Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Luka Kavčič

Sistem za spremljanje varnostnih dogodkov z uporabo odprtokodnih orodij

diplomsko delo

visokošolski strokovni študijski program
prve stopnje
računalništvo in informatika

Mentor: izr. prof. dr. Mojca Ciglarič
Somentor: dr. Dušan Gabrijelčič

Ljubljana, 2018

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Preučite delovanje sodobnih sistemov za upravljanje varnostnih dogodkov in informacij (SIEM) in izpostavite pomanjkljivosti odprtokodnih v primerjavi s plačljivimi sistemi. Zasnujte učinkovitejši sistem SIEM, ki bo temeljil na odprtokodnih orodjih tako, ki naj bi se po funkcionlanosti približal plačljivim sistemom. Sistem nato implementirajte in ovrednotite njegovo delovanje.

Zahvaljujem se mentorici izr. prof. dr. Mojci Ciglarič, somentorju dr. Dušanu Gabrijelčič ter Laboratoriju za odprte sisteme in mreže za strokovne nasvete pri izdelavi diplomske naloge. Hvala tudi staršema in prijateljem za spodbudo in podporo v času študija.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
API	Application programming interface	Aplikacijski programski vmesnik
CIA	Confidentiality, integrity, availability	Zaupnost, integriteta, razpolozljivost
IDS	Intrusion detection system	Sistem za zaznavanje vdorov
IPS	Intrusion prevention system	Sistem za preprečevanje vdorov
IP	Internet protocol	Internetni protokol
REST	Representational state transfer	Predstavitveni prenos stanja
SSH	Secure Shell	varna povezava
SIEM	Security Information and Event Management	Upravljanje varnostnih informacij in dogodkov
USB	Universal serial bus	Univerzalno serijsko vodilo

Povzetek

Naslov: Sistem za spremljanje varnostnih dogodkov z uporabo odprtokodnih orodij

Avtor: Luka Kavčič

Informacijsko-komunikacijski sistemi so pomemben sestavni del v večini današnjih podejtij. Zaradi vse večjega števila naprav, ki so vključene v informacijsko-komunikacijske sisteme postaja vzdrževanje in varovanje le tega vse težje. V diplomski nalogi sem predstavil sisteme za upravljanje varnostnih informacij in dogodkov (SIEM), njihovo delovanje ter kako se razlikujejo od upravljalcev beleženj ter sistemov IDS/IPS. Preveril sem takšne že obstoječe plačljive ter brezplačne odprtokodne sisteme, ki so na tržišču. Nato sem še sam implementiral sistem SIEM z uporabo zgolj odprtokodnih orodij ter ga ovrednotil s pomočjo primerov uporabe.

Ključne besede: Informacijska varnost, Upravljanje varnostnih informacij in dogodkov (SIEM).

Abstract

Title: Security Information and Event Management Using Open Source Tools

Author: Luka Kavčič

Information communication systems are an important component in most of today's enterprises. Due to the increasing number of devices that are connected in information communication systems, maintenance and security are becoming increasingly difficult. In my graduation thesis, I introduced systems for managing security information and events (SIEM), how they work and how they differ from log managers and IDS/IPS systems. I've checked existing commercial, free and open-source SIEM systems on the market. Then I implemented the SIEM system using only open-source components and evaluated it through use cases.

Keywords: Information security, Security information and event management (SIEM).

Poglavje 1

Uvod

Informacijsko-komunikacijski sistemi so pomemben sestavni del v večini današnjih podjetij. V ta sistem so vključeni računalniki, tiskalniki, spletni strežniki, požarni zidovi in usmerjevalniki. Zaradi vedno večjega števila naprav v informacijsko-komunikacijskih sistemih in pomembnost, da ta deluje ves čas, postaja vzdrževanje le tega vse težje. V diplomski nalogi bomo raziskali sisteme, ki so administratorjem v pomoč pri vzdrževanju in zagotavljanju varnosti informacijsko-komunikacijskega sistema. Najprej bomo predstavili kako se določa varnost informacijsko-komunikacijskih sistemov, nato bomo obrazložili vlogo takšnih sistemov za vzdrževanje in varnost, ter kako se razlikujejo od drugih, podobnih orodij, ki prav tako varujejo informacijsko-komunikacijske sisteme. Nato bomo pojasnili princip delovanja takšnih sistemov. Pregledali bomo že obstoječe plačljive ter odprtokodne sisteme ter jih med seboj primerjali. Za konec bomo implementirali takšen sistem, ki je zgrajen z uporabo zgolj odprtokodnih orodij, ter ga še ovrednotili.

Poglavje 2

Pregled področja

Danes so informacijsko-komunikacijski sistemi vse večji, kompleksnejši in težji za vzdrževanje. Z razvojem tehnologije so postali informacijsko-komunikacijski sistemi vse bolj izpostavljeni informacijskim grožnjam ter vdorom. Napadalci danes napadajo podjetja z namenom kraje informacij. V primeru takšnih groženj je hitro ukrepanje ključno. Če želimo pravočasno ukrepati na varnostno grožnjo, jo moramo najprej zaznati in jo tudi razumeti. Za zaznavo varnostnih groženj moramo napravo spremljati v realnem času. To lahko počnemo s prebiranjem beleženj. V velikih informacijsko-komunikacijskih sistemih, ki vključujejo po več tisoč računalnikov, spletnih strežnikov ipd. , je to praktično nemogoče. V ta namen so bili razviti sistemi, ki spremljajo stanje informacijsko-komunikacijskih sistemov v realnem času. Na ta način administratorje obveščajo, pomagajo ter usmerjajo k učinkovitemu obvarovanju ter vzdrževanju informacijsko-komunikacijskih sistemov.

2.1 Dogodki in incidenti

Dogodek [20] je vsak opažen pojav v informacijskem sistemu. Dogodki vključujejo uporabniški dostop do spletne datoteke, strežniška zahteva za spletno stran, pošiljanje e-pošte in blokiranje promeat na mreži z požarnimi

zidovi. Nezaželeni dogodki so dogodki, ki pustijo negativne posledice kot so sistemske zrušitve, poplave paketov, nepooblaščena uporaba sistemskih privilegijev, nepooblaščen dostop do občutljivih podatkov in zagon zlonamerne programske opreme.

Incident v informacijski varnosti je kršitev ali neposredna grožnja kršitve varnostne politike, katera ima lahko vpliv na delovanje informacijsko-komunikacijskega sistema.

2.2 Informacijska varnost

Temeljni cilji v informacijski varnosti so vključeni v trojici CIA [23]. Kratica je sestavljena iz treh besed, katere so primarna načela varnosti. To so zaupnost (angl. *confidentiality*), celovitost (angl. *integrity*) in razpoložljivost (angl. *availability*). Glede na ta načela se navadno ocenjuje varnost informacijsko-komunikacijskega sistema.

Zaupnost je zaščita podatkov v informacijsko-komunikacijskih sistemih. V večini današnjih informacijsko-komunikacijskih sistemih so podatki glavni vir katerega je potrebno zavarovati. Postavljeni morajo biti ukrepi, ki varujejo in zagotovijo, da občutljivih podatkov ne dosežejo nepooblaščeni. Eden takšnih ukrepov je enkripcija podatkov.

Celovitost pomeni zagotavljanje konsistentnosti, pravilnosti in zanesljivosti podatkov. Spreminjanje, dodajanje ter brisanje podatkov predstavlja lahko veliko grožnjo za pravilno delovanje informacijsko-komunikacijskega sistema.

Razpoložljivost je stalna pripravljenost sistema, da izvaja svoje funkcije za svoje uporabnike.

Slika 2.1: Trojica CIA.

2.3 Zanka opazovanja, usmerjanja, odločanja in ukrepanja

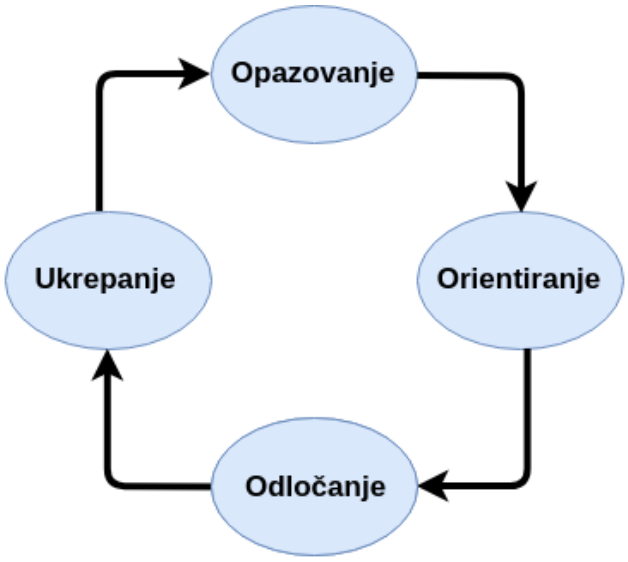
Modeli se pogosto uporabljajo za strukturiranje informacij, tako da jih je mogoče analizirati in glede na njih tudi ukrepati. Eden takšnih modelov je zanka OODA [4, 21] , ki se uporablja za izvajanje hitrih, časovno občutljivih odločitvah v informacijski varnosti. Naredil ga je John Boyd, za vojsko Združenih držav Amerike in sicer želel je ugotoviti, kako najhitreje ukrepati v kriznih situacijah. Zanka OODA je odločitveni cikel oziroma model, ki je sestavljen iz štirih korakov. Ti koraki so opazovanje, usmerjanje, odločanje in ukrepanje oziroma »observe, orient, decide, act«.

Opazovanje: Cilj tega koraka je zbiranje informacij. Zbira se vse, kar bi lahko bilo uporabno. V primeru informacijske grožnje oziroma vdora se v tem koraku zbira beleženja, spremlja se delovanje naprav ter zbira druge informacije za prepoznavo napadalca.

Orientiranje: V koraku orientiranja se zbrane informacije iz prejšnjega koraka postavi v kontekst z že znanimi informacijami. To vključuje pretelke izkušnje, pričakovanja in modele. V primeri vdora v informacijsko-komunikacijski sistem, v orientacijo štejemo informacije zbrane iz beleženj združene z poznavanjem informacijsko-komunikacijskega sistema ter prej identificirane podatke kot so specifčni naslovi IP in imena procesov.

Odločanje: V koraku odločanja so informacije zbrane in postavljene v kontekst. Ta korak vključuje iskanje različnih možnosti ukrepanja, dokler ni končni ukrep izbran. V primeru vdora se v tem koraku odloča ali še vedno opazuje napadalca, ignorira ali kako ukrepa.

Ukrepanje: V zadnjem koraku je izvajanje prej izbranega ukrepa. Tudi, če je ukrep neuspešen je naslednji korak zopet opazovanje.



Slika 2.2: Zanka OODA.

2.4 Sistem SIEM

Security Information and Event Management oziroma SIEM je kompleksna skupina orodij, ki omogočajo celosten pregled nad informacijsko-komunikacijskim sistemom v realnem času. Po modelu zanke OODA administratorjem informacijsko-komunikacijskih sistemov pomagajo pri opazovanju ter orientiranju. Samo odločanje in ukrepanje pa prepustijo administratorjem. V osnovi SIEM omogoča naslednje:

- zbiranje beleženj in dogodkov v informacijsko-komunikacijskem sistemu,
- vizualizacija stanja informacijsko-komunikacisjkega sistema v realnem času imenovana tudi pregledna plošča ali »ptičja perspektiva«,
- normalizacija podatkov oziroma prevajanje računalniško generiranih podatkov v ljudem berljiv način,
- prilagodljivost v pomenu, da omogoča vključitev različnih naprav iz informacijsko-komunikacijskega sistema,
- poročanje in opozarjanje administratorjev na incidente ter grožnje.

2.4.1 Razvoj SIEM sistemov

Sistemi SIEM so se prvič pojavili na trgu leta 1997. Nastali so z namenom, da administratorjem informacijsko-komunikacijskih sistemov pomagajo pri zmanjševanju šuma pri množičnem zbiranju beleženj. Najprej so delovali le kot zbirališča vseh beleženj iz različnih virov, kot so: sistemi IDS, IPS ter spletni strežniki in požarni zidovi. Tehnologija je bila še veliko bolj robustna in samo zbiranje beleženj in pregledovanje le teh je bilo zelo počasno. S časom so sistemi začeli filtrirati beleženja in izločati le najpomembnejše podatke, ki so jih administratorji hitreje pregledali in so na ta način v krizni situaciji hitreje reagirali. Nato so iz zbranih podatkov, preko prej definiranih pravil obveščali administratorje o incidentih ali grožnjah, ki so se pojavili na informacijsk-komunikacisjkem sistemu, kar je še pospešilo njihov odziv. S

pregledno ploščo o informacijsko-komunikacijskem sistemu ter obveščanjem o incidentih ter varnnostnih grožnjah v realnem času je postal sistem SIEM eden pomembnejših varnostnih sistemov v vsakem večjem, pa tudi manjšem podjetju ali ustanovi [22, 5].

2.4.2 Kako se SIEM razlikuje od sistemov IPS in IDS?

IDS (*Intrusion detection system*) in IPS (*Intrusion prevention system*) sta orodji, ki ves čas aktivno spremljata promet na omrežju ter iščeta neobičajne vzorce ali sumljivo obnašanje. Ko kaj zaznata, sprožita obvestilo o varnostni grožnji. Glavna razlika med njima je, da sistem IDS samo obvesti administratorje o grožnji medtem, ko sistem IPS aktivno spreminja omejitve omrežja z npr. uporabo požarnih zidov.

Sistem SIEM lahko deluje vzporedno z orodji IDS/IPS, tako da zbira njihove in od ostalih naprav informacije v informacijsko-komunikacijskem sistemu ter administratorjem ponudi obširnejši pogled nad varnostjo in stanjem le-tega. Prav tako je za SIEM pomembno, da je pasiven sistem, ki zgolj obvešča in je brez administratorjev povsem neuporaben, saj sam ne preprečuje groženj ali napak v informacijsko-komunikacijskem sistemu. Torej zgolj opazuje in usmerja administratorje.

2.4.3 Kako se SIEM razlikuje od navadnih upravljalcev beleženj?

Navadni upravljalci beleženj zgolj zbirajo vsa beleženja naprav v informacijsko-komunikacijskem sistemu na skupno točko. Hranijo jih kot celotna sporočila ter ponujajo administratorjem pogled v trenutno stanje naprave, ter kaj se je z njo dogajalo v preteklosti. Sistemi SIEM pa prejeta beleženja predhodno še normalizirajo v enotno standardno obliko. S pomočjo analize in korelacije med zbranimi podatki iščejo incidente in varnostne grožnje v informacijsko-komunikacijskem sistemu ter obveščajo o tem administratorja v realnem času. S tem nudijo boljši pogled nad celotnim delovanjem informacijsko-

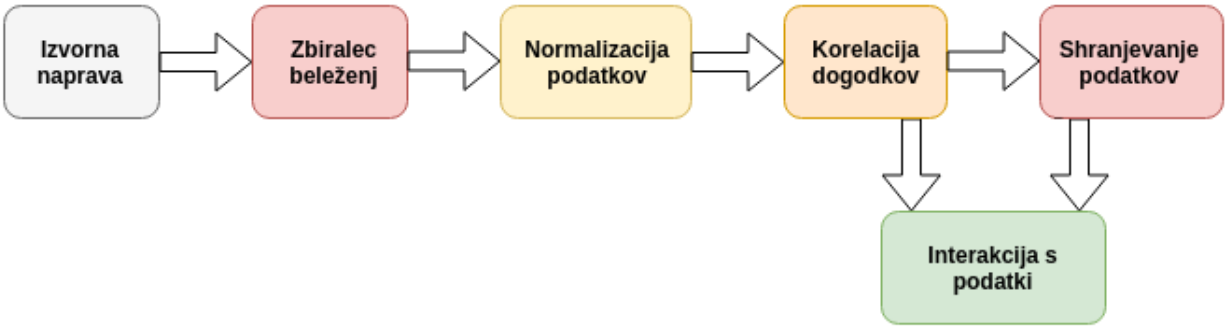
komunikacijskega sistema [5].

Funkcija	**SIEM**	**Upravljalec beleženj**
Zbiranje beleženj	Zbiranje beleženj glede varnosti.	Zbiranje vseh beleženj.
Ohranjanje beleženj	Ohranja le razčlenjene in normalizirane podatke iz beleženj.	Ohranja razčlenjene podatke iz beleženj ter celotna sporočila. Ohranja jih daljša časovna obdobja.
Poročanje	Poročanje je osredotočeno na informacijsko varnost, ki je v realnem času.	Ne poroča.
Analiza	Korelacija podatkov, ocenjevanje ogroženosti ter določanje prioritet incidentov in varnostnih groženj.	Analiza vseh zapisov ter označevanje le-teh.
Obveščanje	Napredno obveščanje glede varnosti.	Enostavno obveščanje.
Druge značilnosti	Upravljanje incidentov ter druge varnostne analize iz podatkov.	Velika razširljivost pri zbiranju in iskanju po beleženjih.

2.4.4 Princip deloveanja SIEM sistema

SIEM lahko primerjamo s kompleksno skupino orodjih sestavljeno iz več delov. Vsak izpolnjuje svojo nalogo, vendar morajo delati skupaj, saj v nasprotnem primeru celoten sistem odpove. Posamezen člen lahko deluje samostojno od drugih, vendar brez skupnega delovanja sistem SIEM ne more pravilno delovati. Princip delovanja sistema SIEM je zgrajen na osnovi verige iz šestih členov, kot prikazuje slika 2.3. Ti posamezni členi so izvorna naprava,

zbiralec beleženj, normalizacija podatkov, korelacija dogodkov, shranjevanje podatkov ter interakcija s podatki. Vsak člen te verige predstavlja točno določeno funkcijo [19, 3].



Slika 2.3: Veriga delovanja sistema SIEM.

Izvorna naprava

Prvi člen verige SIEM je izvorna naprava, ki sistemu SIEM daje informacije. Izvorna naprava je lahko aplikacija, strežnik, usmerjevalnik ali kakšna druga naprava, iz katere želimo pobirati beleženja in jih pošiljati sistemu SIEM. Sama izvorna naprava ni del sistema SIEM, vendar brez beleženj je sistem SIEM neuporaben. Teoretično le z zbiranjem beleženj vseh naprav dobimo točen ter celovit pogled nad informacijsko-komunikacijskim sistemom. Praktično je to nesmiselno, zaradi preobremenitve kasnejših členov verige SIEM. Zato se administratorji sami odločijo, katere naprave so kritične in ključnega pomena za delovanje informacijsko-komunikacijskega sistema.

Zbiralec beleženj

Drugi člen verige SIEM je zbiralec oziroma upravljalec beleženj, torej prenos in zbiranje beleženj izvornih naprav. V principu sta dva načina zbiranja beleženj iz izvornih naprav, vsak s svojimi prednostmi ter slabostmi.

- **Zbiranje beleženj s potiskanjem:** Izvorna naprava pošilja beleženja sistemu SIEM. Prednost takšnega zbiranja beleženj je enostavnost vzpostavitve. Pri takšnem zbiranju pa nastane ranljivost sistema SIEM, saj je občutljiv na poplavljanje s podatki in s tem je mogoče upočasniti njegovo delovanje.

- **Zbiranje beleženj s pobiranjem:** sistem SIEM sam pobira beleženja iz izvorne naprave. Na ta način se izognemo poplavljanju sistema SIEM. Slabost je, da ne deluje v realnem času, saj je lahko izvorna naprava ogrožena vendar sistem SIEM še ni pobral beleženj iz nje in ni zaznal varnostne grožnje.

Normalizacija podatkov

Tretji člen verige SIEM je normalizacija podatkov. Ker zbrana beleženja prihajajo iz različnih naprav, ki so lahko od različnih proizvajalcev, je naslednji korak pretvarjanje beleženj v enotno standardno obliko, ki bo uporabna za sistem SIEM. Prav tako nam nekatera sporočila ni potrebno shranjevati, če menimo, da so neuporabna. Takšno filtriranje lahko izvedemo na dva načina:

- **Filtriranje pri ponoru:** To je najbolj praktična in enostavna izvedba. Vse delo, kaj se filtrira in kaj ne, prepustimo sistemu SIEM. Slabost takšnega filtriranja je nastanek ozkega grla, saj se vse delo prestavi na zbirno točko. Prav tako ustvarimo več nepotrebnega prometa na omrežju.

- **Filtriranje pri izvorni napravi:** Najboljša in najtežja izvedba, saj na ta način pošiljamo le najpomembnejše informacije po omrežju in posledično zmanjšamo nepotreben promet in se izognemo ozkemu grlu. Vendar vse naprave ne omogočajo filtriranja beleženj oziroma je potrebna uporaba tretje programske opreme.

Administratorji se sami odločijo, kako bodo filtrirali beleženja glede na njihovo arhitekturo informacijsko-komunikacijskega sistema in največkrat iz-

berejo mešanico obeh načinov. S filtriranjem beleženj pri izvornih napravah se lahko izognemo nastanku ozkega grla, vendar je implementacija in vzdrževanje le-tega težje od filtriranja pri ponoru [18].

Korelacija dogodkov

Iz zbranih ter normaliziranih podatkov iščemo povezave med informacijami, ki nam povedo ali se je zgodil incident ali varnostna grožnja, ki jo želimo zaznati. Kot primer je neuspela prijava uporabnika na spletni strežnik. Samo en takšen podatek o dogodku nima velikega pomena, vendar večje število neuspelih prijav v spletni strežnik v krajšem časovnem obdobju, pa spominja na napad. Zaznavanje takšnih varnostnih groženj poteka preko napisanih pravil. Ta imajo navadno sistemi SIEM že implementirana in omogočajo tudi dodajanje svojih. Pri zaznavanju incidentov in varnostnih groženj se uporablja pristop »*Known knowns, known unknowns and unknown unknowns.*« Torej nekatere incidente ter grožnje poznamo in imamo že implementirana navodila oziroma jih sami napišemo. Zavedati se tudi moramo, da nimamo pravil za nekatere incidente, kot so razni manj znani napadi na omrežja in jih naš sistem posledično tudi ne bo zaznal. Pravila morajo biti smiselna in ne zavajajoča, saj je namen le-teh obveščanje administratorjev informacijsko-komunikacisjkega sistema o incidentih ter varnostnih grožnjah [11].

Shranjevanje podatkov

Vse podatke, ki jih sistem SIEM zbere ter normalizira, je potrebno shraniti v podatkovno bazo, do katerih v kasnejših členih lahko dostopamo.

Interakcija s podatki

Zadnji člen verige SIEM je interakcija s podatki ter prikaz le-teh. Vsak sistem SIEM ima tudi uporabniški vmesnik, ki je lahko izveden kot spletni vmesnik ali aplikacija. Ti služijo kot pregledne plošče oziroma tako imenovane »ptičje perspektive«, ki prikazujejo ter obveščajo administratorje o

trenutnem stanju informacijsko-komunikacijskega sistema. Pregledne plošče so tudi prilagodljive in si administratorji sami izberejo kaj bodo prikazovale, glede na to kaj sami potrebujejo. Prav tako se s pomočjo njih upravlja sam sistem SIEM.

2.4.5 Center za varnostne operacije

Sistem SIEM le spremlja in zbira podatke, odkriva incidente in varnostne grožnje v informacijsko-komunikacijskem sistemu ter obvešča o njih brez kakšnega ukrepanja. Torej opravlja samo prva dva koraka zanke OODA, to sta opazovanje in orientiranje. Za doseganje informacijske varnosti imajo podjetja usposobljeno osebje, ki se ob zaznavi incidenta ali varnostne grožnje s strani sistema SIEM lahko pravilno odločajo in tudi ukrepajo. To so tako imenovane ekipe za odzivanje na incidente v računalniški varnosti ali »*Computer Security Incident Response Team*« [20].

Poglavje 3

Analiza področja

Namen tega poglavja je pregled nekaterih že obstoječih sistemov SIEM, ki so danes na tržišču. Nekateri so plačljivi, kot so IBM QRadar, Splunk ter Intel Security, zdaj pod drugim imenom McAfee. Nekateri so tudi odprtokodni, kot je Alien Vault OSSIM. Vodilna na tržišču sta IBM QRadar in Splunk, vendar vsak od njiju ponuja različni rešitvi.

3.1 QRadar

QRadar je sistem SIEM, ki ga je razvilo podjetje IBM [13]. Na tržišče je vstopil leta 2001 in po Gartnerjevem poročilu iz leta 2016 tudi postal vodilni sistem SIEM [17]. QRadar je mogoče namestiti kot strojno, programsko ali kot oblačno storitev, ki je nameščena na IBM-ovih strežnikih. Ta vse-v-enem sistem vključuje vsa potrebna orodja za doseganje informacijske varnosti in spremljanje dogodkov v osnovnem produktu. Prav tako sam poskrbi za potrebe po shranjevanju dogodkov, spremljanje omrežnega prometa in obveščanja. QRadar vključuje naslednje funkcije:

- Spremljanje dogodkov glede varnosti,
- spremljanje omrežnega prometa,
- iskalec ranljivosti v informacijsko-komunikacijskem sistemu,

- analiza podatkov,
- korelacija podatkov,
- hevristično prepoznavanje in prioriteta groženj,
- generiranje poročil.

QRadar je eden od enostavnejših sistemov SIEM za vzpostavitev na informacijsko-komunikacijskih sistemih. Prav tako omogoča natančnejše nastavitve pri iskanju ranljivosti, incidentov ter varnostnih groženj. S tem lažje določimo nepotrebno in dolgotrajno zbiranje napačnih-pozitivnih incidentov ter varnostnih groženj. Posebnost QRadar sistema SIEM je, da uporablja avtomatiziran postopek za iskanje novih virov beleženj in poročil o prometu na omrežjih ter naprednejša pravila za iskanje dogodkov, ki pridobljene podatke v velikih količinah strnejo v manjše, lažje obvladljive količine. Vključuje tudi spletno trgovino, iz katere si lahko prenesemo in namestimo razne vtičnike in dodatna pravila. Slabost le-tega je, da so nekateri še dodatno plačljivi. Popolna avtomatizacija, ki sicer ni del sistema SIEM, je prav tako dodatno plačljiva in ni del osnovnega produkta. Osnovni produkt je primeren za manjša in srednje velika podjetja, z dodatki pa tudi za večja podjetja.

3.2 Splunk

Splunk sistem SIEM je razvilo istoimensko ameriško podjetje [25]. Na tržišče je vstopil leta 2012 in po Gartnerjevem poročilu iz leta 2016, se nahaja na drugem mestu med sistemi SIEM [17]. Splunk prihaja v dveh različicah, Splunk Enterprise, ki je osnovni produkt. Ta ponuja osnovno zbiranje beleženj in poročil o prometu na omrežjih, indeksiranje, iskanje po le-teh in generiranje poročil. Podjetja, ki uporabljajo ta produkt si tako sama zgradijo korelacijska pravila za prepoznavanje incidentov in varnostnih groženj, da dosežejo funkcionalnosti sistemov SIEM. Drugi produkt je Splunk App for Enterprise Security, ki deluje na osnovnem produktu in vključuje vnaprej zgrajena ko-

relacijska pravila, obveščanje, pregledne plošče, preglede nad incidenti in iskanje ranljivosti v informacijsko-komunikacijskem sistemu. Dodatni vtičniki in pravila so na voljo na spletni trgovini. Žal ne omogoča dodatnih avtomatizacij, kot jih QRadar. Namestiti ga je mogoče kot programsko ali oblačno storitev. Dostopen je tudi kot 60 dnevna brezplačna različica.

3.3 Alien Vault OSSIM

Alien Vault OSSIM [1] je, kot njegovo ime namiguje, odprtokodni sistem SIEM in je tako brezplačen za uporabo. Brezplačna različica ima omejitve, ki vključujejo zmogljivost, shrambo podatkov in podporo, katere so del plačljive različice. Odprtokodna različica OSSIM sistema vključuje procesiranje in normalizacija beleženj, ki mu jih posredujemo ter korelacijo dogodkov za zaznavo incidentov in varnostnih groženj. Združuje zbiralec beleženj in korelacijo dogodkov iz več odprtokodnih programov, da doseže funkcionalnosti sistema SIEM. Nekateri od programov, ki jih vključuje so FProbe, Munin, Nagios, Snort in TCPTrack. Odprtokodna različica je primerna zgolj za podjetja z manjšimi informacijskimi sistemi. Za večje informacijsko-komunikacijske sisteme morajo podjetja nadgraditi Alien Vault OSSIM na plačljivo različico. Namestiti ga je mogoče kot programsko ali strojno opremo ter kot oblačno storitev.

3.4 SIEMonster

SIEMonster [24] je brezplačni odprtokodni sistem SIEM. Vključuje tudi plačljivo SIEMonster Premium različico z dodatnimi funkcionalnostmi. Zgrajen je s pomočjo Elastic Stack-a, kateri služi za centralizacijo beleženj, normalizacijo, shrambo in vizualizacijo podatkov. Osnova je zgrajena na šestih strežnikih, ki delujejo na virtualnem računalniku z operacisjkim sistemom Linux. Vsak od strežnikov ima svojo vlogo.

- Proteus: za centralizacijo in normalizacijo beleženj.
- Capricorn: za korelacijo in vizualizacijo podatkov.
- Kraken in Tiamat: gruča podatkovnih baz.
- Ikuturso: za spremljanje omrežnih dogodkov Bro/Tardis.
- Hydra: za pošiljanje beleženj iz izvornih naprav.

Ne omogoča avtomatizacije kot QRadar ampak zgolj obveščanje o incidentih in grožnjah na informacijskem sistemu. Primeren je tako za manjša kot večja podjetja. Plačljiva različica vključuje sprotne posodobitve, tehnično podporo in do 7 let shranjevanja podatkov v primeru nesreč. Namestimo ga lahko kot programsko ali oblačno storitev.

Zaradi različnih informacijsko-komunikacijskih sistemov v podjetjih ali ustanovah ne obstaja sistem SIEM, ki prinaša najboljšo rešitev. Sistem SIEM, ki je primeren za en informacijsko-komunikacijski sistem, je morda nepopoln v drugem. Odprtokodni sistemi SIEM so vsestranski in zmogljivi, vendar potrebujejo veliko strokovnega znanja in časa za pravilno vzpostavitev. Plačljivi sistemi SIEM zaradi njihove enostavne namestive, vključene osnovne konfiguracije filtriranja, korelacije in vizualizacije podatkov prevladujejo.

Poglavje 4

Načrt izdelave

4.1 Zahteve

Radi bi implementirali sistem SIEM zgolj z uporabo odprtokodnih orodij. Namenjen je administratorjem in sicer kot pomoč pri spremljanju in varnosti informacijsko-komunikacijskih sistemov. Sistem mora biti prilagodljiv v primeru razširjanja informacijsko-komunikacisjkega sistema in neodvisen od velikosti ter naprav v le-tem. Z zbiranjem ter shranjevanjem podatkov o trenutnem stanju naprav mora, preko določenih pravil, prepoznati nekatere incidente in varnostne grožnje v informacijsko-komunikacijskem sistemu ter obvestiti administratorja. Zbrane podatke si lahko administratorji podrobneje ogledajo ter tudi poljubno vizualizirajo. Omogočati mora dodajanje novih naprav za spremljanje ter novih pravil za prepoznavanje incidentov in varnostnih groženj. Delovati mora po pravilih varnosti CIA, torej uporaba in spreminjanje podatkov je zagotovljena le osebam z določenimi pravicami. Celotno spremljanje informacijsko-komunikacijskega sistema ter obveščanje administratorjev mora biti avtomatizirano ter delovati v realnem času.

4.2 Primeri uporabe

Sistem SIEM sistem je namenjen administratorjem informacijsko-komunikacijskega sistema. V nadaljevanju je opisanih nekaj primerov uporabe, ki služijo tudi kot cilji katere želimo doseči.

4.2.1 Dostop SSH

Napadalec želi dostopati preko SSH do spletnega strežnika. Napad izvaja s poizkušanjem različnih gesel za poljubnega uporabnika. Sistem SIEM spremlja stanje tega strežnika v informacijsko-komunikacijskem sistemu. Vsako beleženje o prijavah SSH shrani, neglede na to ali je bila prijava uspešna ali neuspešna. SIEM zazna veliko število neuspelih prijav SSH na strežnik v krajšem časovnem obdobju. Ustvari poročilo o varnostni grožnji, si ga shrani ter obvesti administratorja informacijsko-komunikacijskega sistema. Administrator s pomočjo poročila izve kateri spletni strežnik je ogrožen ter iz katerega naslova je napaden. Prav tako si lahko podrobneje ogleda posamezna beleženja strežnika. Glede na poročilo SIEM in shranjenih beleženj se administrator odloči kako bo ukrepal, če je sploh potrebno.

4.2.2 Naprave USB

Zaradi varovanja podatkov ter informacijsko-komunikacijskega sistema, je v podjetju na določenih računalnikih prepovedana uporaba ključev USB. Novi delavec je priključil ključ USB v takšen računalnik. Sistem SIEM spremlja stanje naprave in zazna priključitev ključa USB. Sestavi poročilo, ga shrani ter obvesti administratorja informacijsko-komunikacijskega sistema o incidentu. Administrator se glede na poročilo odloči, kako ukrepati.

4.2.3 Spletni strežnik

Spletni strežnik, ki je ključen za delovanje podjetja, postane nedostopen. Sistem SIEM spremlja stanje tega spletnega strežnika in zazna njegovo nedosegljivost. Sestavi poročilo, si ga shrani in obvesti administratorja informacijsko-komunikacijskega sistema o incidentu. Administrator pregleda poročilo ter sam pregleda pretekla beleženja spletnega strežnika, katere je sistem SIEM shranil. S temi podatki lahko sklepa, kaj se je zgodilo s spletnim strežnikom ter odpravi napako.

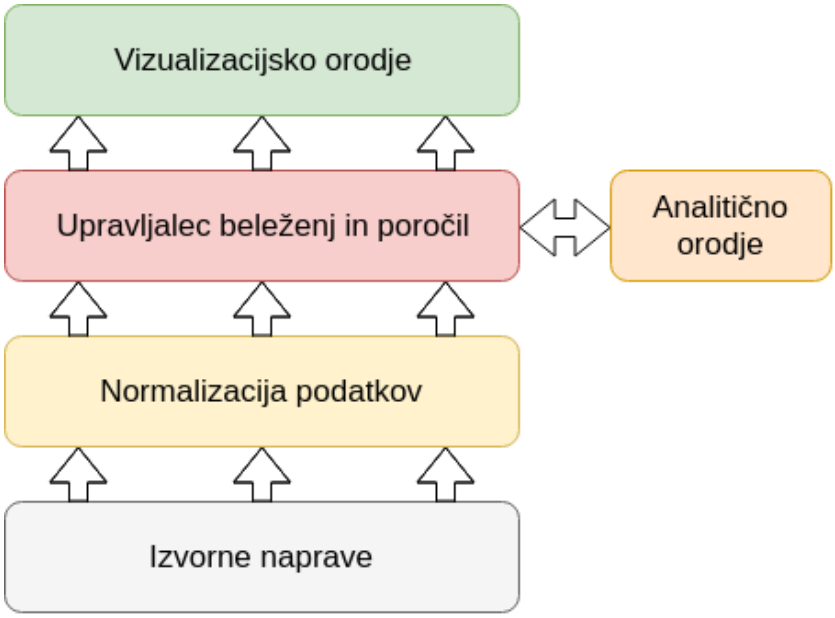
4.2.4 Izvajanje ukaza sudo

Na spletnem strežniku nekdo izvaja ukaze z pravicami *root*. Sistem SIEM to zazna, ustvari poročilo in obvesti administratorja informacijsko-komunikacijskega sistema. Administrator iz poročila razbere kateri ukazi so bili izvedeni in sam odloči ali je to varnostna grožnja.

4.3 Načrt

Sistem SIEM bomo sestavili po principu že obstoječih sistemov, ki so na trgu. Vseboval bo naslednje komponente:

- Izvorne naprave,
- normalizacija podatkov,
- upravljalec beleženj in poročil,
- analitično orodje oziroma pravila za zaznavanje incidentov in varnostnih groženj,
- vizualizacijsko orodje.



Slika 4.1: Arhitektura sistema SIEM.

Na izvornih napravah bomo zbirali beleženja in druge podatke ter jih posredovali sistemu SIEM. Ta jih bo ob prejetju normaliziral v obliko za lažjo obdelavo ter jih shranil v podatkovno bazo, ki bo služila kot upravljalec beleženj in poročil. Analitično orodje s pomočjo pravil, spremlja dogajanje v informacijsko-komunikacijskem sistemu v realnem času. V primeru, da s pomočjo pravil, zazna incident ali varnostno grožnjo, ustvari poročilo ter si ga shrani v podatkovno bazo. Zbrane podatke in poročila si lahko administratorji v realnem času ogledajo s pomočjo vizualizacijskega orodja.

4.4 Podatki in generiranje poročil

Zgolj zbiranje podatkov in beleženj iz informacijsko-komunikacijskega sistema ni dovolj, saj na ta način otežimo zaznavo incidentov in varnostnih groženj. Potrebna je normalizacija le teh. Pri generiranju poročil o dogodkih in anomalijah na informacijsko-komunikacijskem sistemu je pomembno, da so vključene le najpomembnejše informacije, ki administratorje orientirajo in

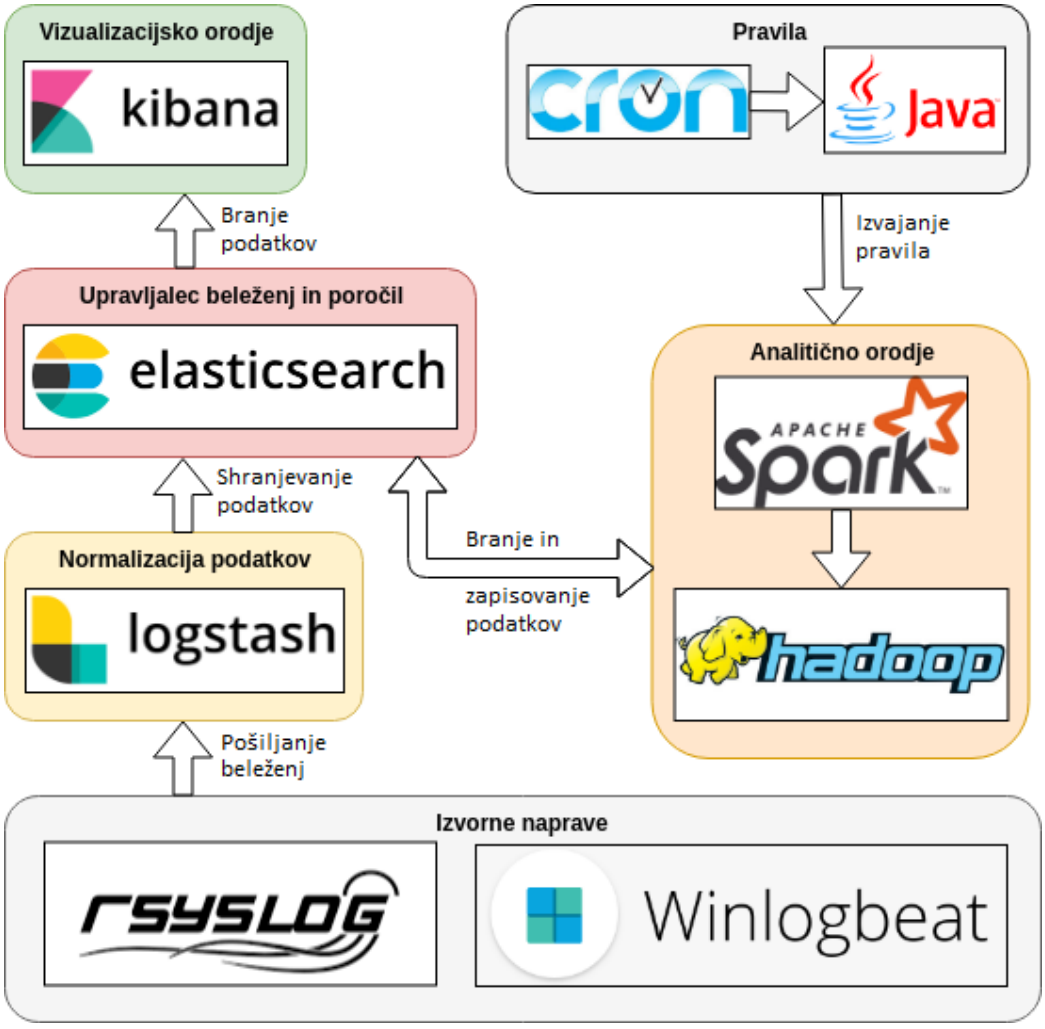
jim pomagajo pri odločitvi kako ukrepati. Pomembno je, da jih ne preobremenimo s preveliko količino podatkov. Poročila morajo odgovoriti na štiri vprašanja, s katerimi si administratorji pomagajo. To so:

- Kdaj se je incident ali varnostna grožnja zgodila?
- Kje se je zgodil?
- Kdo je povzročitelj? (uporabnik ali naprava)
- Kakšen je tip incidenta ali varnostne grožnje?

Poglavje 5

Izvedba

Sistem SIEM zgradimo na operacijskem sistemu Linux in sicer na distribuciji Ubuntu 18.04 LTS [26]. Ta operacijski sistem izberemo zaradi njegove prilagodljivosti. Za osnovo sistema SIEM uporabimo Elastic Stack [7], kot ga uporablja SIEMonster. Ta vključuje Logstash [16] in Elasticsearch [8] za obdelavo in shranjevanje podatkov ter Kibano [15] kot vizualizacijsko orodje. Pravila za zaznavanje incidentov in varnostnih groženj so napisana v programskem jeziku Java [14]. Program Cron [6] skrbi za periodično poganjanje pravil. Analitično orodje, s pomočjo katerega se izvajajo pravila, je sestavljeno iz Apache Sparka [2] ter gruče računalnikov Hadoop [12].



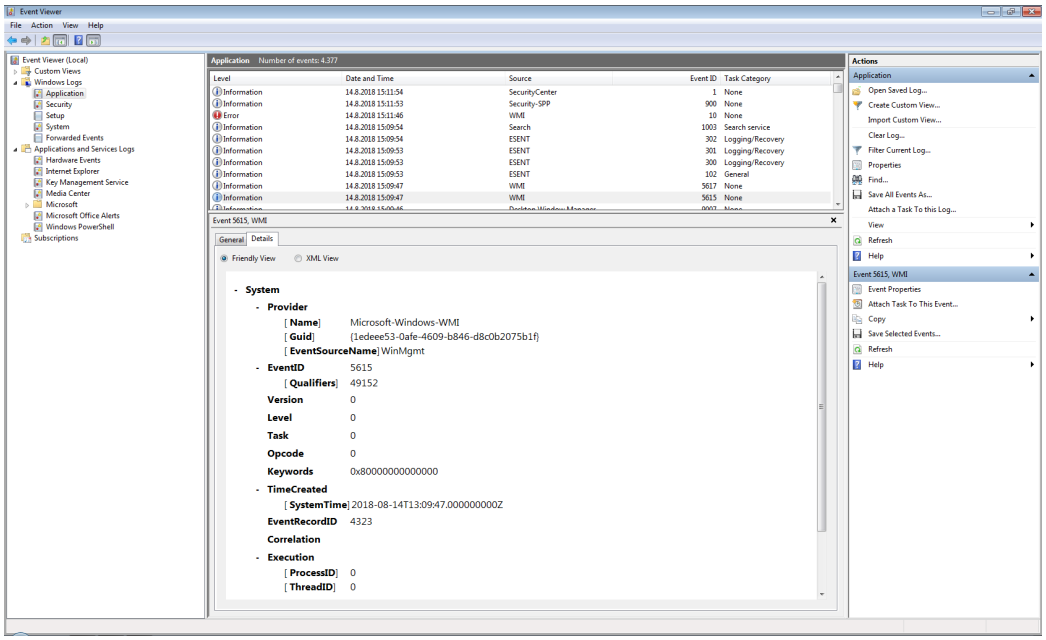
Slika 5.1: Komunikacijski diagram med uporabljenimi orodji.

5.1 Izvorne naprave

Za izvorne naprave smo uporabili spletni strežnik in osebne računalnike z operacijskimi sistemi Linux Ubuntu 18.04 LTS, Windows 7 [28] ter Windows 10 [27]. Vse naprave so bile ves čas vključene v informacijsko-komunikacijski sistem.

5.1.1 Operacijski sistem Windows 7 in 10

Pri operacijskem sistemu Windows lahko spremljamo beleženja s programom Event Viewer. Aplikacije in operacijski sistem tu zapisuje strojne in programske dogodke, kateri administratorjem pomagajo pri odpravljanju težav. Dogodke, ki jih Windows beleži so namestitve programske opreme, varnostna beleženja, sistemska opravila pri zagonu ter napake. Vsak zapisan dogodek vsebuje datum in čas dogodka, prijavljenega uporabnika, ime naprave, identifikacijska številka dogodka, izvorni program, ki je sprožil dogodek ter tip dogodka z dodatnim opisom.



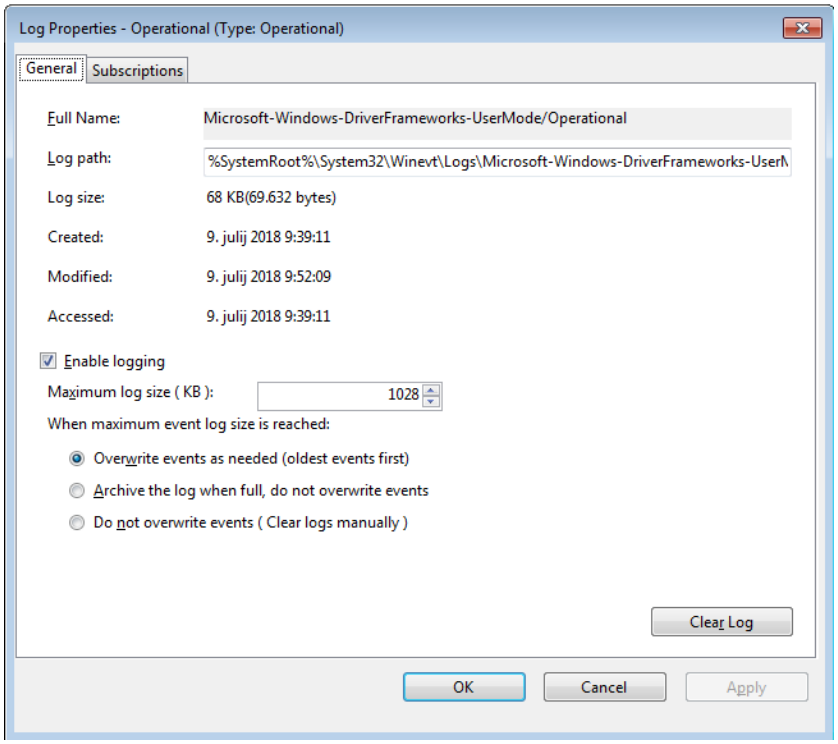
Slika 5.2: Program Event Viewer za spremljanje beleženj na operacijskih sistemih Windows.

Dogodki, ki se zapišejo v Event Viewer ob priključitvi naprave USB, se shranijo pod »Applications and Services Logs/Microsoft/Windows/DriverFrameworks-UserMode/Operational«. To beleženje na samodejnih nastavitvah ni vključeno. Vključimo ga tako, da kliknemo nanj z desnim klikom in izberemo »Properties« ter nato obkljukamo »Enable Logging«, kot pri-

kazuje slika 5.3.

Spodnja tabela prikazuje katere dogodke sproži priključitev naprave USB ter njihov opis.

Identifikacijske številke dogodkov	**Opis**
2003, 2004, 2006, 2010	Naložitev gonilnikov za upravljanje priključene naprave.
2100, 2101, 2105, 2016	Plug-and-Play ali upravljanje z napajanje priključene naprave.



Slika 5.3: Vključevanje dodatnega beleženja.

Za pošiljanje beležen iz operacijskega sistema Windows smo uporabili orodje Winlogbeat. Glede na nastavitve, Winlogbeat prebere enega ali več

dogodkov, jih filtrira ter pošlje na podani naslov in vrata. Branje in pošiljanje beleženj Winlogbeat počne v realnem času. Nastavitve spreminjamo v datoteki »*winlogbeat.yml*«. Primer datoteke z nastavitvami, katero smo uporabili, je vidna na izpisu 5.1. Poleg filtriranja beleženj pri izvorni napravi, Winlogbeat sam normalizira podatke, ki jih posreduje, kar še dodatno zmanjša obremenitev kasnejših členov sistema SIEM.

Izpis 5.1: Nastavitve Winlogbeat.

```
1  winlogbeat.event_logs:
2    – name: Application
3      ignore_older: 72h
4    – name: Security
5    – name: System
6
7  output.logstash:
8    hosts: ["192.168.1.1:5044"]
```

5.1.2 Operacijski sistem Linux

Operacijski sistemi Linux svoja beleženja navadno shranjujejo v tekstovne datoteke ASCII v standardnem formatu in večinoma se nahajajo v direktoriju »*/var/log*«. Večina jih je generiranih s strani prikritega procesa sistemskega beleženja imenovanega syslogd, nekateri pa imajo tudi svoje generatorje beleženj.

Sistemska beleženja beležijo predvsem kaj se dogaja s sistemom in ne o aplikacijah uporabnika. To vključuje avtorizacijo, prikrite sistemske procese in sistemska sporočila.

```
Aug 17 06:47:10 collector systemd[1]: Starting Daily apt upgrade and clean activities...
Aug 17 06:47:12 collector systemd[1]: Started Daily apt upgrade and clean activities.
Aug 17 07:09:01 collector CRON[34119]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 07:17:01 collector CRON[34135]: (root) CMD (   cd / && run-parts --report /etc/cron.hourly)
Aug 17 07:39:01 collector CRON[34148]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 08:09:01 collector CRON[34177]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 08:17:01 collector CRON[34229]: (root) CMD (   cd / && run-parts --report /etc/cron.hourly)
Aug 17 08:39:01 collector CRON[34240]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 09:09:01 collector CRON[34264]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 09:17:01 collector CRON[34276]: (root) CMD (   cd / && run-parts --report /etc/cron.hourly)
Aug 17 09:39:01 collector CRON[34295]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 10:09:01 collector CRON[34317]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 10:17:01 collector CRON[34329]: (root) CMD (   cd / && run-parts --report /etc/cron.hourly)
Aug 17 10:39:01 collector CRON[34344]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 11:09:01 collector CRON[34370]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
Aug 17 11:17:01 collector CRON[34386]: (root) CMD (   cd / && run-parts --report /etc/cron.hourly)
Aug 17 11:39:01 collector CRON[34401]: (root) CMD (  [ -x /usr/lib/php5/sessionclean ] && /usr/lib/php5/sessionclean)
```

Slika 5.4: Primer beleženja syslog pri operacijskem sistemu Linux.

Avtorizacijska beleženja spremljajo in zapisujejo uporabo avtoriziranih sistemov ter mehanizme za avtorizacijo uporabnikov, kateri povprašajo uporabnika po geslu. To med drugim vključuje ukaze sudo in oddaljene prijave SSH. Avtorizacijska beleženja so dostopna v datoteki »/var/log/auth.log«.

Operacijski sistem Linux lahko že sam, brez dodatnih orodij, pošilja svoja beleženja na določen naslov IP ter vrata. To storimo s spreminjanjem nastavitev programa Rsyslog in sicer v datoteki »*/etc/rsyslog.d/50-default.conf*«. Z dodano vrstico »**.* 192.168.0.1:2542*« povemo na kateri naslov IP in vrata naj bodo poslana beleženja. Če želimo spremljati tudi prijave SSH, moramo povišati stopnjo beleženja v datoteki »*/etc/ssh/sshd_config*« kot prikazuje izpis 5.2.

Izpis 5.2: Stopnja beleženja SSH.

```
1 # Logging
2 SyslogFacility AUTH
3 LogLevel VERBOSE
```

5.2 Normalizacija podatkov

Za prejemanje beleženj iz izvornih naprav, ter normalizacijo le-teh smo uporabili Logstash, ki je del Elastic Stacka. Namestili smo ga na napravo z operacijskim sistemom Linux Ubuntu 18.04 LTS. Logstash je orodje za zbiranje, obdelovanje in posredovanje beleženj ali drugih podatkov, ki mu jih

posredujemo. Zbiranje beleženj in podatkov se doseže z uporabo nastavljivih vhodnih vtičnikov. Nato se s pomočjo mnogih filtrov zbrane podatke obdeluje in označuje. Na koncu pa z izhodnimi vtičniki posreduje obdelane podatke drugim programom.

Obdelava oziroma normalizacija podatkov iz izvornih naprav z operacijskim sistemom Windows ni potrebna, saj to počne že program Winlogbeat sam. Potrebno jih je samo sprejeti na prej izbranih vratih ter jih preusmeriti na upravljalec beleženj in poročil. Nastavitve Logstasha so vidne na izpisu 5.3. Podatke prejmemo na vratih 5044 in jih posredujemo na vrata 9200, na katerih posluša upravljalec beleženj in poročil.

Izpis 5.3: Nastavitve Logstash za preusmerjanje podatkov Winlogbeat.

```
1  input {
2          beats {
3                  port => 5044
4          }
5  }
6
7  output {
8          if [type] == "wineventlog" {
9                  elasticsearch {
10                         hosts => ["localhost:9200"]
11                         index => "beats"
12                 }
13         }
14 }
```

Podatke oziroma beleženja, ki jih Logstash prejme iz izvornih naprav z operacijskim sistemom Linux so neobdelana. Zato preden jih posredujemo upravljalcu beleženj in poročil, jih z programom Logstash normaliziramo z uporabo filtrov Grok [9] in Mutate [10]. Z filtrom Grok lahko razčlenimo in strukturiramo poljubno besedilo ter s tem podatki postanejo lažje poizvedljivi. Z filtrom Mutate lahko izvajamo splošne transformacije na poljih strukturiranih podatkov. Polja lahko preimenujemo, odstranimo, nadomestimo ali spremenimo.

Struktura beleženj Syslog, ki jih Logstash prejme iz izvornih naprav Linux, vsebuje datum in uro zapisa, ime naprave, ime programa ali ukaza kateri je naredil zapis ter še dodatno nestrukturirano sporočilo. S pomočjo filtra Grok vsako prejeto beleženje najprej razčlenimo na prej omenjene štiri podatke. Vsa beleženja, katera niso bila zapisana iz strani programa sshd ali ukaza sudo, zavržemo. Nato iz dodatnega sporočila strukturiramo podatke in jih posredujemo upravljalcu beleženj in poročil.

Iz beleženj, ki jih ustvari program sshd, lahko razberemo oddaljene prijave na napravo. Podatke, ki jih shranimo ob vsakem takšnem beleženju so ime naprave in naslov IP, na katero je oddaljena prijava potekala, naslov IP ter vrata iz katerega je prišla prijava, na katerega uporabnika je potekala prijava ter ali je bila prijava uspešna. Te podatke normaliziramo z uporabo filtrov Grok in Mutate, kot prikazuje izpis 5.4. V nekaterih zapisih beleženj lahko kakšni podatki manjkajo, zato jih vstavimo kot prazna polja.

Izpis 5.4: Primer normalizacije beleženj sshd.

```
1  grok {
2    match => { "msg" => "Accepted password for %{USERNAME:user} from %{IP:
         rhost} port %{INT:port} ssh2" }
3  }
4  mutate {
5      add_field => {
6        "authentication" => "successful"
7        "logname" => ""
8        "uid" => ""
9        "euid" => ""
10       "tty" => ""
11       "ruser" => ""
12 "section" => "auth-log"
13     }
14 }
```

Pri uporabi ukaza sudo, se ustvari beleženje, iz katerega lahko razberemo kateri ukaz je bil zagnan z pravicami superuser, kateri uporabnik ga je izvedel ter v katerem direktoriju je bil izveden. Z uporabo filtra Grok ta zapis

strukturiramo in ga posredujemo upravljalcu beleženj in poročil. Primer normalizacije takšnih beleženj je prikazan na izpisu 5.5.

Izpis 5.5: Primer normalizacije beleženj sudo.

```
1 grok {
2     match => {
3         "msg1" => ": %{USERNAME:user} : TTY=%{GREEDYDATA:tty} ; PWD
              =%{GREEDYDATA:pwd} ; USER=%{USERNAME:sudo_user} ;
              COMMAND=%{GREEDYDATA:command}"
4     }
5 }
```

5.3 Upravljalec beleženj in poročil

Za upravljalca beleženj in poročil smo uporabili Elasticsearch, ki je del Elastic Stacka. To je zelo razširljivo, iskalno in analitično orodje. Omogoča hitro shranjevanje, analiziranje in poizvedovanje po velikih količinah podatkov skoraj v realnem času. Elasticsearch podatke shranjuje v tako imenovane indekse, to so zbirke podatkov, ki imajo podobne značilnosti. Sami smo uporabili tri, enega za shranjevanje beleženj operacijskih sistemov Windows, drugega za beleženja operacijskih sistemov Linux ter tretjega za shranjevanje poročil, ki jih ustvari sistem SIEM. Za shranjevanje in poizvedbo po podatkih Elasticsearch uporablja REST api.

5.4 Analitično orodje

Za analitično orodje smo uporabili Apache Spark ter Hadoop. Hadoop je orodje, ki omogoča vzpostavitev infrastrukture za procesiranje ter shranjevanje Big Data podatkov z uporabo gruče računalnikov. Spark je orodje za paralelizacijo procesiranja podatkov, ki deluje na infrastrukturi Hadoop. S tem dosežemo razširljivost sistema, saj v primeru prevelike količine podatkov za obdelavo lahko povečamo gručo računalnikov v infrastrukturi Hadoop. S

tem tudi zagotovimo prepoznavanje incidentov ali varnostnih groženj v realnem času pri večji količini podatkov in se izognemo ozkemu grlu.

Javanski program, ki ga zaženemo s pomočjo Sparka predstavlja pravilo za prepoznavanje incidentov ali varnostnih groženj. Zaganjamo ga periodično na par minut, saj le na ta način lahko pravočasno obvestimo administratorja. To zaganjanje opravlja program Cron. V nadaljevanju je opisano delovanje programa za preverjanje dosegljivosti spletnega strežnika.

Izpis 5.6: Del programske kode za preverjanje dosegljivosti spletnega strežnika.

```
1  public static void main(String[] args) throws IOException {
2   observed_host = args[0];
3   observed_port = Integer.parseInt(args[1]);
4
5   generateQuery();
6
7   System.out.printf("Checking␣host:␣" + observed_host + "␣port:␣" + observed_port);
8
9   boolean isUp = pingHost(observed_host, observed_port, 100);
10
11  System.out.printf("␣status:␣" + isUp + "␣");
12
13  JSONObject json = sendGetRequest(url, port, "GET", pathQuery, query);
14  boolean exists = json.getJSONObject("hits").getInt("total") > 0;
15
16  System.out.printf("report␣entry:␣" + exists + "\n");
17
18  if ( !isUp && !exists ) {
19   createNewReport();
20  } else if ( isUp && exists ) {
21   String reportID;
22   reportID = json.getJSONObject("hits").getJSONArray("hits").getJSONObject(0).
         getString("␣id");
23   changeStatus(reportID);
24  }
25 }
```

Program v 2. in 3. vrstici iz podanih argumentov pridobi naslov in vrata izbranega spletnega strežnika. S pomočjo teh spremenljivk, program v 5. vrstici ustvari poizvedbo za iskanje že obstoječega poročila o nedosegljivosti spletnega strežnika. v 7. vrstici program izpiše na katerem naslovu in vratih preverja dosegljivost spletnega strežnika. V vrstici 9 program ustvari boolean spremenljivko, katera pridobi vrednost glede na pingHost funkcijo. Funkcija pingHost prejme za svoje argumente naslov in vrata spletnega strežnika ter čas za prekinitev. Funkcija vrne boolean vrednost glede na dosegljivost spletnega strežnika. V vrstici 11 program izpiše dosegljivost spletnega strežnika. Funkcija sendGetRequest iz Elasticsearch-a vrne objekt json, ki vključuje rezultate poizvedbe, katero je program ustvaril v 5. vrstici. Kot argumente prejme naslov ter vrata Elasticsearch programa, uporabljeno metodo http, ime indeksa v katerem so iskani podatki ter samo poizvedbo. V vrstici 14 program ustvari boolean spremenljivko, katera je TRUE v primeru, če poročilo obstaja, da ne ustvari novega. To program tudi izpiše v vrstici 16. V primeru, da sta obe boolean spremenljivki FALSE program v 19. vrstici s funkcijo createNewReport ustvari novo poročilo o nedosegljivosti strežnika. Če sta obe spremenljivki TRUE program s funkcijo changeStatus posodobi status poročila. Kot argument funkcija changeStatus prejme identifikacijsko številko poročila. To program pridobi iz rezultatov poizvedbe v vrstici 22.

5.5 Vizualizacijsko orodje

Za vizualizacijsko orodje smo uporabili Kibano, ki je del Elastic Stacka. Ta služi kot vizualizacijsko orodje oziroma pregledna plošča, katera je del vsakega sistema SIEM. Kibana je analitično in vizualizacijsko orodje, zasnovana za delo z orodjem Elasticsearch. Uporablja se za iskanje, ogled in interakcijo s podatki, ki so shranjeni v indeksih Elasticsearch-a.

Kibana je dostopna na privzetih vratih 5601. Z uporabo spletnega strežnika Nginx lahko preusmerimo promet iz vrat 80 na vrata 5601. Prav tako lahko z Nginx dodamo omejitev dostopa z uporabo uporabniškega imena in gesla.

Primer nastavitev spletnega strežnika Nginx je prikazan na izpisu 5.7.

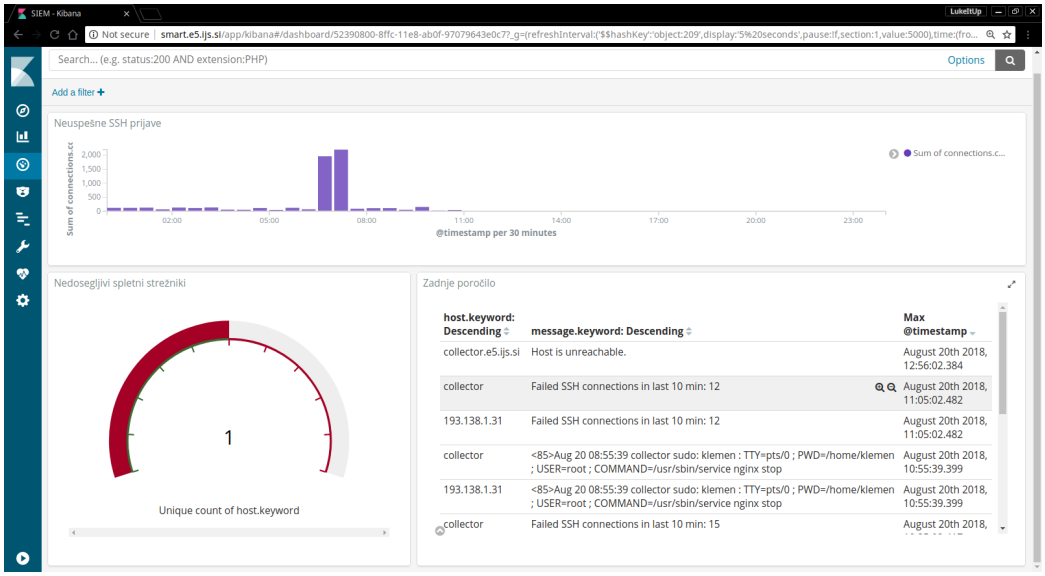
Izpis 5.7: Nastavitve spletnega strežnika Nginx.

```
1  server {
2          listen 80;
3                  server_name kibana;
4
5          error_log /var/log/nginx/kibana.error.log;
6          access_log /var/log/nginx/kibana.access.log;
7
8          location / {
9                  rewrite ^/(.*) /$1 break;
10                 proxy_ignore_client_abort on;
11                 proxy_pass http://localhost:5601;
12                 proxy_set_header X-Real-IP $remote_addr;
13                 proxy_set_header X-Forwarder-For $proxy_add_x_forwarded_for;
14                 proxy_set_header Host $http_host;
15                 auth_basic "Restricted";
16                 auth_basic_user_file /etc/nginx/.htpasswd;
17         }
18 }
```

V Kibani lahko ustvarimo poljubno vizualizacijo s klikom na »Vizualize«. Nato si izberemo tip vizualizacije. Na izbiro imamo diagrame, kot so črtni, stolpčni ter tortni, podatkovne tabele, merilce in zemljevide. Ko izberemo tip vizualizacije, izberemo indeks s podatki, ki jih želimo vizualizirati. Nato pričnemo z vizualizacijo, kjer dodamo filtre nad podatki iz indeksa, jih agregiramo in določimo, kaj bo vizualizacija prikazovala. Ko končamo, vizualizacijo shranimo. S klikom na »Dashboard« lahko ustvarimo pregledno ploščo, kjer prikazujemo istočasno več vizualizacij.

Na pregledno ploščo sistema SIEM, ki je vidna na sliki 5.5, smo vključili tri vizualizacije. Prva vizualizacija prikazuje število vseh neuspelih prijav SSH v razponu celotnega dneva. Kot je na sliki razvidno, se je število neuspelih prijav SSH povišalo ob 7 uri zjutraj. V spodnjem levem kotu pregledne plošče se nahaja merilec, ki prikazuje število nedosegljivih spletnih strežnikov katere spremljamo. V spodnjem desnem kotu pregledne plošče se nahaja tabela, ki

prikazuje zadnja poročila sistema SIEM in sicer o kateri izvorni napravi govori poročilo, kratek opis poročila ter čas, ko je bilo poročilo ustvarjeno.



Slika 5.5: Pregledna plošča sistema SIEM.

Poglavje 6

Ovrednotenje

V naslednjem poglavju preverimo in ovrednotimo našo implementacijo odprtokodnega sistema SIEM. Preverimo, če smo dosegli vse zastavljene cilje in zahteve, ki smo si jih zadali v načrtu izdelave v poglavju 4.

6.1 Doseganje zahtev

Odprtokodni sistem SIEM, katerega smo implementirali vključuje vseh šest členov iz principa delovanja verige SIEM. Sistem je prilagodljiv in razširljiv neglede na velikost informacijsko-komunikacijskega sistema. Priključitev novih izvornih naprav je enostavna, prav tako je dodajanje novih pravil. Normalizacija in vizualizacija prejetih podatkov deluje v realnem času. Dostop do pregledne plošče je omejen z uporabniškim imenom in geslom, prav tako je omejen dostop do spreminjanja normalizacije ter samih podatkov.

6.2 Dostop SSH

Pri zaznavanju neuspelih dostopov SSH sistem SIEM s poizvedbo od upravljalca beležnj in poročil prejme podatke o številu neuspelih prijav SSH v zadnjih 10 minutah ter iz katerih naslovov IP je prijava potekala, za vsako izvorno napravo ki pošilja beleženja syslog. Glede na podane argumente, ki

smo jih podali programu, določi ali je število neuspelih prijav SSH preseglo mejo in ali ustvari poročilo. Poročilo, ki je vidno na sliki 6.1, vključuje časovno značko, naslove IP ter število neuspelih prijav SSH, skupno število vseh neuspelih prijav SSH, naslov ogrožene naprave ter kratko sporočilo.

Slika 6.1: Primer poročila o prevelikem številu neuspelih prijav SSH.

6.3 Zaznavanje naprave USB

Pri zaznavanju priključitve nove naprave USB, sistem SIEM najprej s poizvedbo od upravljalca beleženj in poročil prejme imena vseh izvornih naprav Windows, ki so v zadnji minuti poslale dogodek z identifikacijsko številko 2003, 2004, 2006, 2010, 2100, 2101, 2105 ali 2016. V primeru, da najde kakšen dogodek z eno izmed naštetih identifikacijskih številk, ustvari poročilo, katero je vidno na sliki 6.2 ter ga shrani v upravljalca beleženj in poročil. Ta vključuje časovno značko, ime naprave ter kratko sporočilo o novi priključeni napravi USB.

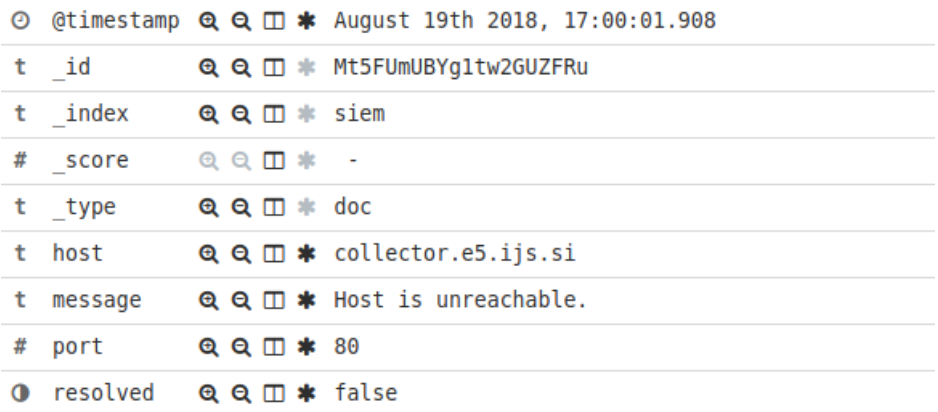
@timestamp		August 20th 2018, 21:54:48.157
t	_id	P4CtgWUBM6j7t6bizmAn
t	_index	siem
#	_score	-
t	_type	_doc
t	host	DESKTOP-AKOAMJU
t	message	Newly discovered USB device.

Slika 6.2: Primer poročila ob zaznavi naprave USB.

6.4 Dosegljivost spletnega strežnika

Pri preverjanju dosegljivosti spletnega strežnika sistem SIEM najprej poskuša vzpostaviti povezavo na podanem naslovu in vratih. V primeru, da povezava ni uspela, program poizve v upravljalcu beleženj in poročil ali že obstaja poročilo o nedosegljivosti strežnika. Če ta še ne obstaja, ustvari novega. V primeru, da strežnik deluje, program prav tako poizve v upravljalcu beleženj in poročil ali obstaja poročilo o nedelovanju in ga popravi v zaključeno ter doda časovno značko kdaj je strežnik postal dosegljiv. Poročilo, ki je vidno na

sliki 6.3, vključuje časovno značko nedosegljivosti, naslov ter vrata spletnega strežnika, kratko sporočilo in oznako, da poročilo ni zaključeno. V primeru, da bi poročilo bilo že zaključeno, bi vsebovalo še časovno značko kdaj je spletni strežnik postal dosegljiv.



Slika 6.3: Primer poročila o nedosegljivem strežniku.

6.5 Uporaba ukaza sudo

Pri zaznavanju uporabe ukaza sudo sistem SIEM najprej v upravljalcu beleženj in poročil poizve ali se je na kateri izmed izvornih naprav Linux v zadnji minuti izvedel ukaz sudo. Za vsako napravo, ki je izvedla ukaz sudo ustvari poročilo, katero je vidno na sliki 6.4. Poročilo vsebuje časovno značko, kateri ukaz je bil izveden, na kateri napravi se je izvedel ukaz, sporočilo, v katerem direktoriju je bil izveden ter kdo ga je izvedel.

@timestamp	August 19th 2018, 18:14:51.014
t @version	1
t _id	fN73UmUBYg1tw2GUuVYQ
t _index	siem
# _score	-
t _type	doc
t command	/usr/sbin/service nginx start
t host	193.138.1.31, collector
t message	sudo: klemen ; PWD=/home/klemen ; USER=root ; COMMAND=/usr/sbin/service nginx start
t program	sudo
t pwd	/home/klemen
t sudo_user	root
t user	klemen

Slika 6.4: Primer poročila o izvedbi ukaza sudo.

Poglavje 7

Zaključek

V informacijsko-komunikacijskih sistemih je danes vključenih vedno več naprav. Vzdrževanje in varovanje takšnega informacijsko-komunikacijskega sistema postaja vse težje. V diplomskem delu smo spoznali sisteme SIEM, kako delujejo ter njihovo funkcijo v informacijsko-komunikacijskih sistemih. Pregledali smo nekatere obstoječe plačljive in odprtokodne sisteme SIEM. Nato smo sami implementirali sistem SIEM z uporabo opdrtokodnih orodij. Ta je dosegel vse zastavljenje zahteve in cilje. Menimo, da je sistem SIEM, ki smo ga implementirali, uporaben tudi v resničnem informacijsko-komunikacijskem sistemu. Z implementacijo dodatnih pravil za zaznavanje incidentov in varnostnih groženj se lahko približamo obstoječim sistemom SIEM na tržišču. Pomanjkljivost našega sistema SIEM je proženje pravil na minutne intervale, saj program Cron ne omogča proženja na krajših intervalih. Zato sistem SIEM ne deluje povsem v realnem času. Prav tako je slabost sinhronizacija ur na napravah v informacijsko-komunikacijskih sistemih, saj poročila vsebujejo časovne značke sistema SIEM, medtem ko zbrana beleženja pa časovne značke izvornih naprav.

Literatura

[1] Alien vault ossim. Dosegljivo: `https://www.alienvault.com/products/ossim`. [Dostopano: 15. 8. 2018].

[2] Apache spark. Dosegljivo: `https://spark.apache.org/`. [Dostopano: 20. 8. 2018].

[3] Sandeep Bhatt, Pratyusa K Manadhata, and Loai Zomlot. The operational role of security information and event management systems. *IEEE security & Privacy*, 2014.

[4] John R. Boyd. Zanka ooda. Dosegljivo: `http://www.goalsys.com/books/documents/DESTRUCTION_AND_CREATION.pdf`, 1976. [Dostopano: 25. 7. 2018].

[5] Dr. Anton Chuvakin. Zgodovina in razvoj siem sistemov ter kako jih vpeljati v podjetja. Dosegljivo: `https://www.microfocus.com/media/white-paper/the_complete_guide_to_log_and_event_management_wp.pdf`, 2016. [Dostopano: 25. 7. 2018].

[6] Cron. Dosegljivo: `http://www.adminschoice.com/crontab-quick-reference`. [Dostopano: 25. 8. 2018].

[7] Elastic stack. Dosegljivo: `https://www.elastic.co/elk-stack`. [Dostopano: 25. 8. 2018].

[8] Elasticsearch. Dosegljivo: https://www.elastic.co/products/elasticsearch. [Dostopano: 25. 8. 2018].

[9] Filter grok za orodje logstash. Dosegljivo: https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html. [Dostopano: 25. 8. 2018].

[10] Filter mutate za orodje logstash. Dosegljivo: https://www.elastic.co/guide/en/logstash/current/plugins-filters-mutate.html. [Dostopano: 25. 8. 2018].

[11] Gradilni bloki sistema siem. Dosegljivo: http://infosecnirvana.com/enterprise-siem-implementation-building-blocks/, 2016. [Dostopano: 2. 8. 2018].

[12] Hadoop. Dosegljivo: http://hadoop.apache.org/. [Dostopano: 20. 8. 2018].

[13] Ibm qradar. Dosegljivo: https://www.ibm.com/us-en/marketplace/ibm-qradar-siem. [Dostopano: 14. 8. 2018].

[14] Java. Dosegljivo: https://www.java.com/en/. [Dostopano: 20. 8. 2018].

[15] Kibana. Dosegljivo: https://www.elastic.co/products/kibana. [Dostopano: 27. 8. 2018].

[16] Logstash. Dosegljivo: https://www.elastic.co/products/logstash. [Dostopano: 25. 8. 2018].

[17] Mark Maiville. Gartner poročilo siem sistemov za leto 2016. Dosegljivo: https://www.idrgrp.com/magic-quadrant-2016/, 2016. [Dostopano: 6. 8. 2018].

[18] Trevan Marden. Filtriranje beleženj. Dosegljivo: http://blog.cygilant.com/blog/making-sense-of-information-security-technologies-ids-ips-utm-and-siem, 2016. [Dostopano: 2. 8. 2018].

[19] David Miller. *Security information and event management (SIEM) implementation*. McGraw-Hill, New York, 2011.

[20] Tim Grance Karen Scarfone Paul Cichonski, Tom Millar. Nist publikacija o odzivanju na incidente in grožnje. Dosegljivo: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf, 2012. [Dostopano: 2. 8. 2018].

[21] Scott J Roberts and Rebekah Brown. *Intelligence-Driven Incident Response: Outwitting the Adversary*. O'Reilly Media, 2017.

[22] Mike Rotham. Zgodovina in razvoj siem sistemov. Dosegljivo: https://searchsecurity.techtarget.com/video/The-past-present-and-future-of-SIEM-technology, 2014. [Dostopano: 18. 7. 2018].

[23] Margaret Rouse. Cia trojica. Dosegljivo: https://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA, 2014. [Dostopano: 17. 7. 2018].

[24] Siemonster. Dosegljivo: https://siemonster.com/. [Dostopano: 27. 8. 2018].

[25] Splunk. Dosegljivo: https://www.splunk.com/. [Dostopano: 17. 8. 2018].

[26] Ubuntu 18.04 lts. Dosegljivo: http://releases.ubuntu.com/18.04/. [Dostopano: 25. 8. 2018].

[27] Windows 10. Dosegljivo: https://www.microsoft.com/en-us/windows. [Dostopano: 25. 8. 2018].

[28] Windows 7. Dosegljivo: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-7/dd349342(v=ws.10). [Dostopano: 25. 8. 2018].