistuttaa mielestäni niin paljon aiempaa LDAP-projektia, joten en koe juurikaan tarpeelliseksi kyseisen aiheen uudelleenkäsittelyn kuin siltä osin, että olen aiemmin perusteellisesti todennut tämänkaltaisten ratkaisujen olevan hyödyllisiä myös muilta kuin tietoturvakontrollien osilta.

Loppuviikosta pääsin vaikuttamaan rekrytointipäätöksiin julkaistun työpaikkailmoituksen ansiosta ja toimimaan UI/UX asioihin liittyvässä projektissa, johon liittyy suoraan useiden sisäisten sidosryhmien toimintaa, jotka vaikuttavat erityisesti uusiin asiakkaisiin, että niiden hankintaan erityisesti koska kyseessä on juuri uusasiakashankintaan liittyvä projekti.

Maanantain ja keskiviikon tapaukset saadun uhkatiedon valossa ovat selkeästi erilaisia, mutta katsoisin, että mielestäni näitä, ja kaikkia muitakin tapauksia voidaan mallintaa yleismaailmallisesti aina samanlaisella reagointimallilla, jossa uhka torjutaan valmiustilasta siirtymällä palautumiseen, kunhan se on välissä todettu todeksi tai epätodeksi.

Maanantain osalta tutkiessani tapoja tiedottaa asiakkaita haavoittuvuuksista löysin vahvistusta julkaistun tutkimuksen kautta oman työni tueksi, että oma työtapani olisi alan käytäntöjen mukainen, mutta mielenkiintoisempana idean tutkimuksesta siitä, että agile-tyyppisesti suunnitteluprosessin käynnistäjänä jonkin prosessin muuttamiseksi kerättäisiin myös palautetta ilmoitetuilta asiakkailta siitä, että asiakkaille lähetettäisiin lista kysymyksiä, joilla evaluoitaisiin sitä, että kuinka tyydyttävästi tahoa informoitiin tapauksesta (Cetin, O. Ganan, C. Korczynski, M & van Eeten, M, s.a.). Yrityksessä ei ole aiemmin kerätty tästä aiheesta tietoa, joten ylipäätään prosessin kehittämiseksi täytyisi saada aikaan palautetta, jotta selviäisi mitä kehitettävää tiedotuslinjassa mahdollisesti on. Tutkimus saattaa olla yleisellä tasolla olla muutenkin hyödyllistä luetuttaa yrityksessä läpi, sillä sen tutkimusongelmana on erityisesti suurien tietoturvatapahtumien raportoinnin tehostaminen ja erityisesti haavoittuvuustiedotteiden lähettäminen on tapa, millä tutkimusta lähestytään.

Keskiviikon osalta taas torjuin ja reagoin samaa identiteettiä petollisesti käyttävään tahoon, mutta verratakseni omaa toimintaa kirjalliseen materiaaliin löysin Stripen (s.a.) tekemän dokumentin haitallista toimintaa harjoittavan käyttäjän tunnistamisesta, ja havaitsin,

että tässä erittäin selvässä tapauksessa on paljon sellaisia tuntomerkkejä, jotka löytyivät myös Stripen dokumentaatiosta. Stripe on eräs suurimmista maksunvälityspalveluiden tarjoajista, joten heillä on todennäköisesti paljon kokemusta tämänkaltaisista tapauksista, eli Stripe kehottaa olemaan tarkka samankaltaisuuksien kanssa tässä tapauksessa epäilyttävältä vaikuttavien käyttäjätunnuksien välillä. Samankaltaisuudet koskivat tässä tapauksessa juuri maata, mihin haitalliset tilit oli merkitty sijaitsemaan ja käyttäjätunnuksien nimissä oli samankaltaisuutta myös. Tämän perusteella voi myös sanoa, että koska tuntomerkit olivat näin selkeitä niin hyökkääjä ei ollut käyttänyt juurikaan vaivaa hyökkäyksen toteuttamiseksi.

Torstain osalta ajattelin tarkastella aiemmin mainittua rekrytointiprosessia ja vertailin sitä tässä tapauksessa Betterteamin (s.a.) ehdotuksiin parhaista periaatteista SOC analyytikoiden palkkaamisesta. Betterteamin mallin mukaisesti rekrytointi on jo edennyt kolmanteen vaiheeseen tässä tapauksessa eli hakemuksien arviointivaiheeseen, mutta aiempia kohtia arvioidakseni näyttää siltä, että aiemmat vaiheet on toteutettu oikein eli työpaikkamainos on julkaistu monikanavaisesti ja yrityksen tavat myydä työpaikka tarjotuilla eduilla tulevat mainoksesta ilmi.

Betterteamin (s.a.) Kolmas osio keskittyykin lähinnä kaikkien hakemuksien käsittelyyn ennen kuin hakijoille lähetetään viestejä mitä tällä hetkellä tehdään, jotta selviäisi ketkä ovat tulevien vaiheiden kandidaatteja, mistä voidaan jatkaa mahdollisiin hakijoille kohdistettuihin lisäkysymyksiin, että mahdollisiin taustoituksiin. Koska en erityisen suuresti kuitenkaan tähän prosessiin osallistui kuin niin, että annan mielipiteitä koskien erityisesti hakijoiden teknistä taustaa, ja koska näyttää siltä, että rekrytointiprosessi toimii sen perusteella, että se noudattaa verrokkina muiden yritysten hyväksi esittelemiä käytäntöjä. Mielipidettä vahvistaa tietysti aiemmat henkilöstöhallinnon opintoni.

Perjantain, eli viimeisen tarkastelupäivän osalta jäljellä jää tutkittavaksi asiakashankintaan liittyvä projekti. Lähestyn tätä näkökulmaa niin, että koska perinteikkäästi käytettävyys ja turvallisuus ovat olleet usein sellaisia asioita, joiden ei ole nähty toimivan käsi kädessä, niin koen parhaaksi tavaksi omien mielipiteiden eli turvallisuusnäkökulman tuomisen käyttöliittymäsuunnittelukeskusteluun muiden tekemän kirjallisuuden kautta, jota ajattelin jakaa eteenpäin projektiin osallistuvien henkilöiden kesken eli Maassin (2012) mukaisesti turvallisen, mutta käyttäjäystävällisen käyttäjäkokemuksen luonti käyttöliittymän kautta alkaa uusasiakashankintaan suhteutettuna siitä, että käyttäjiltä kysytään progressiivisesti, mutta informoivasti käyttäjädataa, joka on tärkeää käyttäjätunnuksen luomiseksi, mutta myös käyttäjän tunnistamiseksi kaikkien voimassa olevien lakien mukaisesti. Tietojen keräämisessä tulisi priorisoida sitä, että kun erilaisia varmennuksia kuten sähköpostivahvistuksia tehdään kerroksittain, niin tästä huolimatta näiden vaiheiden tulisi edetä saumattomasti ja

nopeasti. Koska projekti on vasta aluillaan ja suunnitteluvaiheessa, niin tämä on mielestäni erittäin hyvä hetki tuoda siihen mukaan erilaisia näkökulmia, jotta ne voidaan huomioida toteutusvaiheissa.

4 Pohdinta ja päätelmät

Koska olin jo opinnäytetyön aloitusvaiheessa ollut pidemmän aikaa työelämässä, niin päätin, että opinnäytetyön suurimmat hyödyt ja tavoitteet voisivat liittyä oman ammatillisen kehittymiseni tukemiseen. Päiväkirjamuotoinen työ tuli valittua työn tyypiksi käytännössä juuri siitä lähtökohdasta, että siinä tulee väistämättä luotua sellaisia merkintöjä, joihin voin tarvittaessa palata tarkemmin työn ohessa, ja näin olen oikeastaan jo tehnyt seurantaviikkojen ohessa, sekä koko päiväkirjaraportointia koskevan osuuden valmistumisen jälkeen.

Päiväkirjatyyppinen opinnäytetyö vaikutti myös sopivalta oman aikatauluni, työelämän ja vapaa-ajan yhteensovittamiseksi, ja arvioin että pystyn työni luonteen vuoksi myös tuomaan teoriatietoa mukaan tämäntyyppiseen opinnäytetyöhön. Koska päiväkirjatyyppistä työtä tehdään sen pisimmältä, eli raportointiviikkojen osuudelta juuri viikoittain, niin pidin selvänä sitä, että työ tulee kuin väistämättä lopulta valmiiksi.

Ensimmäisten seurantaviikkojen aikana oli itselleni vaikeaa löytää rutiinia tämänkaltaisen työn raportointiin, mutta mielestäni pääsin kiinni parempaan raportointitapaan mukaan saamani palautteen avulla, ja tätä tukee lisäksi työstä päiväkirjaraportoinnin ja pohdinnan välissä, eli opponointivaiheessa saamani palaute. Rutiinin ja oikean aikataulutuksen löytämisen kirjoittamista varten oli suurimpia syitä varmasti siksi, että miksi työ eteni lopulta palautusvalmiiksi kohti arvosteluvaihetta. Katsoisin myös, että päiväkirjatyyppinen raportointi auttoi hyvin luomaan parempaa suurta kuvaa itselle koko työni mittakaavasta, ja tekemään siitä aiempaa organisoidumpaa, koska juuri laajojen merkintöjen avulla minun oli mahdollista reflektoida juuri aiempia tekemisiäni nykyhetken toimintatapoihini.

Kehittymistä kaikkien 8 seurantaviikon osalta on tapahtunut käytännössä niillä alueilla, jotka ovat olleet minulle ennen tuntemattomampia ja olen pystynyt tunnistamaan sellaisia alueita helposti päiväkirjaraportoinnin ansiosta, ja tämä koskee eniten sovelluskehitystehtäviä, että integrointitehtäviä, jotka eivät menneisyydessä ole olleet ennen niin merkittävä osa työni kuvaa, kuin ne nykypäivänä ovat olleet. Niihin keskittyminen on mielestäni myös tärkeää siksi, että todennäköisesti sovelluskehitystyö SOC-prosessien tukena tulee olemaan yhä merkittävämmässä roolissa, ja olen työn ohella havainnut, että näinkin lyhyellä tarkastelujaksolla kysyntää tämäntyyppiselle toiminnalle on ollut työpaikalla.

Konkreettisia esimerkkejä kehityksestä ovat olleet mielestäni sellaiset tilanteet juuri, joissa olen tunnistanut selviä ongelmia, esittänyt ja implementoinut niihin jonkin uuden tuotantovalmiin ratkaisun työn aikana, ja siirtänyt aiempia toimintamalleja syrjään, mutta tuonut tilalle juuri uusia toimintamalleja, jotka sitten noudattavat kuitenkin alalla hyväksi todettuja periaatteita. Näistä voisi nostaa esimerkkeinä esimerkiksi keskitettyyn käyttäjähallintaan

siirtymiset LDAP-järjestelmän avulla tiketöintijärjestelmässä, mitä käsitellään eri tavoilla seurantaviikkojen lomassa siirtymällä tarpeen syntymisestä, suunnittelusta; että testauksesta implementointiin.

Rajapintakehitys on mielestäni myös esimerkki uudesta tavasta toimia, sillä siinä uuden REST-tyyppisen sovelluksen tarpeita tarkastellaan aiemman vastaavan sovelluksen iteraation perusteella, jotta sovellusta olisi mahdollista ylipäätään suunnitella oikein. Rajapintakehityksestä mielestäni on kannattavaa nostaa esiin myös se, että olen esimerkiksi halunnut suunnitella järjestelmän elinkaarta tulevaisuuteen toteuttamalla tuen erilaisille autentikointimenetelmille, ja tekemällä sellaisia teknologiavalintoja, joilla järjestelmän ylläpito onnistuu melko vaivattomasti.

Koska sovelluskehitystyö ja integraatioiden luominen eri tietojärjestelmien välille on ollut luonteeltaan sellaista, että se liittyy SOC-työn erilaisiin prosesseihin, niin se on ollut mielekästä itselleni, mutta päiväkirjatyön ansioista kehittyminen juuri ohjelmistokehityksessä on ollut tavoitteellisempaa. Katsoisin, että päiväkirjatyössä tämä johtuu juuri tarkastelujaksoista, jotka seuraavat jokaista seurantaviikkoa, missä on tärkeää yrittää löytää vaihtoehtoisia ratkaisumalleja kohdattuihin ongelmiin, mikä käytännössä mielestäni tarkoittaa sitä, että jos opinnäytetyöstä haluaa tehdä ohjeiden ja arviointikriteerien mukaisesti sellaisen, että tavoitteena on ainakin hyvä arvosana, niin perehtyminen alan teoriatietoon, että ammattilaisten kirjoituksiin olisi vähintään pakollista, ja niin mielestäni olen tehnytkin.

Vaikka käytetty teoriatieto esimerkiksi olisi ollut melko pitkälti linjassa omien näkemyksieni kanssa siitä, että mitkä ovat hyviä työskentelyperiaatteita alalla, ja kuten on ollut, niin huomasin että tämäntyyppisestäkin tiedosta on ollut itselleni hyötyä siksi, että se tuo kuitenkin itsevarmuutta ja luottamusta omiin työtapoihin varsinkin, kun työtehtävien johto perustuu enemmänkin tavoitteellisuuteen, ja vapauteen käyttää erilaisia keinoja saavuttaa tavoitteet.

Seurantaviikkojen aikana vähemmän kehittymistä puolestaan tapahtui eniten rutinoituneissa prosesseissa, mitkä koskivat haitallisen toiminnan lopettamista loppuasiakkaiden kohdalta, sillä olen tehnyt tämäntyyppisiä tehtäviä huomattavasti pidempään, kuin aiemmin mainitsemiani ja kuvailemiani sovelluskehitystehtäviä. Katsoisin, että kuitenkin teoriatiedon valossa käyttämäni prosessit ovat melko universaaleja, eikä niissä ole juurikaan parannettavaa toisinkuin luomissani ja ylläpitämissäni tietojärjestelmissä, joiden tehtävä on tukea kaikkia prosessejani.

Tiivistetysti jatkokehitys tähän työhön liittyen tuleekin todennäköisimmin saamaan jatkoa juuri ohjelmistokehityspuolella, sillä käytännössä en näe, että mikään muu asia voisi

paremmin skaalata ylöspäin SOC-työn tehokkuutta. Todennäköisesti tämä pääsee tapahtumaan parhaiten tämän vuoden loppupuoliskolla, kun SOC-tiimiin tulee ilmeisimmin uusi rekrytointi aiempien seurantaviikkojen mukaisesti, mikä mahdollistaa sen, että minun on mahdollista käyttää enemmän aikaa kehityskohteiden toteuttamiseen. Konkreettisin kohde, mikä on mainittu tässä työssä on todennäköisimmin siis aloittamani uuden rajapinnan luomistyö, johon SOC-työtä tukevat prosessit sitten perustuvat, ja jonka ympärille työssä käytettävä käyttöliittymä sitten rakennetaan.

Katsoisin samalla, että minun kannattaisi perehtyä vielä tarkemmin jo valmiiksi organisaatiossa tarjolla oleviin tietolähteisiin, ja tutkia että voisinko käyttää tietolähteitä vielä paremmin oman työni tehostamiseksi.

Tiivistäen voisi sanoa työn lopetukseksi, että lopputuloksena monivaiheisesta työstä syntyi sen raportointiin liittyvistä alkuvaikeuksista huolimatta yllättävän paljon kerrottavaa varsinkin sen jälkeen, kun johonkin aihepiiriin pääsi syventymään kunnolla ja varsinkin niissä tapauksissa, joissa oli mahdollista vertailla juuri omaa työtapaa kohden vaihtoehtoisia toteutustapoja.

Lähteet

Accessnow. Digital Security Helpline Community Documentation - Force Tor Browser to use specific exit nodes. 2019. URL: https://communitydocs.accessnow.org/147-Tor_force_exit_nodes.html. Luettu: 14.02.2022.

Au-Yeung, J. Donovan, R. Best practices for REST API design. StackOverflow. 2020. URL: https://stackoverflow.blog/2020/03/02/best-practices-for-rest-api-design/. Luettu: 20.02.2022.

Arias, D. Adding Salt to Hashing: A Better Way to Store Passwords. 2021. auth0. URL: https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/. Luettu: 26.02.2022.

ARIN. IP Geolocation: The Good, The Bad, & The Frustrating. 2018. URL: https://www.arin.net/blog/2018/06/11/ip-geolocation-the-good-the-bad-the-frustrating/. Luettu: 14.02.2022.

Betterteam. How to Hire Information Security Analysts. s.a. URL: https://www.betterteam.com/how-to-hire-information-security-analysts. Luettu: 12.03.2022.

Cannata, G. Welcome to ldap3's documentation. 2020. URL: https://ldap3.readthedocs.io/en/latest/. Luettu: 17.01.2022.

Catalin, C. Second data wiper attack hits Ukraine computer networks. 2022. The Record. URL: https://therecord.media/second-data-wiper-attack-hits-ukraine-computer-networks/. Luettu: 25.02.2022.

Cetin, O. Ganan, C. Korczynski, M. van Eeten, M. Make Notifications Great Again: Learning How to Notify in the Age of Large-Scale Vulnerability Scanning. s.a. URL: https://mkorczynski.com/WEIS2017Cetin.pdf. Luettu: 12.03.2022.

Copes, F. JWT authentication: Best practices and when to use it. 2021. LogRocket. URL: https://blog.logrocket.com/jwt-authentication-best-practices. Luettu: 26.02.2022.

Debian. Release Notes for Debian 11 (bullseye), 64-bit PC - Chapter 4. Upgrades from Debian 10 (buster). s.a. URL: https://www.debian.org/releases/stable/amd64/release-notes/ch-upgrading.en.html. Luettu: 20.02.2022.

Federal Trade Commission. How To Recognize and Avoid Phishing Scams. s.a. URL: https://www.consumer.ftc.gov/articles/how-recognize-and-avoid-phishing-scams. Luettu: 04.02.2022.

HAYS. Why company values matter. s.a. URL: https://www.hays.net.nz/employer-insights/management-issues/why-company-values-matter. Luettu: 26.02.2022.

HR Partner. 5 Employee Induction Best Practices from HR Experts. s.a. URL: https://blog.hrpartner.io/employee-induction/. Luettu: 14.01.2022.

Janus, J., Chojnacki, M., Szwajkiewicz, W. Pros and Cons of Using Third-Party Software in Your App Development. 2020. URL: https://www.thedroidsonroids.com/blog/third-party-software-pros-and-cons. Luettu: 04.02.2022.

Kasarev, A. Evaluating threat intelligence sources. Kaspersky. 2019. https://www.kaspersky.com/blog/evaluating-threat-intelligence/26952/. Luettu: 06.03.2022.

Kearl, B. 5 Things We Can All Learn from Valve's Exceptional Employee Handbook. BambooHR. 2016. URL: https://www.bamboohr.com/blog/5-things-we-can-all-learn-from-valves-exceptional-employee-handbook/. Luettu: 26.02.2022.

Klensin, J. Simple Mail Transfer Protocol. 2008. URL: https://data-tracker.ietf.org/doc/html/rfc5321. Luettu: 14.01.2022.

Levin, G. 4 Most Used REST API Authentication Methods. 2019. RestCase. URL: https://blog.restcase.com/4-most-used-rest-api-authentication-methods/. Luettu: 26.02.2022.

Lind, M. Should I use virtualenv or Docker containers with Python? 2019. URL: https://coderbook.com/@marcus/should-i-use-virtualenv-or-docker-containers-with-python/. Luettu: 20.02.2022.

Maass, M. Security vs. Design: Standing at Odds?. UXMMag. 2012. URL: https://uxmag.com/articles/security-vs-design-standing-at-odds. Luettu: 12.03.2022.

McKenzie, C. Take advantage of these 5 benefits of server-side rendering. 2020. TheServerSide. URL: https://www.theserverside.com/feature/Take-advantage-of-these-5-benefits-of-server-side-rendering. Luettu: 20.01.2022.

Microsoft. External Authentication Services with ASP.NET Web API (C#). 2020. URL: https://docs.microsoft.com/en-us/aspnet/web-api/overview/security/external-authentication-services. Luettu 04.02.2022.

MITRE. ATT&CK - Geofencing, Technique T1581. s.a. URL: https://attack.mitre.org/techniques/T1581/. Luettu: 04.02.2022.

Muniz, J., Lakhani, A., Santos, O., Frost, M. The Modern Security Operations Center. 2021. Addison-Wesley Professional. USA. E-kirja. Luettu 8.12.2021.

Nathans, D. Designing and Building Security Operations Center. 2014. Elsevier Science & Technology Books. Waltham. E-kirja. Luettu: 8.12.2021.

OpenStreetMap. Copyright and License. s.a. URL: https://www.openstreetmap.org/copyright. Luettu: 13.01.2022.

Oracle. Database Security Guide - Administering Authentication. s.a. URL: https://docs.oracle.com/cd/B19306_01/network.102/b14266/admnauth.htm#i1006459. Luettu 04.02.2022.

Pair. Email Spam: How to Identify It and Protect Yourself. 2020. https://www.pair.com/support/kb/email-spam-how-to-identify-it-and-protect-yourself/. Luettu: 14.01.2022.

Red Hat. Red Hat JBoss Operations Network - Extended Example: memberOf and LDAP Configuration. s.a. URL: https://access.redhat.com/documentation/en-us/red_hat_jboss_operations_network/3.1/html/admin_initial_setup_inventory_groups_and_users/ex-ldap-authz. Luettu: 20.01.2022.

Real Python. Python Virtual Environments: A Primer. Real Python. s.a. URL: Python Virtual Environments: A Primerhttps://realpython.com/python-virtual-environments-a-primer/. Luettu: 20.02.2022.

Reitz, K. The Hitchhiker's Gudie to Python - Pipenv & Virtual Environments. s.a. Real Python. URL: https://docs.python-guide.org/dev/virtualenvs/. Luettu 20.02.2022.

Tschabitscher, H. Email Headers Can Tell You About the Origin of Spam. 2021. LifeWire. URL: https://www.lifewire.com/email-headers-spam-1166360. Luettu: 14.01.2022.

SOCRadar. The Ultimate List of Free and Open-source Threat Intelligence Feeds. 2021. https://socradar.io/the-ultimate-list-of-free-and-open-source-threat-intelligence-feeds/. Luettu: 01.03.2022.

SHRM. How to Develop an Employee Handbook. s.a. URL: https://www.shrm.org/resourcesandtools/tools-and-samples/how-to-guides/pages/developemployeehandbook.aspx. Luettu: 26.02.2022.

State Service of Special Communication and Information Protection of Ukraine. Another cyberattack on government websites and banks. 2022. URL: https://cip.gov.ua/en/news/chergova-kiberataka-na-saiti-derzhavnikh-organiv-ta-banki. Luettu: 25.02.2022.

Stripe. Identifying potential fraud. s.a. URL: https://stripe.com/docs/disputes/prevention/identifying-fraud. Luettu: 12.03.2022.

The Internet Society. Lightweight Directory Access Protocol (LDAP): The Protocol. 2006. URL: https://datatracker.ietf.org/doc/html/rfc4511. Luettu: 05.01.2021.

Trull, J. Top 5 best practices to automate security operations. Microsoft. 2017. URL: https://www.microsoft.com/security/blog/2017/08/03/top-5-best-practices-to-automate-security-operations/. Luettu: 06.03.2022.

Valve. Handbook for new employees. 2012. URL: https://steamcdn-a.akamaihd.net/apps/valve/Valve_NewEmployeeHandbook.pdf. Luettu: 26.02.2022.

Young, S. How much theory should you learn for practical skills? 2016. URL: https://www.scotthyoung.com/blog/2016/06/15/why-learn-theory/. Luettu: 04.02.2022.