Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

SIEM - monitoring bezpečnosti sítě

Petr Vyhnal

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Petr Vyhnal

Systémové inženýrství a informatika
Informatika

Název práce

SIEM – monitoring bezpečnosti sítě

Název anglicky

SIEM – network security monitoring

Cíle práce

Cílem práce je re-design řešení SIEM pro monitoring bezpečnosti sítě v prostředí velké firmy s ohledem na škálovatelnost a spolehlivost.
Dílčími cíli jsou:
- identifikace slabých míst současného řešení
- otestování možných řešení nalezených slabých míst

Metodika

Práce je založena na studiu odborné literatury k dané problematice a na základě konzultací odborníků z praxe.

První dílčí cíl bude řešen především analýzou současného stavu.

Druhý dílčí cíl bude řešen formou pozorování a srovnávání.

Na základě syntézy získaných poznatků bude formulováno doporučení pro nový design řešení SIEM.

Doporučený rozsah práce

60 – 80 stran

Klíčová slova

SIEM, security, SOC, log, event, bezpečnost

Doporučené zdroje informací

Conklin, W. A., White, G., Cothren, C., Davis, R. L., Williams, D. CompTIA Security+ All-in-One Exam Guide, Fifth Edition. New York: McGraw-Hill/Osborne, 2018. ISBN 978-1260019322.

Dostálek, L. a kolektiv Velký průvodce protokoly TCP/IP: Bezpečnost 2. aktualizované vydání. Praha: Computer Press, 2003. ISBN 80-7226-849-X.

Dye, M., McDonald, R., Rufi, A. Network Fundamentals, CCNA Exploration Companion Guide. Indianapolis: Cisco Press, 2007. 560 s. ISBN-10: 1-58713-208-7.

Gordon, A. Official (ISC)2 Guide to the CISSP CBK, Fourth edition. New York: CRC Press, 2015. ISBN 978-1-4822-6275-9.

McClure, S., Scambray, J., Kurtz, G. Hacking Exposed 7: Network Security Secrets & Solutions. New York: McGraw-Hill, 2012. ISBN 978-0-07-178028-5.

Miller, D. R., Harris, S., Harper, A. A., Vandyke, S., Blask, C. Security Information and Event Management (SIEM) Implementation. New York: McGraw-Hill/Osborne, 2011. ISBN 978-0071701099.

Stewart, J. M., Chapple, M., Gibson, D. CISSP Study Guide. Indianapolis: John Willey & Sons, Inc., 2015. ISBN : 978-1-119-04271-6.

Strebe, M., Perkins, Ch. Firewally a proxy-servery Praktický průvodce. Brno: Computer Press, 2003. 450 s. ISBN 80-7226-983-6.

Předběžný termín obhajoby

2019/20 LS – PEF

Vedoucí práce

Ing. Martin Havránek, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 14. 10. 2019

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 14. 10. 2019

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 29. 02. 2020

Čestné prohlášení:

Prohlašuji, že svou diplomovou práci “SIEM - monitoring bezpečnosti sítě” jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 29. 2. 2020 Petr Vyhnal

Poděkování

Rád bych poděkoval svému vedoucímu práce, panu Ing. Martinu Havránkovi, Ph.D., za poskytnuté konzultace k vypracování této práce.
Také bych rád poděkoval manželce Janě za podporu a pomoc s korekturou práce a celé rodině za trpělivost. Dále bych chtěl poděkoval i svým kolegům Stacy D. Uden, Valerie H. Devera, Daniel Voicu a Justin H. Haynes za odborné připomínky a konzultace a Tadeášovi Menglerovi a společnosti Stable.cz, s.r.o. za laskavé zapůjčení hardware na provádění testů.
V neposlední řadě bych rád poděkoval i tvůrcům nástrojů TEX a LATEX, s jejichž pomocí byl tento dokument vysázen.

SIEM - monitoring bezpečnosti sítě

Abstrakt

Práce si klade za cíl představit nástroje SIEM, které se využívají při sledování bezpečnosti sítí. V úvodní části je přiblíženo co vlastně SIEM nástroje či platformy jsou a jak mohou pomoci se sledováním bezpečnosti. Jsou zde také představeny nejrozšířenější zástupci SIEM nástrojů. Druhá část pak předkládá současný stav nasazení SIEM platformy ArcSight. Dále pojednává o dalších možnostech nasazení či integrace jednotlivých komponent. Následně představuje návrh optimalizovaného řešení, které zajišťuje dostatečnou škálovatelnost a spolehlivost.

Klíčová slova: SIEM, Zabezpečení, Síť, Událost, ArcSight, Splunk, Qradar, LogRhythm, Firewall, IDS, IPS, syslog

SIEM - network security monitoring

Abstract

This thesis aims to introduce SIEM tools or solutions which are designed to be used to monitor network security. First part brings overview of what SIEM tools are and how they can help to security professionals to monitor network security. Also the biggest players in this industry are presented there. Second part then introduces existing implementation of SIEM ArcSight solution. It further presents other available options of design or integration. Finally it suggest new design optimizes to provide scalability and reliability.

Key words: SIEM, Security, Network, Event, ArcSight, Splunk, Qradar, LogRhythm, Firewall, IDS, IPS, syslog

Obsah

Úvod

Kdo má informace, má moc. To je historií prověřené pravidlo, které v dnešní informační době platí dvojnásob. Informace - a především ty uniklé - jsou "vyvažovány zlatem" nebo častěji dolary či nově dokonce kryptoměnami, které zajišťují vyšší anonymitu. Největším strašákem v oblasti bezpečnosti IT velkých firem je právě únik dat. Ten může významně ohrozit reputaci firmy, v krajním případě odstartovat i její pád.

Datová komunikace je dnes všudypřítomná a to jak v pracovní, tak v osobní sféře. Rozsáhlá integrace různých systémů nebo provázanost a vysoká dostupnost obsahu jen zvyšují riziko úniku potenciálně nebezpečných dat. Na internetu se pohybuje množství "black hat" hackerů, kteří se cílenými útoky na privátní a tajná data živí. Podle ukazatele Breach Level Index došlo v roce 2017 k odcizení 2,5 miliard datových záznamů, což představuje nárůst o 88% oproti předchozímu roku (Gemalto NV, 2017). Neslavné prvenství získala americká společnost Equifax, jíž unikly data až 143 miliónů subjektů. Útočníkům se kromě citlivých informací jako jména, adresy, či čísla sociálního pojištění, podařilo získat i kolem dvou set tisíc čísel kreditních karet.

Nástroje z rodiny SIEM by pak měly především pomoci předejít nebo alespoň dostatečně rychle odhalit probíhající útok nebo jeho přípravu. Zajišťují ale také archivaci událostí, která může pomoci při případné forenzní analýze útoku. Dále je možné jejich využití pro doložení shody s definovaným standardem například při bezpečnostních auditech.

1 Cíl práce a metodika

1.1 Cíl práce

Tato práce se sestává ze několika dílčích cílů. Předně půjde o zmapování současných možností a trendů předních platforem v oblasti SIEM. Důraz při tom bude kladen na dostatečnou škálovatelnost platformy a podporu předpokládaného portfolia zdrojů. Zároveň bude provedeno šetření aktuálních potřeb společnosti pro nasazení SIEM platformy, jelikož současný produkt SIEM ArcSight byl nasazen před více než šesti lety a celkový design odpovídá možnostem té doby. Dále bude definováno portfolio zařízení a aplikací, které by měly být s řešením integrované a jejichž události by měly být předmětem korelačních analýz a případných notifikací či alarmů. Shromážděné podklady pomohou rozhodnout jakým způsobem bude naloženo s aktuálně nasazeným SIEM ArcSight, tj. zda se dále budeme zabývat pouze změnou architektury existujícího řešení tak, aby vyhovovala aktuálním požadavkům a využívala možnosti dané výrobcem nebo jeho úplným nahrazením. Dále se práce bude zabývat identifikací slabých míst v současném návrhu architektury sběru událostí a jejich možnou eliminací.

Hlavním cílem práce pak bude navrhnout optimální design architektury SIEM platformy právě s ohledem na specifické potřeby společnosti a plánovaný budoucí rozvoj. Součástí by měl být plán na realizaci uvažovaných změn a to jak v případě migrace na jiný produkt, tak v případě pouhé úpravy architektury stávající.

1.2 Metodika práce

Nejprve bude provedena analýza aktuálních potřeb společnosti na řešení SIEM a bude vytvořen aktuální model architektonického řešení a portfolio zdrojů, které má být integrováno se SIEM platformou. Následně bude provedena rešerše na trhu dostupných řešení, která by měla potřeby splňovat. Po té bude následovat komparace jednotlivých řešení s vytvořeným portfoliem produktů a určení nejvhodnějšího. Na základě tohoto výběru bude vytvořen konečný model nového řešení.

2 Teoretická část

2.1 Co je SIEM

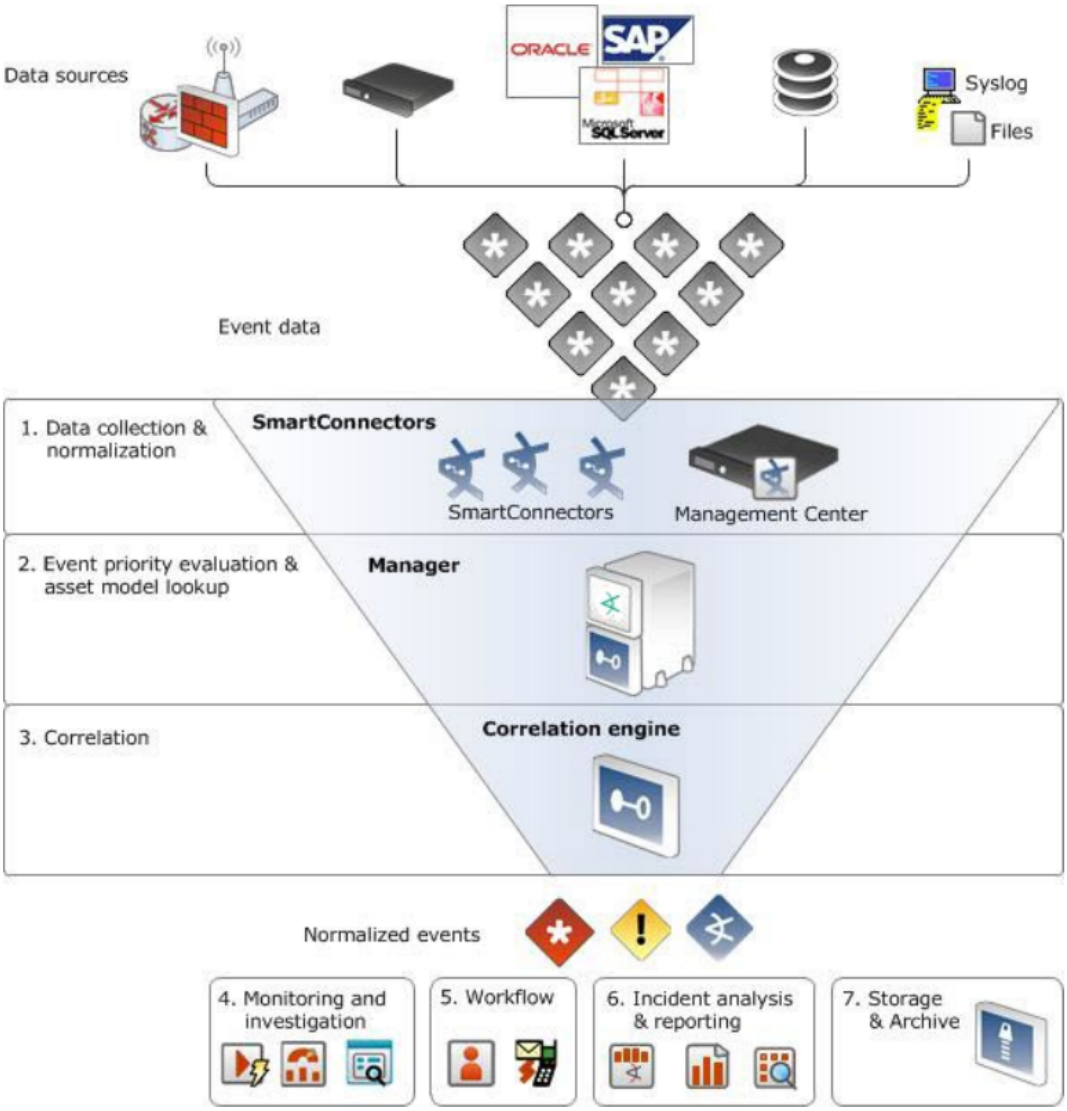
SIEM je anglická zkratka Security Incident and Event Management. Jedná se o termín zahrnující širší okruh technologií, které mají za cíl agregovat data o přístupech a o vybraných aktivitách systémů a ukládat je pro jejich následnou analýzu a korelaci. (Gordon, A., 2015) Daty jsou v tomto případě logy, které jsou v SIEM terminologii označovány jako události a jsou generovány prakticky všemi zařízeními v síti, servery i aplikacemi. Primárním cílem produktů SIEM je analyzovat tyto události a na základě zadaných podmínek vyhledávat potenciální kritické události, které mohou znamenat ohrožení bezpečnosti.

Úkoly SIEM můžeme rozdělit do dvou hlavních oblastí. První je SIM (Security Information Management). SIM je zaměřen primárně na sběr a uchovávání bezpečnostních událostí. Dále je možné provádět nad uloženými daty různé analýzy či sledovat trendy. Druhou oblastí je SEM (Security Event Management), který se soustředí na analýzu a korelaci událostí v reálném čase tak, jak přicházení do systému, případně v krátkých časových úsecích. V příchozích událostech vyhledává podezřelé vzory a upozorňuje na evidentní nebo jen potenciální problémy. Produkty ze segmentu SIEM se u společností těší rostoucí popularitě při řešení automatizace monitoringu bezpečnosti. (McClure, S., Scambray, J., Kurtz, G., 2012)

Jak již bylo zmíněno, SIEM nástroje jsou primárně zaměřené na bezpečnost a jejich hlavním uživatelem jsou týmy, které se povětšinou označují jako SOC, z anglického Security Operation Center. Týmy SOC jsou většinou první úrovní analytiků a primárně pracují se SEM částí celého SIEM řešení. Především u větších společností existují i další úrovně - týmy nazývané obvykle nějakou ze zkratek jako CSIRT (Computer Security Incident Response Team) nebo CERT (Computer Emergency Response Team) nebo nějakou podobnou variací. V těchto týmech většinou pracují zkušení analytici a jejich úkolem jsou pak pokročilejší analýzy útoků, např. forenzní analýzy a podobně. Při jejich vyšetřování velmi často využívají i archivní data, čili část SIM.

2.2 Jak SIEM pracuje

Jak již bylo zmíněno, platformy SIEM mají dvě hlavní úlohy. Zajistit analýzu událostí v (téměř) reálném čase a poskytovat služby správy historických událostí. Díky tomu pomáhá zajistit organizaci odhalování vnitřních či vnějších hrozeb, monitorování činnosti uživatelů - především přístupy do systémů nebo k privilegovaným funkcím, poskytuje důkazy v případě vyšetřování incidentů, poskytuje výstupy pro případné audity shody a zpravidla disponuje i nějakými procesy na řešení incidentů.



Obrázek 2.1: Životní cyklus události v SIEM ArcSight (Hewlett Packard Enterprise Development, LP , 2018)

A. Gordon shrnuje hlavní důvody pro využití systémů SIEM do těchto bodů:

- regulatorní požadavky a kontrola shody
- kontrola odpovědnosti a nezpochybnitelnosti
- kontrola rizik
- sledování výkonnosti a trendů
- korelace událostí a analýza příčin
- odbavování incidentů (událostí)
- bezpečnostní vyšetřování

(Gordon, A., 2015)

2.2.1 Sběr dat

Zdrojem dat - událostí bývají především obvyklé typy síťových zařízení jako jsou firewally, IDS/IPS senzory, proxy servery, routery či switche, ale i servery, antivirové systémy nebo konkrétní aplikace či dnes stále populárnější cloudové služby. K samotnému sběru dat se obvykle používá nějaký typ agenta, který je schopen získat data ze zdrojového systému. Každý výrobce totiž nabízí jiné možnosti přenosu událostí a také samotný formát se povětšinou liší.

U síťových zařízení je obvyklým transportním protokolem syslog, což je poměrně jednoduchý protokol přenášející pomocí paketů UDP logy v čistém textu. Obvykle je jeden UDP datagram nese jeden záznam události. Alternativně lze využít i protokolu TCP pro dosažení jistějšího doručení. Existuje i možnost celý přenost šifrovat pomocí TLS. Vrstva TLS nikterak nezkoumá data, která jsou jí zasílána aplikační vrstvou, pouze je zabezpečí a předá protokolu TCP. (Dostálek, L. a kolektiv, 2003) I tak je ale problémem, že krom samotné hlavičky syslog zprávy, která je standardizovaná v RFC5424 (viz. obr. 2.3) případně starším RFC3164 (viz. obr. 2.2), je samotná zpráva ve formátu, který si vymyslel výrobce. Aby tedy mohl SIEM s danou zprávou nějak dále pracovat. Musí být schopen porozumět jejímu obsahu. Tedy přesněji který údaj co značí.

Jak je na první pohled patrné, oba formáty zprávy syslog se výrazně liší. Novější formát obsahuje přesnější časový údaj, tzv. časové razítko. Dokáže díky tomu pracovat s časy v

```
<86>Dec 29 12:18:27 goliath sudo: pam_unix(sudo:session): session closed for user root
```

Obrázek 2.2: Událost ve formátu syslog dle RFC3164 (zdroj: vlastní)

```
<86>1 2018-12-29T12:20:15.451248+01:00 goliath sudo - - - pam_unix(sudo:session): session closed for user root
```

Obrázek 2.3: Událost ve formátu syslog dle RFC5424 (zdroj: vlastní)

různých časových pásmech. V případě staršího formátu se automaticky předpokládá, že čas je ve shodném časovém pásmu jako je nastaveno na straně příjemce. Vedle časového razítka obsahuje hlavička také prioritu, v našem případě je to 86 a také jméno zdroje, který událost vytvořil - v našem případě "goliath". Po jméně pak následuje samotné tělo syslog zprávy, které nese detaily o dané události. V případě novějšího RFC5424 je součástí hlavičky ještě informace o verzi protokolu, která je umístěna hned za prioritou.

U jiných typů zdrojových zařízení se mohou využívat jiné typy agentů. Lze se setkat s agenty databázovými, agenty čtoucími textové log soubory na zdrojovém systému, u webových aplikací a cloudových služeb jsou pak často využíváno nějaké API, například REST API ve spojení s nějakým formátem určeným pro výměnu dat - např. JSON či XML. I zde je však potřeba vyřešit problém porozumění obsahu dat.

2.2.2 Zpracování dat

Aby analytická část SIEM řešení mohla nějak se sesbíranými daty pracovat, analyzovat je, vyhodnocovat, případně korelovat, musí nutně chápat jejich obsah. Musí zde existovat nějaký převod nebo nějaké mapování mezi formátem dat poskytnutým výrobcem zdrojového produktu a formátem používaným interně v SIEM řešení. Možné jsou v zásadě dva přístupy. Jedním je normalizace dat za pomoci nějakých šablon, kdy je původní formát zcela přeměněn na formát nový, strukturovaný. Toto řešení používá například platforma MicroFocus ArcSight, která za pomoci tzv. "parserů", které jsou přítomny na agentu přeformátuje původní data do formátu zvaného CEF, což je strukturovaný formát, se kterým dál pracují všechny komponenty.

Další možností je ponechání dat v jejich původním formátu, kdy k překladu na strukturovaný unifikovaný formát dochází v reálném čase při prezentaci výsledků za pomocí indexace. Toto řešení je využíváno například platformou Splunk. Nespornou výhodou tohoto přístupu je zachování původního formátu, kdy je zde jasná zpětná návaznost na

dokumentaci logů výrobce zdrojového zařízení a nehrozí, že nějaká informace bude díky chybě parseru nebo úmyslně vypuštěna. Data v původním formátu lze pak využít i pro aktivity nesouvisející přímo s bezpečností, například administrátory daných zdrojových zařízení. Na druhou stranu vyžaduje opakované převádění do strukturované formy při každém vyhledávacím dotazu nebo analytickém procesování. Transport a zpracování událostí by mělo v každém případě proběhnout téměř v reálném čase, pakliže mají události přinést kýženou hodnotu z hlediska identifikace bezpečnostních hrozeb a ne pouze z pohledu následné forenzní analýzy. (Gordon, A., 2015)

2.2.3 Analýza dat

Po zpracování dat do podoby, kdy je možné s nimi dále pracovat nastupuje fáze automatizované analýzy v reálném čase. Ta může být založena na několika typech přístupu. Nejjednodušší variantou je analýza na základě události. Zdrojem takové události je zpravidla nějaký bezpečnostní produkt jako např. antivirus, který detekoval virus, nebo IDS, které detekovalo známou signaturu. Další možností je vyhledávání anomálií, kdy je aktuální dění v definovaném intervalu porovnáváno s dlouhodobým trendem. Příkladem může být počet otevřených spojení na jeden server, které výrazně překoná v krátkém intervalu běžný trend a může tak naznačit možný DoS či DDoS útok. Mezi takové lze zařadit například "ICMP flood", SYN Flood" nebo útoky na aplikační vrstvu jako např. "Low Orbit Ion Cannon". (McClure, S., Scambray, J., Kurtz, G., 2012) Dalším typem je analýza na základě podmínek či pravidel. Například když dojde k opakovanému selhání přihlášení uživatele na jedno či více zařízení během určité doby, je to vyhodnoceno jako pokud o prolomení hesla. Samozřejmě jsou možné i kombinace výše uvedených typů. V konečném důsledku je vždy výsledkem nějaký alarm, který je nutné dále prověřit člověkem.

2.3 SIEM platformy na trhu

Podle poradenské společnosti Gartner byla oblast SIEM nejrychleji rostoucí oblastí v rámci informační bezpečnosti. Její růst dosáhl meziročně 15,8% (Gartner Inc., 2016). V době psaní této práce existuje na trhu kolem dvou desítek různých SIEM platforem. Liší se jednak rozsahem poskytovaných komponent, tak i zaměřením. Především s ohle-

Obrázek 2.4: Garner Magic Quadrant 2017 (Gartner Inc., 2017)

dem na velikost organizace nebo předpokládaný objem událostí, které mají být paralelně zpracovávány.

Mezi lídry řadí společnost Gartner produkty Splunk Enterprise Security, IBM QRadar, LogRhytm Enterprise a McAfee Enterprise Security Manager. V předchozích letech byl v tomtéž kvadrantu i Micro Focus (dříve HP, resp. HPE) ArcSight, ale v poslední době jej konkurence předhonila. V rámci této práce se zaměříme především právě na Micro Focus ArcSight, jelikož ten je v současné době ve společnosti nasazen. Dále se seznámíme i se současnými lídry trhu, abychom mohli určit vhodnou strategii pro další nasazení a rozvoj řešení SIEM.

2.3.1 Micro Focus ArcSight

Historie produktů z rodiny ArcSight začíná v roce 2000, kdy byla založena původní firma ArcSight, která si dala za cíl poskytnout platformu na SIEM analýzu velkoobjemových dat. Později v roce 2010 byla společnost koupena firmou Hewlett-Packard a její produkty přešly pod značku HP. Následně bylo celé produktové portfolio ArcSight v rámci reorganizace HP vyčleněno do společnosti HP Enterprise. V roce 2017 pak byly některé produkty z portfolia HPE včetně produktů ArcSight prodány společnosti Micro Focus.

Hlavním produktem z rodiny ArcSight je ArcSight Enterprise Security Manager (zkráceně ESM), který je zodpovědný za analýzu přicházejících událostí v reálném čase, jejich vzájemnou korelaci a vytváření alarmů. ESM disponuje možností vytvářet komplexní pravidla, která umožní identifikovat podezřelé události s ohledem na jejich obsah či frekvenci. Lze zde také vytvářet tzv. "případy", které obsahují jeden či více alarmů, které je nutné dále řešit lidskou obsluhou. Aktuálně poslední verzí je verze 7.0, která byla uvolněna v roce 2018 a jako hlavní novinku nabízí možnost distribuovat korelaci událostí mezi více ESM. Krom serverové části je k dispozici konzole pro systémy Windows, MacOS, Linux a Solaris. Dále je k dispozici také webové rozhraní, které ale neposkytuje všechny funkce konzole.

Dalším stěžejním produktem je ArcSight Logger, který slouží především pro dlouhodobou správu a ukládání událostí. Ten disponuje pouze webovým rozhraním, ale díky možnosti paralelního vyhledávání napříč několika Loggery najednou dokáže výrazně rychleji poskytnout požadované údaje než je tomu u ESM, které má centralizované úložiště. Umožňuje také vytvářet nad událostmi reporty, které pak mohou být automatizovaně periodicky generovány.

Oba výše uvedené produkty jsou určeny především pro příjem událostí ve formátu CEF. ESM ani jiný nepřipouští. To je pevně definovaný otevřený formát, kde jednotlivé informace jsou přiřazeny nějaké položce, vzniká tedy pár klíč-hodnota. O toto přiřazování se stará produkt ArcSight SmartConnector. Jedná se v podstatě o agenta, který na vstupu dostává nebo získává data v nějakém (vetšinou) proprietárním formátu, ty dle mapovacích definic, tzv. "parserů", převede na jednotný formát CEF a následně odešle do ESM či Loggeru (nebo oběma).

2.3.2 Splunk Enterprise Security

Vznik společnosti Splunk se datuje do roku 2003 a to v San Francisku. U jejího zrodu stála trojice Michael Baum, Rob Das and Erik Swan. Již od založení se společnost zaměřila na analýzu real-time dat a jejich následnou vizualizaci pomocí webových technologií. Splunk jako produkt byl poprvé představen v roce 2007 a rovnou ve verzi 3.0, kdy byl představen jako nástupce předchozího produktu pod názvem "IT Search engine".

Nad původním produktem Splunk, jenž byl primárně cílen na analýzu operačních logů byl později v roce 2012 vystavěn produkt Splunk App for Enterprise Security 2.0 z něhož později vznikl Splunk Enterprise Security (ES), jak je nabízen v dnešní době. Jedná se o nadstavbu základní aplikace Splunk, umožňující právě bezpečnostní analýzu a korelaci tak jak, je využívána v rámci SIEM.

Splunk je také rozdělen do několika dílčích komponent. Předně jsou zde Splunk forwardery - Universal a Heavy. Ty zajišťují sběr dat a předání dalším komponentám. Základním rozdílem zde je, že Universal Forwarder slouží v podstatě jen k přeposílání událostí v neupravené, tzv. "RAW" formě. S těmi si dále musí poradit Indexer. Naproti tomu Heavy Forwarder dokáže události parsovat, čili jim "porozumět" a indexovat podstatná data, která posílá dál do systému spolu s původní událostí. Následuje Splunk Indexer nebo v případě větších implementací Splunk Indexer Cluster. Tato komponenta se stará o parsování logů (pokud tak již nebylo učiněno Heavy Forwarderem) a jejich indexaci.

Poslední klíčovou komponentou je Splunk Search Head nebo v případě nadstavby Enterprise Security pak Search Head cluster. Pro instalaci nadstavby Enterprise Security je totiž Search Head Cluster přímo vyžadován.

2.3.3 IBM Security QRadar

Původním tvůrcem SIEM produktu QRadar byla americká společnost Q1 Labs, jejíž vznik se datuje do již roku 2001. Později v roce 2011 se produkt díky akvizici dostal do portfolia gigantu IBM v rámci jeho výstavby divize bezpečnosti. Stejně jako v předchozím případě byla i společnost Q1 Labs od počátku zaměřena na síťovou bezpečnost a s ohledem na to vyvíjela produkt QRadar.

Z hlediska architektury je QRadar rozdělen do tří hlavních komponent. První dvojicí

jsou QRadar Event collector a QRadar Qflow connector. Zatím co první uvedený je zodpovědný za sběr jednorázových událostí, jako např. přihlášení do systému, druhý dokáže zpracovávat události, které mají delší trvání v čase. Typicky se jedná např. o události vázající se k jednomu konkrétnímu spojení HTTP nebo například události typu NetFlow, které poskytují routery a switche. Přicházející data jsou parsována pomocí zásuvných modulů označovaných jako DSM a normalizována, podobně jako je tomu v případě ArcSightu. QRadar ale využívá vlastní formát LEEF.

Další komponentou jsou QRadar Event Processor a QRadar Flow Processor. Zde probíhá analýza sebraných událostí, přičemž se zde využívá CRE, který jednak nabízí řadu předdefinovaných pravidel na identifikaci hrozeb a samozřejmě také možnost vytvoření pravidel vlastních. Poslední komponentou je pak QRadar Console, která slouží jednak pro uživatelský přístup k celému systému a také pro administraci jednotlivých komponent QRadaru.

2.3.4 LogRhythm Enterprise

Společnost LogRhythm stojící za SIEM řešením LogRhythm Enterprise (původně jen LogRhythm) byla založena v roce 2003 právě s cílem nabídnout produkt pro správu událostí. Za jejím založením stojí Chris Petersen a Phillip Villella a její původní název byl Security Conscious, Inc. Později v roce 2015 se společnost přejmenovala právě na LogRhythm.

I v případě LogRhytm Enterprise je produkt rozčleněn do několika dílčích komponent. Na samém počátku je Data Collector, který existuje ve formě appliance a také jako softwarový produkt. Data Collector je zodpovědný za sběr událostí z rozličných systému a zdrojů všude tam, kde lze použít sběr bez agenta na straně zdrojového systému. Typicky se tak jedná například o události sbírané přes API nebo přicházející např. protokolem syslog. Pro monitoring koncových bodů (stanic, serverů) je možné využít LogRhythm SysMon, který funguje jako agent a sbírá události lokálně. V současné době jsou podporovány platformy MS Windows, Linux, HP-UX a AIX. Posledním typem kolektoru je Network Monitor, který je možné použít pro analýzu paketů, či zprostředkování odchycení paketů pro další analýzu.

Další důležitou komponentou je Data Processor. Ten dostává data z kolektorů a pro-

vádí parsování událostí, normalizaci, kategorizaci a doplnění o další informace. Výsledný syntetizovaný záznam je nazýván Machine Data Intelligence (MDI) a zahrnuje řadu metadat popisující původní data. Procesování událostí probíhá distribuovaně a Processor je možné škálovat horizontálně i vertikálně. Události jsou Processorem archivovány v jejich původní formě a také dále předány ve strukturované i původní formě další komponentě a sice Data Indexeru. Tato komponenta opět umožňuje škálování a využívá Elasticsearch k vytvoření platformy pro rychlé vyhledávání a zajištění vysoké dostupnosti dat. Indexer umžňuje vyhledávání jak za pomocí MDI, tak nestrukturovaně v původních událostech. Data Indexer je nicméně volitelnou součástí. Je možné události z Data Processoru směrovat přímo do hlavní komponenty AI Engine.

AI Engine je hlavní komponentou tohoto SIEM řešení. Jeho úkolem je provádět automatizovanou analýzu událostí, které obdrží od Data Processoru v reálném čase. Na základě vyhodnocení pak případně vytváří alarmy. Ty jsou pak následně spravovány v poslední klíčové komponentě - Platform Manager. Ten se stará jak o správu alarmů, tak o reporty či přehledy správu událostí a případů. S tímto systémem pracují bezpečnostní analytici.

2.4 SIEM ArcSight komponenty

2.4.1 SmartConnector

Jak již bylo předesláno, ArcSight SmartConnector je klíčovou komponentou nutnou pro porozumění obsahu událostí a jejich konverzi z nativního formátu do vnitřně používaného formátu CEF. Konektor je v podstatě softwarová aplikace naprogramovaná v jazyce Java, díky čemuž je poměrně multiplatformní. Konektor se vnitřně dělí na kontejner (Container) a samotný konektor (Connector) někdy nazývaný také agent. Kontejner je jakýsi Framework, který následně umožňuje spustit jeden či více konektorů. Ty se pak ale dělí o prostředky dedikované kontejneru, především o paměť (Java Heap Memory), jejíž alokace je staticky definována. Na toto rozdělení je potřeba pamatovat i při případné konfiguraci, jelikož jsou konfigurační parametry, které ovlivňují chování celého kontejneru a vedle toho jsou konfigurační parametry specifické pro konkrétní konektor. Obvyklou

praxí je, že v rámci jednoho kontejneru je hostován pouze jeden konektor. Jedním z konfiguračních parametrů celého kontejneru je například deklarace portu užívaného pro jeho administraci. Ten je následně využíván například aplikací ArcSight Management Center pro centralizovanou správu a monitoring nebo ArcSight Load Balancerem v případě syslog konektorů.

ArcSight SmartConnector je postaven na univerzálním frameworku. Hovoříme-li o různých specifických konektorech, jedná se stále o jeden identický produkt, ale v odlišné konfiguraci pro specifický účel, resp. typ zdroje. Konektor obecně plní dvě hlavní role. Předně nabízí řadu možností jak získat události. Může například naslouchat na nějakém UDP či TCP portu pro příjem událostí ve formátu syslog. Umožnuje číst lokální soubory na disku, připojit se k databázi přes rozhraní JDBC a spustit nějaký dotaz, získat událost skrz webové REST API ve formátu JSON či XML nebo získávat události pomocí Windows Eventing API v případě MS Windows. Druhou rolí je pak parsování, resp. překlad události z původního formátu do formátu CEF. Náročnost překladu závisí na formátu získaných dat. V případě JSON či XML nebo databází jsou již mají data nějakou pevnou strukturu, která je charakterizována dvojicemi klíč-hodnota. Formát CEF také využívá struktury klíč-hodnota, ale klíče jsou definované pevně ve specifikaci formátu. (Micro Focus, 2019) V takovém případě je překlad z původního formátu do formátu CEF v podstatě jen o přemapování původních klíčů těmi, které jsou definovány v CEF formátu a jejich naformátování v souladu se syntaxí CEF formátu.

```
Jan 18 11:07:53 host CEF:Version|Device Vendor|Device Product|Device
Version|Device Event Class ID|Name|Severity|[Extension]
```

Obrázek 2.5: CEF formát se syslog hlavičkou. Zdroj: Micro Focus

Události v CEF formátu mohou být buď samostatné, například pokud jsou logovány do souboru nebo mohou být součástí syslog zprávy, kdy je před samotnou zprávu v CEF formátu přidána syslog hlavička (viz. obr. 2.5). Samotný CEF formát se pak skládá také z hlavičky, kde jsou deklarovány hodnoty jako výrobce, produkt, verze zařízení nebo jméno události. Za hlavičkou následuje sekce s rozšířenými informacemi, kde se nacházejí zmiňované dvojice klíč-hodnota jak je vidět na obrázku č. 2.6.

Ne vždy jsou však zdrojová data dostupná v nějaké strukturované formě. To platí pře-

```
Sep 19 08:26:10 host CEF:0|Security|threatmanager|1.0|100|worm successfully
stopped|10|src=10.0.0.1 dst=2.1.2.2 spt=1232
```

Obrázek 2.6: Příklad CEF události. Zdroj: Micro Focus

devším o událostech přicházejících prostřednictvím protokolu syslog. Zde je nutné provést parsování za pomoci regulárních (regex) výrazů, kdy se konektor dle dostupného souboru regex výrazů snaží určit o jaký typ události se jedná a vhodně prezentovat data a převést je do formátu CEF. Je-li parsování úspěšné, je celá událost převedena do formátu CEF, přičemž obsahuje informace o výrobci zdrojového zařízení, produktu a další informace vyextrahované z původní události. Ne vždy je však parsování úspěšné. Jelikož jsou využívány regex pravidla může se stát, že událost je částečně rozpoznána - např. je správně určen výrobce a produkt, ale samotnému obsahu zprávy nebylo porozumněno. V takovém případě nová událost v CEF formátu obsahuje správné pole s označením dodavatele a produktu, ale v poli jméno (name) obsahuje řetězec "Unparsed Event" a v poli zpráva (message) obsahuje tělo syslog události v původní formě jako jeden řetězec. Pakliže není SmartConnector schopen rozpoznat událost vůbec, je vygenerována nová CEF událost s obecným určením výrobce a produktu jako "Unix", přičemž obě pole zpráva i jméno obsahují událost v její původní formě. V extrémním případě, kdy obdržená událost ani nerespektuje formátování pro syslog nedojde ani k tomu a celá událost je v původní formě v poli zpráva.

Konektor také nabízí několik možností jak události dále doručit, přičemž vždy musí být stanoven alespoň jeden cíl. Na výběr jsou následující cíle:

- ArcSight Manager - Poskytuje obousměrnou šifrovanou komunikaci s ArcSight ESM využívající port TCP/8443 a API ESM, které konektoru umožňuje doučení událostí. Skrz ArcSight ESM je naopak možné měnit konfigurační parametry tohoto cíle, jako např. agregaci, či filtrování, či případně provést aktualizaci konektoru na vyšší verzi.

- ArcSight Logger SmartMessage - Poskytuje jednosměrnou šifrovanou komunikaci s ArcSight Loggerem, obvykle přes port TCP/9000 nebo TCP/443, využívající Logger API. V případ tohoto cíle je pouze možné přenášet události z Connectoru do Loggeru.

- ArcSight Logger SmartMessage Pool - Jedná se prakticky o stejný typ cíle jako předchozí s tím rozdílem, že lze definovat více Loggerů jako cíl (tzv. "pool"). Události jsou pak při doručování rovnoměrně rozděleny mezi všechny dostupné Loggery.
- CEF File - Umožňuje zápis událostí ve formátu CEF do souboru v počítači hostícím konektor.
- Event Broker - Zasílání událostí do tématu v ArcSight Event Brokeru.
- CEF syslog - Zasílání událostí ve formátu CEF prostřednictvím protokolu syslog (TCP/UDP/TLS).
- CEF Encrypted syslog (UDP) - Zasílání šifrovaných událostí pomocí protokolu syslog, přičemž se pro příjem předpokládá opět vhodně nastavený ArcSight Connector. V tomto případě je použito symetrické šifrování na bázi sdíleného klíče.
- CSV File - Parsované události jsou ukládány do lokálního textového souboru ve formátu CSV.
- RAW syslog - Umožňuje přeposílání událostí pomocí syslog protokolu v původní nezměněné formě.

(Micro Focus, 2019)

Pro každý typ cíle lze na konektoru definovat filtrování CEF událostí. Filtr přitom může využívat všech standardně definovaných klíčů, které CEF protokol zná. Kromě filtrace je možná i agregace. V případě agregace se definují klíče, které se musí schodovat u více událostí. Pokud pak konektor za daný časový úsek přijme více takto shodných událostí, dojde k jejich spojení v jednu událost, která je doplněna o počet výskytů. Typickým příkladem je třeba vícečetné neúspěšné přihlášení z jednoho počítače na jeden server při použití stejného uživatelského jména. Byť je pro každý pokus vygenerována samostatná událost, liší se nejspíše pouze časovým razítkem, případně nějakým sekvenčním číslem události. V takovém případě má smysl takové události agregovat, jelikož každá další nepřináší žádnou přidanou hodnotu a pouze spotřebují více místa v úložišti.

Kromě filtrování událostí směrem k cílům je zde také možné filtrovat události na vstupu. K tomu je určena funkcionalita aplikující regex filtry na vstupující události v

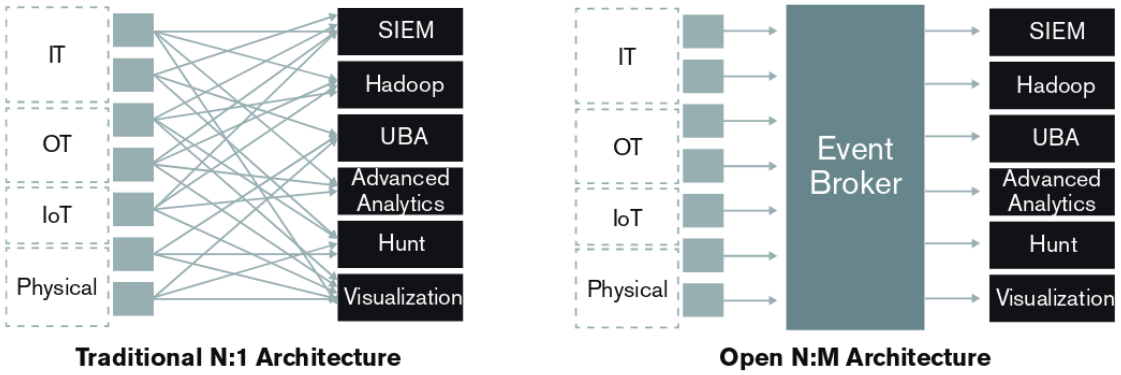
jejich původním formátu. Je však možné specifikovat pouze jeden regex a to buď inkluzivní a nebo exkluzivní. Byť lze jedním regex filtrem s využitím logického OR obsáhnou více vzorků, je toto řešení z hlediska správy ne zrovna přívětivé.

Vedle výrobcem předpřipravených konektorů nabízí SmartConnector i variantu FlexConnector. Je to opět totožný produkt, ale na místo předpřipravené konfigurace je možné zvolit manuální definici parseru. Hlavní nevýhodou předpřipravených konektorů je, že použité parsery nejsou dostupné, čili není možné je jednoduše rozšířit či modifikovat v případě potřeby a zákazník se tak buď musí spolehnout na podporu výrobce, že požadované rozšíření doplní a nebo musí vytvořit vlastní FlexConnector, kdy se ale může dostat do situace že bude muset znovu nadefinovat parser i pro jinak fungující typy událostí, pakliže nechce či nemůže data sbírat nezávisle dvěma konektory. U předpřipravených konektorů je sice možnost tzv. "parser overide", kdy je možné docílit úpravy v dílčím chování parseru, ale bez znalosti jak přesně parser vypadá jsou možnosti využití bez spolupráce s výrobcem poměrně omezené.

2.4.2 Load Balancer

ArcSight Load Balancer má sloužit jako mezivrstva primárně pro události doručované prostřednictvím protokolu syslog, byť jej lze využít i pro soubory načítané skrze FTP. Jeho úlohou je přijímat takto doručované události a distribuovat je skupině konektorů podle zadané metody. Metod je k dispozici několik, včetně obvyklé round-robin, ale z hlediska následné agregace na konektorech je ideální metoda nazvaná "Aggregation Preferred". Jedná se v zásadě o celkem obvyklou metodu balancování na základě zdrojové IP adresy, kdy jsou události pocházející z jednoho zdroje doručovány vždy jednomu konektoru. To je kýžený stav, pakliže chceme, aby konektory co nejlépe agregovaly podobné události. Avšak na rozdíl od konvenčních load balancerů bere ArcSight Load Balancer v úvahu, že se jedná o syslog události, tedy o jednosměrný tok dat. V kombinaci s další jeho funkcionalitou, kdy neustále monitoruje konektory prostřednictvím management portu a má tak informaci o jejich vytížení, mu to umožňuje změnit pro určitý zdroj cílový konektor, pakliže ten je momentálně přetížen.

Pakliže se soustředíme na syslog protokol, nabízí ArcSight Load Balancer podporu jak UDP, tak TCP a to včetně varianty TLS. Není však možné na load balanceru pro-



Obrázek 2.7: Tradiční model vs. Event Broker

tokol změnit a oba protokoly mají interně vlastní frontu. Load balancer nabízí kromě režimu samostatné instance i režim vysoké dostupnosti, bohužel však pouze v režimu aktivní/pasivní. Navíc neumožňuje změnu konfigurace takového HA clusteru a ta je sdílena mezi oběma instancemi v původním stavu. Pro aplikaci změny konfigurace je tak nutné úplné vypnutí clusteru, aby došlo k načtení nové konfigurace. To by snad mělo být vyřešeno v nové verzi produktu. Nicméně pro zpracování velkého objemu událostí není tento HA model ideální.

2.4.3 Event Broker/Transformation Hub

V roce 2016 představilo (Hewlett Packard Enterprise, 2016) tehdy ještě HPE nový model pro zpracování událostí nazvaný ArcSight Data Platform, zkráceně ADP. Oproti tradičnímu modelu přichází nový ADP model s otevřeným řešením na sdílení událostí i mimo platformu ArcSight (viz. obr. 2.7). Mezi konektory a cílové systémy je zde vložena další vrstva v podobě produktu ArcSight Event Broker. Event Broker je v podstatě Apache Kafka zabalená v kontejnerové platformě Kubertenes (Micro Focus, 2018). Cílovým systémem tak může být v zásadě jakákoli aplikace, umožňující registraci v Apache Kafka tématu a podporující CEF formát. Event Broker je koncipován jako vysoce škálovatelné řešení s vysokou dostupností. V nedávné době byl Event Broker přejmenován na Transformation Hub.

Event Broker centralizuje tok událostí a umožňuje jejich směrování k různým cílovým aplikacím (konzumentům). K tomu využívá systém témat (topics), které umožní oddělit jednotlivé skupiny událostí a zpřístupnit je jen vybraným konzumentům. Event Broker je

koncipován jako vysoce škálovatelné řešení s vysokou dostupností. V rámci Event Brokeru rozlišujeme dva typy kontejnerů, resp. uzlů. Jsou tu řídící uzly (master nodes) a pracovní uzly (worker nodes). Pracovní uzly jsou zodpovědné za příjem událostí, zatím co k řídícímu uzlu se připojují koncové aplikace konzumující události. Aby mohl Event Broker zajistit vysokou dostupnost, pracuje v režimu clusteru, přičemž jsou vyžadovány alespoň tři pracovní uzly. V případě řídících uzlů je možné mít pouze jeden, což ale není doporučeno pro produkční nasazení. Pro zajištění vysoké dostupnosti i směrem ke konzumentům je doporučený model s více řídícími uzly, kde je pevně stanovený počet uzlů tři, přičemž jsou připojení balancována za pomocí virtuální IP.

Události ze zdrojových zařízení lze do Event Brokeru posílat několika způsoby. První možností je využít ArcSight Connectory, které od verze 7.6 nabízejí i možnost volby Event Brokeru jako možné destinace. Dále je možné využít ArcSight Collectory vyvinuté primárně pro Event Brokery. Poměrně nově zde existuje také možnost zprovoznit vybrané Connectory přímo v Event Brokeru - tzv. Connectors in Event Broker (CEB). V současné verzi 2.21 je podporován v režimu CEB pouze syslog Connector, přičemž jeden Event Broker cluster může hostovat až 50 Connectorů (Micro Focus, 2018). Toto řešení má poskytnout lepší škálovatelnost, jelikož Event Broker se postará o rozdělení událostí mezi dostupné CEB Connectory. CEB Connectory však na rozdíl od klasických Connectorů mohou data doručovat jen do témat v Event Brokeru.

Ve výchozím stavu disponuje Event Broker třemi předkonfigurovanými tématy, které však lze v případě potřeby rozšířit do další vlastní témata, kam mohou Connectory doručovat události. V základní sadě jsou zde témata eb-cef, sloužící pro události od Connectorů v CEF formátu, dále eb-esm, jež přijímá události v binární formě a slouží výhradně pro přenos dat k ArcSight ESM a také téma eb-con-syslog, jež je vyhrazené pro události ve formátu syslog od Collectorů, které mají být zpracovány syslog CEB Connectory Event Brokeru.

2.4.4 Logger

Hlavním posláním Loggerů je zajistit uložení a archivaci událostí, dále pak rychlé vyhledávání a vytváření reportů. Jedná se o robustní aplikaci s webovým rozhraním. Právě přes webové rozhraní dochází k veškeré interakci s uživateli, konektory a případně

dalšími Loggery a to v rámci jednoho WebAPI. Logger využívá nativní engine pro MySQL pro uchovávání událostí, přičemž se rozlišují dva typy úložišť. Online úložiště, kdy jsou události uložené v indexované databázi uchovávané lokálně a offline úložiště v archivech, které mohou být deponovány na vzdáleném úložišti jako například NFS. Offline úložiště je tak z principu pomalejší při prohledávání událostí na rozdíl od online úložiště, byť je možné jednotlivé archivy dodatečně indexovat. Funkce archivace je dostupná na úrovni aplikace včetně automatického mazání starých archivů, přičemž archiv se vždy vytváří za předchozí celý den. V tomto kontextu je potřeba zmínit způsob, jakým Logger pracuje s událostmi pakliže jde o datum a čas, jelikož při vyhledávání i archivaci se řídí časem doručení události Loggeru, byť by se z praktického hlediska spíše nabízelo jako vhodnější primárně řadit události podle času jejich vzniku.

Vyhledávání v Loggeru funguje na principu definice dotazů pomocí poměrně jednoduchého dotazovacího jazyka. Obecně tento jazyk definuje klíče, které vycházejí z klíčů CEF formátu a nabízí řadu operátorů jako například "=", "!=", "CONTAINS", "STARTSWITH" nebo "INSUBNET" pro validaci operandů. Nadto je možné podmínky řetězit s využitím logických operátorů "OR", "AND" a "NOT". (Micro Focus, 2019) Vedle toho je možné použít i fulltextové vyhledávání. Výsledky je možné ještě dále filtrovat a upravovat např. do grafického znázornění dle zvoleného klíče. Logger také nabízí funkci distribuovaného vyhledávání, kdy lze prostřednictvím jednoho Loggeru hledat na více nebo všech Loggerech najednou. Další funkcí je pak přeposílání událostí dle zadaných kritérií dalším aplikacím. Předně je to ArcSight ESM, kdy může Logger fungovat v rámci hierarchického návrhu jako další filtr a přeposílat ESM pouze takové události, které mají z pohledu bezpečnosti nějakou informační hodnotu. Dále je možné přeposílat události jako běžný text ve formátu CEF zabalené do syslog hlavičky a to jak pomocí UDP, tak i TCP.

Jedním z omezení Loggeru je, že zde limit 250 maximálního počtu současných připojení. To zahrnuje jak připojení od konektorů, kterým přicházejí události, tak připojení obsluhy nebo dalších Loggerů v případě distribuovaného vyhledávání. Navíc některé typy konektorů, které přenášejí velké množství událostí, vyžadují nastavení využívání více vláken k přenosu dat a tedy i více připojení k Loggeru.

2.4.5 ArcSight ESM

ArcSight ESM je hlavním systémem pokud jde o identifikaci bezpečnostních hrozeb. Jedná se o ještě robustnější balík aplikací než je Logger. Vlastně upravený Logger je jednou z interních komponent ESM sloužící jako úložiště. Hlavní komponentou je ale korelační engine, který umožňuje vytváření komplexních pravidel sloužících pro realtime analýzu příchozích událostí a jejich případnou korelaci. Příchozí události jsou označovány jako základní události (base events). V případě, že nějaká příchozí událost nebo více příchozích událostí odpovídá definovanému pravidlu, je vygenerována další událost, která se nazývá korelovanou událostí (correlation event). Ta již neobsahuje všechny pole tak jak byly v původní základní události, ale pouze vybraná pole tak, jak byly určeny v pravidle. Je ale na původní základní událost či události navázána a je tak možné je zobrazit.

Vedle korelačních pravidel existuje v ESM celá škála dalších objektů. Jsou zde k dispozici tzv. "Active Lists", které mohou být využívány v rámci pravidel jako jakési dočasné přehledy. Například je možné uchovávat seznam všech uživatelů, kteří se přihlásili do nějakého systému za určitý časový rámec. Dále je zde možné vytvářet trendy, různé přehledy a podobně. Nechybí ani správa bezpečnostních incidentů.

ESM je možné provozovat v hierarchické struktuře. K tomu slouží speciální konektor nazývaný SuperConnector, který je napojen na oba systémy ESM, zdrojový a cílový a umožňuje obousměrnou komunikaci. Předávané události je možné omezit filtry, díky čemuž je možné předávat do nadřazeného ESM např. pouze korelované události bez základních událostí, přičemž relevantní základní události jsou předány až dodatečně na vyžádání, pakliže je chce na nadřazeném ESM někdo zobrazit.

V poslední verzi 7.0 přibyla možnost provozovat ESM v distribuovaném režimu, kdy korelační engine běží na pozadí paralelně na více strojích. Další možností je také provoz ESM v režimu vysoké dostupnosti pomocí modulu ESM HA. Při dimenzování hardware pro ESM je nutné brát v potaz i velmi vysoké IOPS.

2.4.6 ArcSight Management Console

ArcSight Management Console (zkráceně ArcMC) je velmi užitečná při každodenní správě všech konektorů. Umožňuje spravovat a monitorovat všechny připojené konektory.

To se děje přes předem nastavený administrativní port každého kontejneru. ArcMC umožňuje detailní konfiguraci snad všech parametrů jak kontejneru, tak konektoru a to včetně možnosti vzdáleně jej restartovat či aktualizovat. Nabízí také možnost monitorovat některé parametry konektorů, například stav cache, EPS a další. Nenabízí ale bohužel příliš individuální nastavení notifikací a při velkém počtu konektorů nejsou udávané informace a stavy často zcela aktuální.

Kromě správy konektorů umožňuje ArcMC v omezené míře spravovat i další komponenty jako Logger, Event Broker, Connector Appliance, ArcSight Collector či další ArcMC. V případě ArcMC se ale nedá hovořit o možnosti hierarchického uspořádání. ArcMC může pouze konfigurovat aplikační nastavení "podřízené" ArcMC a monitorovat její stav, ale nemá přístup ke konfiguraci nebo monitorování dalších komponent připojených k takové "podřízené" ArcMC.

2.5 Technologie

2.5.1 Syslog

Syslog protokol je de facto standard využívaný pro logování drtivou většinou systémů na bázi Unix/Linux a je též velmi často užívaný síťovými zařízeními, jako jsou routery, firewally a podobně. Syslog protokol je definován ve dvou RFC standardech. Starší RFC3164, někdy také nazývaný BSD-syslog, je datován rokem 2001, ale samotný protokol byl vytvořen již na začátku 80 let(O'Reilly, 2019). V této starší variantě je pro přenos definován pouze protokol UDP s doporučeným číslem portu 514. Samotný formát je prostý text, kdy jeden UDP datagram nese jednu syslog zprávu. Základní struktura zprávy se pak skládá z priority, hlavičky a těla zprávy. Priorita (PRI) je reprezentována číslem ohraničeným znaménky menší než a větší než, přičemž numerická hodnota je odvislá od zdroje zprávy, např. emailový server či jádro OS, a její závažnosti. Hlavička potom nese časové razítko a jméno případně IP adresu zdrojového systému (hosta). Formát časového razítka je poměrně triviální, omezuje se jen na den, měsíc, hodinu, minutu a vteřinu. Tělo zprávy se pak skládá z informace o procesu (tag), který danou zprávu vygeneroval a samotné textové informace popisující událost.

Obrázek 2.8: Syslog zpráva ve formátu RFC3164. Zdroj: vlastní

V roce 2009 byl uveřejněn nový syslog standard RFC5424, někdy nazývaný též IETF. Ten přinesl několik zásadních změn. Z hlediska přenosu nedošlo v rámci tohoto RFC k žádné změně a nadále uvažuje pouze UDP. Nicméně v souběhu je publikován i RFC5425 definující posílání syslog zpráv pomocí TCP za využití TLS k šifrování. Ten je však především nadstavbou RFC5424 a nepředpokládá doručování bez šifrování. V praxi je však možné setkat se s implementací zasílání událostí formátovaných dle stašího RFC3164 za pomocí TLS šifrování nebo naopak zasílání událostí dle obou RFC pomocí nešifrovaného TCP. V prvním případě se dá hovořit pouze o jakémsi rozšíření RFC5425, jelikož samotný TCP přenos a TLS šifrování nemají vliv na formát hlavičky syslog protokolu dle některé ze dvou specifikací. V případě druhém je pak situace poněkud nepřehledná. Lze narazit na RFC6587, který popisuje posílání událostí skrze nešifrovaný TCP protokol, nicméně je z hlediska bezpečnosti nedoporučovaný. Vedle něj zde ještě existuje starší RFC3195, který také řeší doručování syslog událostí prostřednictvím nešifrovaného TCP.

Hlavní změnou v případě RFC5424 je formát samotné zprávy. Jednak je zavedeno verzování samotného syslog formátu přímo ve zprávě. Podstatným rozdílem je i redefinice formátu časového razítka, které vychází ze standardu RFC3339, přičemž umožňuje jít v detailu až k mikrosekundám. Zároveň zahrnuje rok a také informaci o časové zóně, respektive posunu vůči UTC. Dále zavádí několik dalších částí hlavičky jako je MSGID či PROCID, který nahrazuje původní tag. Kromě standardního nestrukturované informace v těle zprávy přichází i s alternativní možností strukturované informace ve formátu "klíč=hodnota".

I přes fakt, že nový standard RFC5424 je již 20 let starý, řada výrobců síťových zařízení standardně upřednostňuje starší RFC3164. Jeho hlavní nedostatek, totiž ne příliš precizní formát časového razítka pak často různě upravují přidáním např. roku či časové zóny nebo milisekund. Byť to může být považováno za užitečné rozšíření při kontrole logů

lidskou obsluhou, zapříčiňuje to značné problémy při automatickém zpracování. Jelikož syslog zpráva je v podstatě textový řetězec, je k jejímu pochopení nutné provést parsování, povětšinou s využitím regulárních výrazů. Pakliže výrobce nedodržuje RFC, může nastat problém v porozumění obsahu syslog zprávy a chybné interpretaci jednotlivých částí. Poměrně často se vyskytujícím problémem je interpretace přidaného roku jako jména zdrojového systému. Bohužel ani u výrobců, kteří umožňují přepnutí zařízení do novějšího standardu RFC5424 není vždy vyhráno. Řada výrobců má v implementaci RFC5424 ve svých produktech chyby a výsledný formát zpráv není zcela v souladu se standardem. Adopce strukturovaného formátu informace je pak zcela ojedinělý jev.

Ukázka zprávy dle RFC5424 obsahující pouze nestrukturované informace (zdroj: vlastní):

```
<165>1 2019-04-24T14:22:33.000003-01:00 myserver.mydomain.local myproc 789
- - User john.doe logged out.
```

Ukázka zprávy dle RFC5424 obsahující kombinaci strukturované a nestrukturované informace (zdroj: vlastní):

```
<165>1 2019-04-24T14:22:33.000003-01:00 myserver.mydomain.local myproc 789
ID1 [mySDID@123 user="john.doe" action="logout"] User john.doe logged out.
```

Ukázka zprávy dle RFC5424 obsahující pouze strukturované informace (zdroj: vlastní):

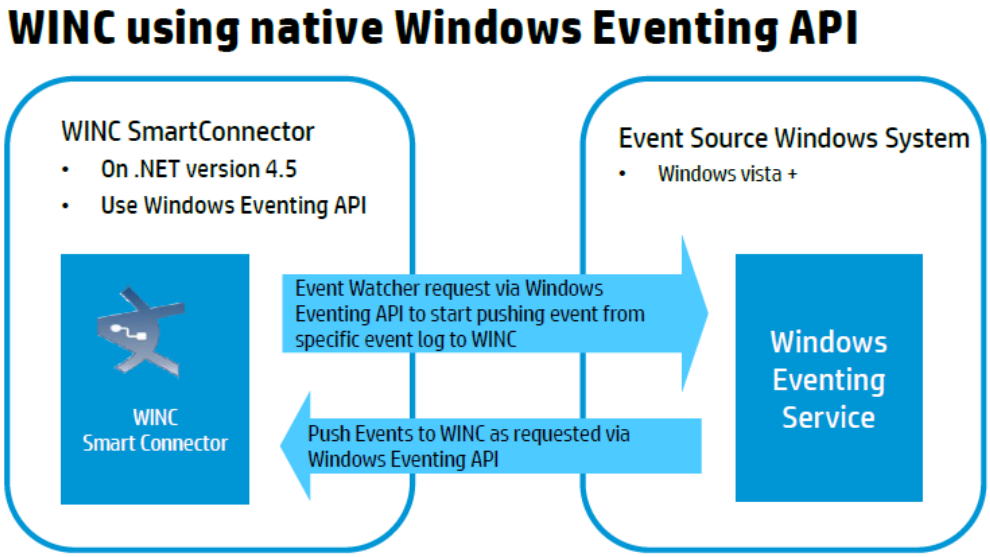
```
<165>1 2019-04-24T14:22:33.000003-01:00 myserver.mydomain.local myproc 789
ID1 [mySDID@123 user="john.doe" action="logout"]
```

Ukázka stejné zprávy dle RFC3164 (zdroj: vlastní):

```
<165>Apr 24 14:22:33 myserver myproc[789]: User john.doe logged out.
```

2.5.2 MS Windows Event Log

Pokud jde o sběr událostí ze systémů rodiny Microsoft Windows, je situace zcela odlišná než tomu bylo v případě syslogu. ArcSight nabízí hned dva typy konektorů. Starší nazvaný Windows Unified Connector (WUC) využívá knihovnu JCIFS, přes kterou je schopen získávat události uložené v souborech evtx, ve kterých MS Windows běžně ucho-



Obrázek 2.9: WiNC konektor a WE API (Micro Focus, 2015)

vávají události (Microsoft, 2018). Jeho nevýhodou je jednak podpora pouze protokolu SMBv1 a především pak celková nespolehlivost tohoto konektoru. Běžná konfigurace spočívá v nadefinování seznamu serverů, které konektor postupně kontaktuje a stahuje soubory etv/etvx s událostmi. Konektor využívá hned deset vláken pro stahování souborů s událostmi paralelně. Pokud je ale seznam serverů rozsáhlejší, což v praxi bývá i několik desítek, jsou servery kontaktovány postupně po deseti. V kombinaci s možnými problémy s dostupností některých serverů nebo chybách a vypršení spojení při přenosu to znamená časté problémy s dostupností logů. Novějším typem konektoru je Windows Native Connector (WiNC). Ten využívá k získávání událostí nativního Windows Eventing API. Na rozdíl od WUC, který defacto pracuje v režimu "poll", nový WiNC konektor funguje v režimu "push". Konektor se skrze API zapíše (subscribe) u serveru, ze kterého má probíhat sběr událostí a ten mu následně zasílá jím požadované události (viz. obr. 2.9). Tím je eliminován původní hlavní problém WUC konektoru.

Díky využití Windows Eventing API je navíc možné využít pro centralizaci událostí další nativní technologii systému MS Windows a to Windows Event Forwarding (WEF). Ta umožňuje nadefinovat skrze doménové politiky vybrané servery, tzv. Windows Events Collector (WEC), kam budou ostatní servery přeposílat své události. Z těchto WEC serverů pak můžou být události přeposílány právě WiNC konektorům. WiNC konektor pak nemusí být vybaven seznamem serverů a zpracovává všechny události ze všech serverů,

které byly doručeny danému WEC serveru. Zde je však potřeba vyřešit rozdílnou výkonnost WEC a WiNC, kdy u WEC se uvádí až 9 tisíc událostí za vteřinu, zatím co doporučená propustnost u WiNC je uváděna 1500 událostí za vteřinu.

3 Praktická část

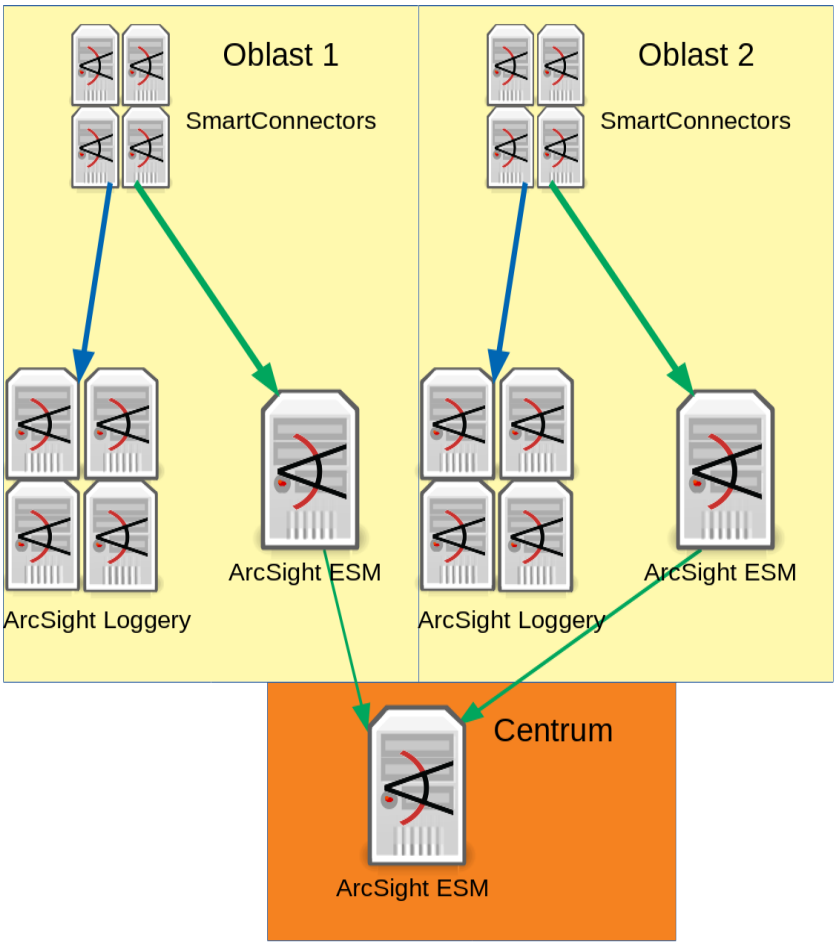
3.1 Stávající design

3.1.1 Obecný návrh

V současné době společnost využívá řešení Micro Focus ArcSight. Přičemž návrh implementace celého řešení byl vytvořen před více než šesti lety, kdy se toto řešení zavádělo a klíčová aplikace ArcSight ESM byla v té době ve verzi 5.2. V průběhu času došlo sice k dílčím vylepšením, ale celá architektura stále vychází z původního konceptu. To s sebou přináší řadu problému, ať už z hlediska škálovatelnosti řešení nebo z hlediska dostupnosti. Vedle možnosti přepracovat architekturu stávající implementace, je druhou uvažovanou možností kompletní výměna za zcela jiné řešení. Důvodem zde je především poměrně nízká pružnost dodavatele s integrací různých produktů především z dynamicky se rozvíjejícího trhu cloudových služeb.

Společnost využívá v současné době pět produktů z portfolia ArcSight. Jedná se o ArcSight ESM, ArcSight Logger, ArcSight ArcMC a ArcSight SmartConnector a ArcSight Load Balancer (viz. obr. 3.1). V současném návrhu je využitý hierarchický model, kdy je vymezeno několik oblastí. V každé oblasti jsou pak instalovány všechny komponenty včetně ArcSight ESM. Oblasti tak fungují autonomně (viz. obr. 3.2). ArcSight ESM v jednotlivých oblastech provádí analýzu událostí dle shodného souboru pravidel. Jelikož jsou ale správou bezpečnostních hrozeb a incidentů pověřeny týmy s globální působností, jsou události, které jsou v oblastních ArcSight ESM vyhodnoceny jako škodlivé či podezřelé zasílány do centrálního ArcSight ESM, kde jsou dále prozkoumány obsluhou a je zde také kontrolován průběh práce na jednotlivých incidentech.

Události jsou sbírány jednotlivými konektory a předávány jak oblastnímu ESM, tak oblastním Loggerům. Těch je v každé oblasti tři až pět. Loggery primárně zajišťují archivaci dat po vyžadovanou dobu. To vychází jak z interních nařízení společnosti, tak ze zákonných požadavků. Oproti ESM také dostávají všechny typy událostí. V případě ESM jsou na konektorech definovány filtry, které vyfiltrují některé typy nepotřebných událostí. Naneštěstí jsou díky prvotnímu návrhu všechny filtry koncipovány jako exkluzivní, tedy

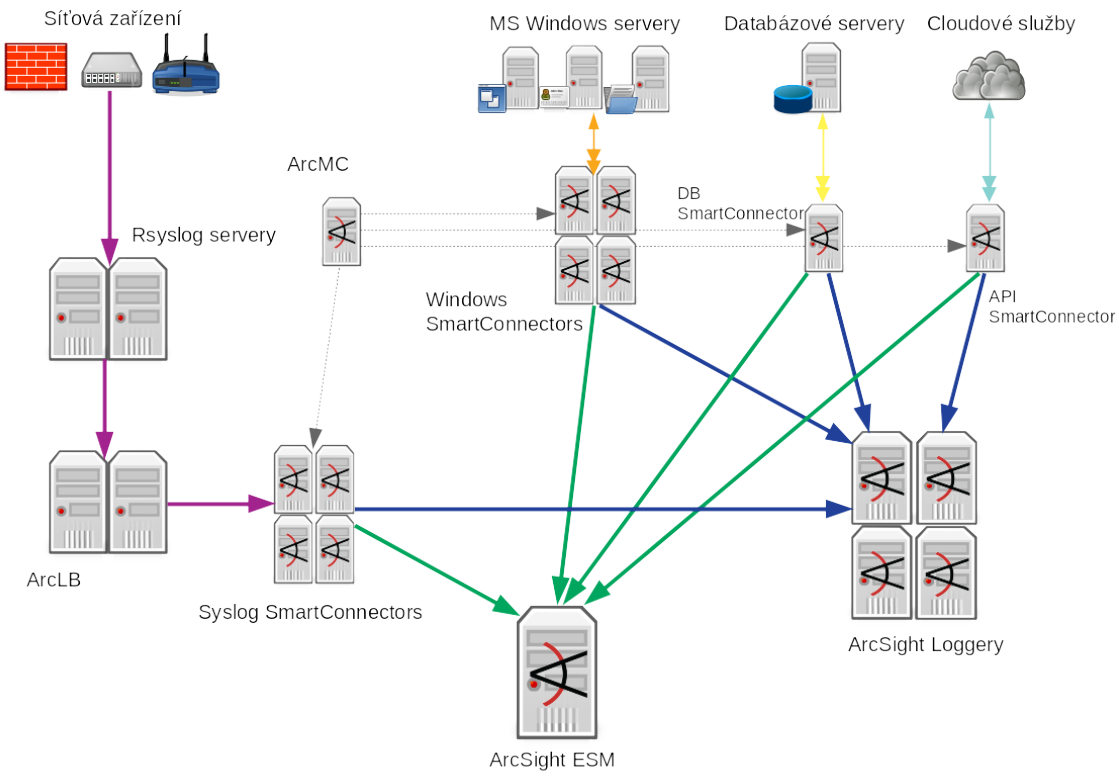


Obrázek 3.1: Hierarchická infrastruktura (zdroj: vlastní)

definují pouze vybrané události, které nebudou na ESM doručeny. To v praxi znamená, že jsou odfiltrovány pouze některé neužitečné události, které je možné identifikovat nějakým jednotícím vzorcem. Obecně je doporučována spíše opačná metodika, čili inkluzivní filtry, kdy jsou na ESM zasílány pouze události, pro které existuje nějaké pravidlo resp. použití.

3.1.2 Sběr událostí pomocí protokolu syslog

Současný návrh sběru událostí ze zařízení podporující syslog protokol je poměrně složitý a skládá se z více úrovní. To je dáno historickým vývojem. V každé oblasti je umístěna dvojice load balancerů. Ty jsou nakonfigurovány v režimu vysoké dostupnosti (aktivní/pasivní) a hostují virtuálních IP adresu, která slouží jako cílová IP adresa pro směrování syslog UDP datagramů ze zdrojových zařízení. Tato virtuální IP adresa má přiřazeno několik reálný IP adres hostovaný na koncových serverech. Na ty jsou load



Obrázek 3.2: Diagram oblasti (zdroj: vlastní)

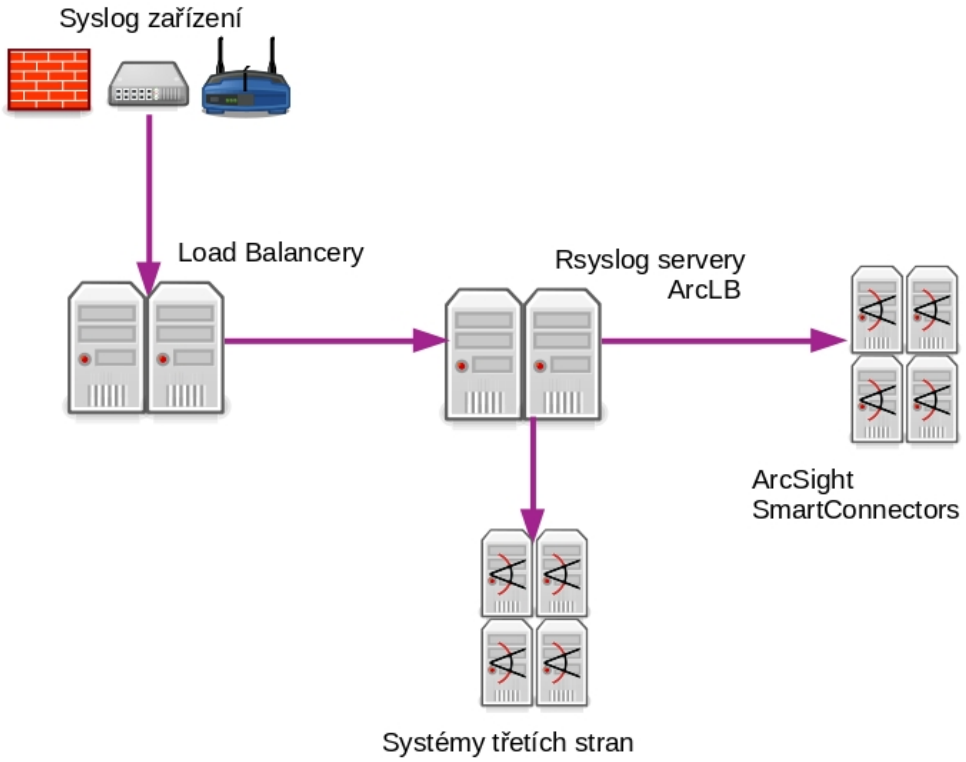
balancerem směrovány UDP datagramy, přicházející na virtuální IP. Distribuce mezi koncovými servery je nastavena v režimu tzv. “round-robin”, čili jsou všechny příchozí datagramy rovnoměrně distribuovány oběma koncovým serverům. Směrování ke koncovým serverů je nastaveno v transparentním režimu, tak nedochází k přepisu zdrojové IP adresy v UDP datagramu, pouze se příslušně upraví IP adresa cíle. Jedná se v jistém smyslu o funkcionalitu nazývanou tzv. DNAT.

Koncové servery běží na operačním systému GNU/Linux a díky páru load balancerů před nimi je zajištěna i vysoká dostupnost, jelikož load balancer monitoruje stav koncových serverů a v případě detekce výpadku přesměruje automaticky všechny syslog události na zbývající funkční koncové servery. Sice v takovém případě dojde k jisté ztrátě událostí, než je výpadek detekován, ale periodicita kontrol je v řádu sekund, což činí ztrátu přijatelnou. Koncové servery využívají pro příjem syslog UDP datagramů otevřený software Rsyslog, který je běžně dostupný v celé řadě Linuxových distribucí. Důvodem centralizace syslog událostí pomocí Rsyslog je především neexistence žádného řešení výrobce v době prvotního návrhu a implementace, které by umožňovalo příjem velkého množství

syslog událostí. Dalším důvodem pak je požadavek na přeposílání událostí pocházejících z vybraných zdrojů dalším aplikacím třetích stran za účelem dalšího zpracovávání. Při průchodu přes Rsyslog jsou tedy některé události, identifikované na základě zdrojové IP adresy, přeposílány na další cíle. Vítanou vlastností Rsyslogu (resp. v praxi snad většiny syslog daemonů) je, že se snaží rozložit přijatou událost dle specifikace syslog protokolu (RFC3164, případně novější RFC5424).

Standardně by měla událost přenášená protokolem syslog obsahovat syslog hlavičku, sestávající se především z priority, času a jména systému, který ji vygeneroval a samotné tělo zprávy. Nesprávně nakonfigurovaná zdrojová zařízení mohou ale zasílat syslog události s nekompletní hlavičkou. V řadě případů pak dokáže Rsyslog takovou chybně deklarovanou hlavičku rozpoznat a je schopen ji před přeposláním opravit tak, aby odpovídala standardu. V praxi se nejčastěji jedná o doplnění jména zdrojového systému, které Rsyslog získá z reverzního záznamu pro IP adresu uvedenou v UDP datagramu jako zdroj. Pokud neexistuje reverzní záznam, je vložena IP adresa samotná. Taktéž může Rsyslog vložit chybějící čas události, ale zde je potřeba mít na paměti, že takové časové razítko reflektuje moment, kdy byla událost doručena Rsyslogu a ne, kdy byla vygenerována zdrojovým systémem. Obvykle je ale rozdíl v řádu jednotek vteřin, čili vzniklá nepřesnost nepředstavuje zásadní problém při následné analýze události. Nadto může tělo události obsahovat další časové údaje, pokud je zdrojový systém do zprávy vkládá, které mohou například popisovat začátek a konec - typicky např. v případě logů spojení na firewallech. Není tedy jisté, že bude čas ze syslog hlavičky systémem SIEM použit.

Kromě předávání dalším systémům je klíčovou rolí Rsyslogu předat události do systému SIEM. Při standardním přeposílání syslog událostí obsahuje UDP datagram jako zdrojovou IP adresu adresu příslušného Rsyslog serveru. V našem případě by tak všechny události přicházely na další komponenty jen ze dvou zdrojových IP adres. Následující komponentou, kam jsou události mířící do SIEM řešení směrovány je ArcSight Load Balancer a následně konektory (viz. obr. 3.3). Aby správně fungovala agregace na konektorech, je potřeba aby z události z jednoho zdrojového zařízení byly doručeny v ideálním případě jednomu ArcSight konektoru, jelikož ArcSight Load Balancer používá zdrojovou IP pro mapování zdrojového zařízení s patřičným konektorem. V případě běžného přeposílání událostí z Rsyslogu má UDP datagram jako zdrojovou IP adresu adresu samotného



Obrázek 3.3: Implementace syslog (zdroj: vlastní)

Rsyslog serveru. V takovém případě by ale nedocházelo k požadovanému balancování a mapování. Pro správnou funkčnost námi používaného balancovacího mechanismu byl při konfiguraci přeposílání v Rsyslogu zvolen poněkud méně obvyklý výstup zvaný "omudpspoof". Jak název napovídá, jedná se o modul umožnující přepis nebo-li podvržení (angl. spoofing) zdrojové IP adresy. Jedná se tak o podobnou funkcionalitu, jakou je transparentní přeposílání na load balancerech první úrovně. Bohužel tento způsob předávání je díky tomu, jak je modul napsán, poměrně dost náročný na systémové prostředky. Pro každý jednotlivý UDP datagram je otevřen nový síťový socket a po jeho odeslání je opět uzavřen. Díky tomu je tento modul pomalejší (Gerhards, R. a kolektiv, 2018) oproti běžnému předávání s IP adresou Rsyslog serveru, kde je možné ponechat síťový socket otevřený po celou dobu a využít jej pro odeslání všech datagramů ve frontě. Toto "úzké hrdlo" systému lze sice řešit přídáním více paralelních Rsyslog serverů, ale vzhledem ke značným nárokům na systémové prostředky by bylo vhodné najít vhodnější řešení.

3.1.3 Sběr událostí z platformy MS Windows

Pokud jde o sběr událostí ze systémů rodiny Microsoft Windows, je situace zcela odlišná než tomu bylo v případě syslogu. V době zavádění řešení SIEM do společnosti nabízel a doporučoval výrobce Windows Unified Connector (WUC), který byl později nahrazen konektorem WiNC na platformě .NET. V důsleku časové tísně, způsobeného akutní bezpečnostní hrozbou v implementaci protokolu SMBv1 používaného WUC konektory, byl přechod z WUC na WiNC konektory ve společnosti řešen nejjednodušší možnou metodou. Došlo tedy k zachování původního konceptu konektorů s definovaným seznamem serverů, pouze WUC konektory byly nahrazeny moderními WiNC. Hlavní nevýhodou tohoto přístupu je hromadná správa. Seznamy serverů v konektorech musejí být manuálně udržovány. Nelze využít doménové politiky pro zjednodušení konfigurace. To má za následek, že i poměrně triviální změna filtru musí být provedena v každém konektoru samostatně a dokonce pro každý jednotlivý server v seznamu. Tuto nevýhodu naštěstí pomáhá eliminovat centralizovaná správa v podobně ArcSight Management Center, která nabízí možnost úpravy celých skupin konektorů najednou.

3.1.4 Sběr událostí z dalších zdrojů

Dalšími důležitými zdroji událostí jsou systémy shromažďující události z IDS senzorů. Ty jsou do SIEM ArcSight předávány prostřednictvím výrobcem dodaného databázového konektoru. Obdobně je tomu v případě událostí z antivirového řešení, které jsou centralizovaně ukládány v databázích. V celém současném návrhu SIEM prostředí je také několik dalších konektorů, ať už dodávaných přímo výrobcem nebo tzv. Flexi konektorů, které umožňují nadefinovat vlastní pravidla pro získávání a převod událostí do formátu CEF. Tyto však nepředstavují z hlediska redesignu žádnou podstatnou aktivitu.

3.2 Alternativy k syslogu

3.2.1 Výběr výrobců

Aby bylo možné určit požadované vlastnosti na SIEM řešení a případnou architekturu, bude zapotřebí analyzovat možnosti doručování událostí u výrobců zařízení tradičně po-

užívajících protokol syslog. Cílem bude jednak zjistit stav podpory logování s využitím protokolu syslog a případně i další cesty, které výrobce, respektive konkrétní produkt nabízí. Kromě podpory v produktech samotných budou vzaty v úvahu i případné další nadstavby pro správu, které nabízí možnost nějakého centralizovaného logování.

Při výběru posuzovaných technologií bude práce vycházet v první řadě z obecné kategorizace zdrojů podle typu a objemu v rámci sbíraných dat protokolem syslog. Rozdělení zachycuje následujídí tabulka:

Tabulka 3.1: Přehled zdrojů syslog událostí

Zdroj	**Podíl**
Firewally	84.9%
Servery	13.3%
Bezdrátové prvky	0.7%
VPN a AAA zařízení	0.6%
Routery a switche	0.3%
Jiné	0.3%

Jak je patrné, majoritní podíl z objemu událostí ze síťových zařízení připadá na firewally. Další síťová zařízení neprodukují ani zdaleka tolik událostí a touto optikou je jejich podíl na celkovém objemu marginální. I z toho důvodu bude důraz kladen především na segment firewallů.

Z hlediska výběru produktů jednotlivých výrobců bude srovnání zaměřeno na zařízení pro enterprise segment. Výrobci v každém segmentu nabízejí většinou širší škálu produktů, které se liší především výkonem, či rozsahem funkcí. Jelikož ale tyto produkty v rámci jedné řady sdílejí většinu vlastností, s výjimkou licenčně omezené funkcionality a jsou postaveny na stejném software, bude se analýza opírat právě o variantu užitého software.

Byť by se nabízelo soustředit se pouze na výrobce a produkty užívané společností, nebylo by z hlediska bezpečnosti ideální odhalit konkrétní portfolio zařízení ve společnosti. Z toho důvodu budou pro analýzu uvažováni přední výrobci dle hodnocení společnosti Gartner (viz. obr. 3.4), přičemž budou do srovnání zahrnuty především zařízení výrobců z kvadrantu “vůdců” seqmentu doplněné o několik dalších výrobců pohybujících se poblíž středu.

Obrázek 3.4: Magic quadrant - Firewally (Palo Alto Networks Inc., 2019)

V případě firewallů budou zahrnuty produkty Palo Alto Networks, Fortinet, Cisco, Check Point Software Technologies, Juniper Networks, Sophos a Forcepoint. Pokud jde o další typy síťových zařízení, budou zahrnuty opět hlavní výrobci jako Cisco, Juniper Networks, Exterme Networks, Aruba (HPE) nebo Pulse Secure (část produktů Juniper Networks byla vyčleněna do Pulse Secure). V případě serverů pak budou zahrnuty hlavní Linuxové distribuce s komerční podporou. Sem lze zařadit Red Hat Enterprise Linux, SuSE Linux Enterprise Server a Ubuntu. Je zřejmé, že se nejedná o vyčerpávající počet distribucí; mezi další by šlo zahrnout např. Oracle Linux nebo Amazon Linux, které jsou založeny na Red Hat Enterprise Linuxu, ale mají nezávislé řešení oprav. Avšak v kontextu

zasílání událostí systémům SIEM zde nedochází k zásadnějšímu rozdílu.

3.2.2 Stav podpory protokolu syslog

Jak již bylo v teoretické části uvedeno, je syslog poměrně problematický protokol a to především v tom smyslu, že neexistuje možnost validace takzvaně "za pochodu" a vzhledem k jednosměrné komunikaci mají výrobci de-facto volnou ruku v tom, jak logování za pomoci syslogu implementují.

Z pohledu objemu přesenených událostí jsou nejvýznamnější skupinou firewally. Na základě dokumentace jednotlivých výrobců a za pomoci ukázkových událostí bylo provedeno srovnání se standardy RFC3164 a RFC5424 s ohledem na validitu formátu událostí. Výsledky srovnání nabízí následující tabulka 3.2.

Tabulka 3.2: Firewally a podpora syslog implementací

Výrobce	**Platforma**	**RFC3164**	**RFC5424**	**RFC5424 strukt.**
Cisco	ASA 8.4 a vyšší	ne[1]	ne	ne
Checkpoint	R80.20	ne[1]	ne	ne[2]
Forcepoint	NGFW 6.6	ano	ne	ne
Fortinet	FortiOS 6.2	ne[3]	ano	ne
Juniper	SRX 18.3	ano	ano	ano
Palo Alto	PAN-OS 9.1	ano	ano	ne
Sophos	XG Firewall 18.0	ano	ne	ne

Jak je z tabulky patrné, není podpora dle jednotlivých variant dle RFC zcela běžná. Specifikaci staršího RFC3164 dodržuje pouhých 57% zařízení, novější RFC5424 pak pouhých 43% a strukturované události implementuje správně pouze jedna platforma. Kupříkladu platforma Cisco ASA (stejně jako odvozená Cisco FWSM) neformátují události ani podle jedné varianty RFC. Události využívají vlastní formát v prostém textu, který je zřejmě odvozen od RFC3164, ale nedodržuje definované formátování. Od verze software 9.10(1) je pak avizována (Cisco, 2019) podpora RFC5424, avšak omezená pouze na formát časové značky.

[1] Vlastní formát odvozený od RFC3164
[2] Nepřesně implementovaný formát RFC5424
[3] Vlastní formát

Událost v tradičním formátu používaným Cisco ASA (zdroj: vlastní):

```
<182>Dec 01 2019 12:00:00 ciscofw1 : %ASA-6-302014: Teardown TCP connection
448911 for ZONE1:10.10.10.10/65432 to ZONE2:10.20.20.20/80 duration 0:00:00
bytes 4567 TCP FINs
```

Na uvedeném příkladu je možné si povšimnout, že časové razítko zahrnuje v rozporu se standardem RFC3164 navíc rok. To je v zásadě jediný prohřešek proti standardu, byť zpráva neobsahuje ani žádný tag, který je však nepovinný. S ohledem na charakter zprávy by se z hlediska budoucího automatického zpracování nabízelo upravit formát tak, aby byl řetězec "ASA-6-302014" použitý právě jako tag a bylo tak možné jednodušeji případně filtrovat specifické typy zpráv, které firewall zasílá. Vhodnější by s ohledem na RFC3164 bylo tedy naformátovat událost například takto:

```
<182>Dec 01 12:00:00 ciscofw1 ASA-6-302014: Teardown TCP connection 448911
for ZONE1:10.10.10.10/65432 to ZONE2:10.20.20.20/80 duration 0:00:00 bytes
4567 TCP FINs
```

Pakliže by byl firewall nakonfigurován k využití formátu časového razítka dle RFC5424, vypadala by zpráva ve výchozím stavu následovně:

```
<182>2019-12-01T12:00:00Z ciscofw1 : %ASA-6-302014: Teardown TCP connection
448911 for ZONE1:10.10.10.10/65432 to ZONE2:10.20.20.20/80 duration 0:00:00
bytes 4567 TCP FINs
```

Výrobce zde v souladu s proklamací použil z RFC5424 pouze formát časového razítka. Zjednodušil si tak implementaci, ale dopustil se tak dalších provinění proti standardu. Událost tak není formátována ani podle jednoho z platných standardů. Z pohledu RFC3164 nemá validní časové razítko. Z pohledu RFC5424 jí zase chybí některé povinné části. Předně chybí verze syslog protokolu, dále nejsou správně definovány pole APP-NAME, PROCID, MSGID a STRUCTURED-DATA, které by případně mohly být

interpretovány následovně:

- APP-NAME: ":"
- PROCID: "%ASA-6-302014:"
- MSGID: "Teardown"
- STRUCTURED-DATA: "Teardown"

Přesná interpretace by se nicméně mohla v různých programech lišit, jelikož každá aplikace se s nestandardním vstupním formátem vypořádává jiným způsobem. To je dáno mimo jiné tím, že autoři programů pro příjem syslog událostí si jsou vědomi faktu, že zdrojové systémy velmi často implementují formát zpráv v nesouladu se standardy RFC. Aplikace se tak pokoušejí porozumět formátu události nad rámec definice a pokud možno ji interpretovat co nejsprávněji a nejsrozumitelněji. Například Rsyslog by tak nativní formát zprávy z Cisco ASA, co se časového razítka týče, interpretoval správně. V prvním případě by ignoroval nadbytečnou informaci o roku a v druhém případě by validně porozuměl časovému razítku formátovanému dle RFC5424, přičemž se zbytkem zprávy by zacházel jako by byla formátována dle RFC3164.

V případě řešení od společnosti Checkpoint je na místě zmínit, že kromě toho, že ne zcela dodržuje požadavky RFC3164, je poměrně specifické i pokud jde o obsah přeposílaných událostí. Strategie Checkpointu vychází z koncepce firewallů (platformy Security Gateway / GAiA), serverů pro správu a log serverů jakožto provázaného celku. A byť samotný firewall umožňuje zasílání událostí prostřednictvím syslog protokolu třetím stranám, je doporučovanou alternativou přeposílání z log serveru či serveru pro správu. Je to jednak z důvodu, že server pro správu může událost doplnit o některé další informace, jež firewall nemá k dispozici a pak také protože v případě některých typů událostí firewall skrývá z důvodu bezpečnosti vybrané informace. Popisované chování se týká například filtrů na úrovni sedmé vrstvy, jako filtrování URL.

Někteří výrobci firewallů poskytují ke svým produktům i nějakou nadstavbu pro správu či centralizovaný sběr událostí nebo obojí. Tyto produkty pak někdy poskytují rozsáhlejší škálu podporovaných formátů. Následující tabulka shrnuje stav podpory syslog protokolu v těchto produktech.

Tabulka 3.3: Podpora syslog implementací v nadstavbách firewallů

Výrobce	**Produkt**	**RFC3164**	**RFC5424**	**RFC5424 strukt.**
Checkpoint	Log Server / Log Exporter	ano	ne	ano
Forcepoint	Security Management Center	ano	ne	ne
Fortinet	FortiAnalyzer	ano	ne	ne
Juniper	Juniper Secure Analytics	ano	ano	ano
Palo Alto	Panorama	ano	ne	ne

Kromě podpory syslog protokolu dle konkrétního RFC, která se výjma strukturovaného formátu RFC5424 omezuje především na hlavičku, je také zapotřebí specifikovat podporované formáty těla zprávy. Kromě nativního formátu totiž někteří výrobci zahrnuli podporu i pro jiné formáty, jako jsou například ArcSight CEF nebo IBM Qradar Leef, případně i další. Podobné srovnání pak lze provést i s nadstavbami firewallů, které podporují logování na systémy třetích stran.

Tabulka 3.4: Firewally a podpora formátů

Výrobce	**Platforma**	**nativní**	**CEF**	**LEEF**	**další**
Cisco	ASA 8.4 a vyšší	ano	ne	ne	ne
Checkpoint	R80.20	ano	ne	ne	ne
Forcepoint	NGFW 6.6	ano	ne	ne	ne
Fortinet	FortiOS 6.2	ano	ne	ne	CSV, WELF
Juniper	SRX 18.3	ano	ne	ne	WELF
Palo Alto	PAN-OS 9.1	ano	ano	ano	definovatelné[4]
Sophos	XG Firewall 18.0	ano	ne	ne	ne

Tabulka 3.5: Nadstavby firewallů a podpora formátů

Výrobce	**Produkt**	**nativní**	**CEF**	**LEEF**	**další**
Checkpoint	Log Server - Log Exporter	ano	ano	ano	ne
Forcepoint	Security Manager Center 18.0	ano	ne	ne	CSV, XML
Fortinet	FortiAnalyzer	ano	ano	ne	ne
Juniper	Juniper Secure Analytics	ano	ne	ne	ne
Palo Alto	Panorama	ano	ano	ano	definovatelné[4]

[4]Umožňuje definovat zcela vlastní formát dat.

V neposlední řadě je nezbytné identifikovat jaké IP protokoly mohou rozličná zařízení využít pro doručení dat, pakliže uvažujeme nějakou variantu nebo variaci protokolu syslog. Zde přichází v úvahu v zásadě jen nejobvyklejší UDP, pak TCP a případně TLS. Jak již bylo zmíněno v teoretické části, existuje více možností implementace posílání událostí prostřednictvím nešifrovaného TCP a rovněž se lze setkat s událostmi přenášenými prostřednictvím TLS dle RFC5425, které přitom nedodržují formátování dle RFC5424. Někteří výrobci navíc ani neuvádějí o jakou přesnou implementaci se v případě jejich produktu jedná. Pro účely porovnání bude podpora zobecněna na nešifrované TCP a šifrované TLS, a to jakožto na čistě transportní mechanismy bez ohledu na formátování samotné syslog události. Obdobné srovnání lze sestavit i pro nadstavby firewallů, jejichž výsledky přináší tabulka 3.7.

Tabulka 3.6: Firewally a podpora protokolů

Výrobce	**Platforma**	**UDP**	**TCP**	**TLS**
Cisco	ASA 8.4 a vyšší	ano	ano	ano
Checkpoint	R80.20	ano	ano	ano
Forcepoint	NGFW 6.6	ano	ano	ano
Fortinet	FortiOS 6.2	ano	ano	ano
Juniper	SRX 18.3	ano	ano	ano
Palo Alto	PAN-OS 9.1	ano	ano	ano
Sophos	XG Firewall 18.0	ano	ne	ano

Tabulka 3.7: Nadstavby firewallů a podpora protokolů

Výrobce	**Platforma**	**UDP**	**TCP**	**TLS**
Checkpoint	Log Server - Log Exporter	ano	ano	ano
Forcepoint	Security Management Center	ano	ano	ano
Fortinet	FortiAnalyzer	ano	ano	ano
Juniper	Juniper Secure Analytics	ano	ano	ne
Palo Alto	Panorama	ano	ano	ano

Objem událostí z jiných typů síťových zařízení tvoří necelá dvě procenta z celkového objemu dat, ale pro úplnost je nezbytné provést u nich podobné porovnání jako bylo provedeno v případě firewallů. Pro výběr konkrétních výrobců byly opět primárně zohledněny

výsledky v Gartner Magic Quadrant. Vzhledem k výrazně nižšímu objemu událostí a s ohledem na to, že to povaha srovnání nevylučuje, bude v tomto případě provedena komparace napříč různými typy zdrojů najednou.

Tabulka 3.8: Podpora syslog implementací v ostatních síťových zařízeních

Výrobce	**Platforma**	**RFC3164**	**RFC5424**	**RFC5424 strukt.**
Cisco	IOS 15.6	n[5]	n	n
Cisco	NXOS 9.2	n[5]	n	n
Cisco	AireOS 8.7	n[5]	n	n
Aruba (HPE)	ArubaOS 8.6	y	y	n
Extreme Networks	EX-OS 22.6	y	n	n
Pulse Secure	PCS 8.3R4	y	y	n

Jak je ze srovnání patrné, jeden v nejznámějších výrobců síťových zařízení, společnost Cisco Systems, opět neimplementuje formát RFC3164 validním způsobem a používá vlastní odvozený formát, avšak jiný než v případě firewallů Cisco ASA. Naopak další výrobci potom specifikaci RFC3164 dodržují. Ve dvou případech je podporována i novější RFC5424, avšak strukturovaný formát neimplementuje u vybraných platforem žádný z výrobců. Podporu formátů zpráv a podporu protokolů pak zachycují následující dvě tabulky.

Tabulka 3.9: Podpora formátů zpráv ostatních síťových zařízeních

Výrobce	**Platforma**	**nativní**	**CEF**	**LEEF**
Cisco	AireOS 8.7	y	n	n
Cisco	IOS 15.6	y	n	n
Cisco	NXOS 9.2	y	n	n
Aruba (HPE)	ArubaOS 8.6	y	n[6]	n
Extreme Networks	EX-OS 22.6	y	n	n
Pulse Secure	PCS 8.3R4	y	n	n

[5]Vlastní formát odvozený od RFC3164

[6]Výrobce sice deklaruje podporu CEF, avšak formát není validně implementován jak z hlediska syntaxe tak i formátování.

Tabulka 3.10: Podpora protokolů ostatních síťových zařízeních

Výrobce	**Platforma**	**UDP**	**TCP**	**TLS**
Cisco	AireOS 8.7	y	y	y
Cisco	IOS 15.6	y	y	y
Cisco	NXOS 9.2	y	y	y
Aruba (HPE)	ArubaOS 8.6	y	y	n
Extreme Networks	EX-OS 22.6	y	n	n
Pulse Secure	PCS 8.3R4	y	y	y

Máme-li zhodnotit výsledky srovnání je zřejmé, že prakticky univerzálně podporovaný je formát syslog hlavičky dle RFC3164, byť se někteří výrobci od definice více či méně odchylují. Novější RFC5424 je implementován spíše výjimečně, avšak je-li implementován, pak odpovídá RFC. Strukturovaný formát dle RFC5424 je pak podporován jen dvěma výrobci firewallů, u jiných síťových zařízeních se s ním vůbec nesetkáváme.

Co do podpory formátů samotných zpráv se výrobci v tomto segmentu drží nativních formátů. To má pochopitelně negativní dopad na jednoduchost automatizace zpracování takových událostí, jelikož je potřeba, aby cílová SIEM platforma dokázala události správně interpretovat a použít. V případě ArubaOS lze pozitivně nahlížet na snahu výrobce poskytnout události i v nějaké univerzálnější podobě, čili ve formátu CEF, avšak způsob jakým je formát implementován způsobuje neinterpretovatelnost prakticky všech informací kromě CEF hlavičky, která poslouží spíše pouze k základní identifikaci původu události.

V případě podpory obvyklých protokolů je situace o poznání lepší a až na výjimky je podporovaný jak UDP, tak TCP a také TCP s nadstavbou TLS. To lze z hlediska zabezpečení hodnotit kladně. Jen těstě před dokončením této práce se dočkal podpory TLS i Sophos Firewall XG, což byl jediný porovnávaný produkt ze segmentu firewallů, který nedisponoval šifrovaným zasíláním událostí.

3.2.3 Podpora jiných metod

Kromě obvyklého syslogu lze u některých výrobců a platforem najít i další možnosti integrace se systémy SIEM třetích stran. Obvykle se však jedná o integraci prostřednictvím nějaké nadstavby pro centralizovanou správu a podobně.

Tabulka 3.11: Firewally a podpora jiných forem logování

Výrobce	**Platforma**	**Podporované možnosti**
Cisco	ASA 8.4 a vyšší	Sběr událostí je možný pouze pomocí syslog protokolu přímo z firewallů.
Checkpoint	R80.20	Lze využít LUA - Log Extraction API v Checkpoint Log serverech.
Forcepoint	NGFW 6.6	Sběr událostí je možný pouze pomocí syslog protokolu z firewallů nebo SMC.
Fortinet	FortiOS 6.2	Sběr událostí je možný pouze pomocí syslog protokolu z firewallů nebo FortiAnalyzeru.
Juniper	SRX 18.3	Události je možné přeposílat z Juniper JSA ve formátu JSON protokolem TCP.
Palo Alto	PAN-OS 9.1	Události je možné zasílat ve volně definovatelném formátu pomocí HTTP/HTTPS.
Sophos	XG Firewall 18.0	K získání událostí lze využít Sophos Central API, data jsou poskytována v JSON formátu.

Jak je patrné z tabulky 3.11, některé platformy nabízejí i jiné možnosti vedle tradičního syslogu. Můžeme je rozdělit na pasivní, tj. takové, které vyžadují protistranu, aby se sama aktivně dotazovala na dostupnost nových událostí a získávala je a na aktivní, tj. ty, které samy aktivně posílají události určenému cíli. K pasivním patří Checkpoint LUA a Sophos Central API. V takovém případě je nutné, aby SIEM disponoval funkcionalitou či komponentou, která umí prostřednictvím daného API události získávat. Samozřejmě by bylo možné vložit mezi API a SIEM např. nějaký script, který by fungoval jako prostředník, ale z hlediska podpory při provozu to není žádoucí stav.

Jako aktivní pak mohou být označeny Juniper JSA a Palo Alto. V případě JSA se ukazuje jako problém nedostupná specifikace transportního mechanismu, který má přenášet data ve formátu JSON. Využití této metody pro doručování událostí do SIEM se tak nejeví reálně. Ani doručování prostřednictvím HTTP/HTTPS, jak jej nabízí Palo Alto, se nejeví jako využitelná metoda pro doručování do SIEM. Z dokumentace vyplývá, že každá událost je zasílána pomocí samostatného volání HTTP POST nebo PUT. To v kontextu s vysokou frekvencí událostí, jež lze u firewallu očekávat znamená zvýšené vytížení, jelikož kromě potvrzování na úrovni TCP dochází také k potvrzování na úrovni aplikačního protokolu. Z dokumentace vyplývá, že je tato metoda přeposílání primárně určená například

pro založení incidentu na základě události splňující nějaká konkrétní kriteria. Není tedy určena pro přeposílání všech událostí o firewallovém provozu.

3.3 Porovnání řešení SIEM

Jak již bylo předesláno, je jedním z dílčích cílů práce i zhodnocení, zda současně používaný SIEM produkt MF ArcSight dostačuje všem požadavkům společnosti a zda jej případně nenahradit jiným produktem, který by lépe vyhovoval potřebám a strategii společnosti. Jedním ze strategických cílů společnosti je posilování využití cloudových služeb, především pokud jde o hostování virtuálních serverů. Od SIEM řešení je tedy požadována co nejlepší integrovatelnost s předními cloudovými poskytovateli, tj. Amazon AWS, Microsoft Azure a Google Cloud Platform. Důraz by měl přitom být kladen především na dostupnost a jednoduchost integrace cloudových služeb a zdrojů bezpečnostních informací se SIEM řešením. S ohledem na strategii společnosti by i samotné řešení SIEM mohlo být poskytováno jako cloudové. Vedle cloudových událostí budou pochopitelně i další zdroje mimo cloud, pocházející od běžné infrastruktury či aplikací vlastněných a provozovaných společností.

3.3.1 Identifikace zdrojů

Posuzované zdroje budou jednak tzv. "on-premise", čili zdroje od síťových zařízení, serverů či aplikací provozovaných v rámci vlastní sítě společnosti, a pak cloudové zdroje, tedy ty, které jsou poskytovány v rámci cloudových služeb cloudovými poskytovateli. V rámci "on-premise" zdrojů bude práce nadále uvažovat všechny typy zařízení, pro které byla prováděna analýza z hlediska logování pomocí syslog protokolu. Jako další budou zahrnuté operační systémy Windows, Red Hat Enterprise Linux a Ubuntu. Dále budou zahrnuté DHCP servery a to implementace v MS Windows a Linuxový ISC DHCP server. Také budou zahrnuty produkty z oblasti "Endpoint Security", čili nová generace antivirových produktů. Při výběru těchto produktů budou opět zohledněny přední výrobci v daném segmentu trhu jak je uvádí Gartner.

Tabulka 3.12: On-Premise zdroje událostí

Výrobce	**Platforma**	**Typ zdroje**	**Významný**
Cisco	ASA/PIX/FWSM	firewall	ano
CheckPoint	GAiA / Security gateway	firewall	ano
Fortinet	FortiGate	firewall	ano
Palo Alto	PAN-OS	firewall	ano
Sophos	XG Firewall	firewall	ano
Forcepoint	NGFW	firewall	ano
Juniper	SRX	firewall	ano
Cisco	IOS	LAN/wifi	ne
Cisco	NXOS	LAN/wifi	ne
Cisco	AireOS	LAN/wifi	ano
Aruba (HPE)	ArubaOS	LAN/wifi	ano
Extreme Networks	EX-OS	LAN/wifi	ne
Pulse Secure	PCS	LAN/wifi	ano
Microsoft	Windows	Desktop/Server	ano
RedHat	RHEL	Desktop/Server	ano
Ubuntu	Ubuntu	Desktop/Server	ano
Crowd Strike	Falcon Endpoint Protection	Endpoint prot.	ano
Microsoft	Antimalware/Defender	Endpoint prot.	ano
Sophos	Endpoint Protection	Endpoint prot.	ano
Symantec	Endpoint Protection	Endpoint prot.	ano
Trend Micro	Apex One	Endpoint prot.	ano
Microsoft	DHCP server	DHCP	ano
OSS	DHCP server	DHCP	ano

Tabulka 3.12 přináší přehled uvažovaných zdrojů událostí pro "on-premise" zařízení. Jak je zřejmé, obsahuje i sloupec určující, zda se jedná o významný zdroj událostí či nikoli. Obecně jsou jako nevýznamné klasifikovány především zdroje typu switch nebo router, jelikož významná část událostí z těchto zdrojů je spíše využitelná pro operační dohled, ale z pohledu bezpečnosti nepředstavují žádnou významnou užitnou informaci. Snad s výjimkou událostí týkajících se přihlašování k samotným zařízením. V následující tabulce je stejný pohled na zdroje cloudové.

Uvedené cloudové integrace jsou spíše ty základní. Oblast cloudových služeb a řešení se dynamicky rozvíjí. Také je zde patrná snaha poskytovatelů nabídnout univerzálně využitelné systémy předávání zpráv, či událostí, které mohou být využity jak pro komunikaci

Tabulka 3.13: Cloudové zdroje událostí

Výrobce	**Platforma**	**Typ zdroje**	**Významný**
Amazon	AWS CloudTrail	Cloud auditing	ano
Amazon	AWS VPC	Cloud firewall	ano
Amazon	AWS GuardDuty	Cloud IDS	ano
Amazon	AWS WAF	Cloud firewall	ano
Google	Cloud Monitoring	Cloud operations	ne
Google	Cloud PubSub	Cloud auditing	ano
Google	Cloud VPC	Cloud firewall	ano
Microsoft	Azure EventHub	Cloud auditing	ano
Microsoft	Azure AD	Cloud AAA	ano
Microsoft	Azure NSG/VPC	Cloud firewall	ano

mezi jednotlivými komponentami v cloudu, tak pro bezpečnostní monitoring. Vedle v tabulce zmíněných Azure Event Hub a GCP Pub/Sub nabízí i Amazon podobné řešení pod názvem Kinesis.

3.3.2 Podpora v jednotlivých SIEM

Pokud jde o porovnání jednotlivých řešení SIEM, je klíčovým ukazatelem podpora nejčastějších typů událostí, resp. těch, které přinášení nějakou hodnotu z pohledu bezpečnosti. V předchozím kroku bylo definováno celkem 33 hlavních typů zdrojů událostí, jejichž integrovatelnost s platformou SIEM je z hlediska společnosti důležitá. Přičemž integrovatelností je zde myšleno jak samotné získání událostí a jejich vložení do platformy SIEM, tak i schopnost platformy zpracovat informace nesené v události, tedy "porozumění" obsahu zprávy a možnost dále s ním analyticky pracovat. Je sice pravdou, že snad všechny SIEM platformy nabízejí možnost vytvořit vlastní parsery, které mohou zajistit zpracování i takových událostí, které platforma v základu neumí, avšak problémem může být jednak dlouhodobá podpora a spolehlivost takového postupu a také samotný proces získání událostí a jejich vložení do platformy SIEM. To může vyžadovat např. externí scripty komunikující s API poskytovatele, což přidává na komplexnosti řešení a vnáší potenciální rizika do celého procesu zpracování událostí.

Jak je patrné z tabulky 3.14 aktuálně využívaná platforma ArcSight nabízí integruje

Tabulka 3.14: ArcSight - podpora zdrojů událostí

Výrobce	**Platforma**	**Metoda integrace**	**P**	**V**	**I**
Cisco	ASA/PIX/FWSM	syslog	1	2	1
CheckPoint	GaiA / Security gateway	Syslog / LUA	1	2	1
Fortinet	FortiGate	syslog	1	2	1
Palo Alto	PAN-OS	syslog	1	2	1
Sophos	XG Firewall	syslog	1	2	1
Forcepoint	NGFW	syslog	1	2	1
Juniper	SRX	syslog	1	2	1
Cisco	IOS	syslog	1	1	1
Cisco	NXOS	syslog	1	1	1
Cisco	AireOS	syslog	1	2	1
Aruba (HPE)	ArubaOS	syslog	1	2	1
Extreme Networks	EX-OS	–	0	1	1
Pulse Secure	PCS	syslog	1	2	1
Microsoft	Windows	Win Eventing 6.0 / WEC	1	2	1
RedHat	RHEL	Syslog / file	1	2	1
Ubuntu	Ubuntu	Syslog / file	1	2	1
Microsoft	Antimalware/Defender	Win Eventing 6.0	1	2	1
Symantec	Endpoint Protection	Syslog / jdbc	1	2	1
Sophos	Endpoint Protection	jdbc	1	2	1
Trend Micro	Apex One	syslog	1	2	1
Crowd Strike	Falcon Endpoint Protection	API+file	1	2	2
Microsoft	DHCP server	file	1	2	1
OSS	DHCP server	Syslog / file	1	2	1
Google	Cloud Monitoring	–	0	2	1
Google	Cloud PubSub	–	0	2	1
Google	Cloud VPC	–	0	2	1
Amazon	AWS CloudTrail	API	1	2	1
Amazon	AWS VPC	API+syslog	1	2	2
Amazon	AWS GuardDuty	API	1	2	1
Amazon	AWS WAF	–	0	2	1
Microsoft	Azure EventHub	API+syslog	1	2	2
Microsoft	Azure AD	API+syslog	1	2	2
Microsoft	Azure NSG/VPC	API+syslog	1	2	2

téměř všechny požadované zdroje. Výjimku tvoří síťové prvky od Extreme Networks, které ale nejsou považovány za významný zdroj událostí. Jako hlavní nedostatek lze tedy vytknout nemožnost integrovat události z Google Cloud Platform. Také navázanost některých dalších cloudových zdrojů na syslog protokol je přinejmenším diskutabilní, jelikož vyžaduje instalaci syslog Connectoru, aby události mohly být doručeny do SIEM.

Tabulka 3.15: LogRhythm - podpora zdrojů událostí

Výrobce	**Platforma**	**Metoda integrace**	**P**	**V**	**I**
Cisco	ASA/PIX/FWSM	Syslog / eStreamer	1	2	1
CheckPoint	GaiA / Security gateway	LUA	1	2	1
Fortinet	FortiGate	Syslog	1	2	1
Palo Alto	PAN-OS	syslog	1	2	1
Sophos	XG Firewall	–	0	2	1
Forcepoint	NGFW	–	0	2	1
Juniper	SRX	syslog	1	2	1
Cisco	IOS	syslog	1	1	1
Cisco	NXOS	syslog	1	1	1
Cisco	AireOS	syslog	1	2	1
Aruba (HPE)	ArubaOS	syslog	1	2	1
Extreme Networks	EX-OS	syslog	1	1	1
Pulse Secure	PCS	syslog	1	2	1
Microsoft	Windows	WinEvt 6.0 / WEC	1	2	1
RedHat	RHEL	file	1	2	1
Ubuntu	Ubuntu	file	1	2	1
Microsoft	Antimalware/Defender	WinEvt 6.0 / WEC	1	2	1
Symantec	Endpoint Protection	Syslog / sql	1	2	1
Sophos	Endpoint Protection	n/a	0	2	1
Trend Micro	Apex One	SNMP	1	2	1
Crowd Strike	Falcon Endpoint Protection	syslog	1	2	1
Microsoft	DHCP server	WinEvt 6.0 / WEC	1	2	1
OSS	DHCP server	file	1	2	1
Google	Cloud Monitoring	–	0	2	1
Google	Cloud PubSub	–	0	2	1
Google	Cloud VPC	–	0	2	1
Amazon	AWS CloudTrail	API	1	2	1
Amazon	AWS VPC	API	1	2	1
Amazon	AWS GuardDuty	API	1	2	1
Amazon	AWS WAF	API	1	2	1
Microsoft	Azure EventHub	API	1	2	1
Microsoft	Azure AD	API	1	2	1
Microsoft	Azure NSG/VPC	API	1	2	1

Pro účely porovnání bude použita následující metodika. Pro každý zdroj bude zohledněna podpora [P], kdy podporovaný zdroj nabývá hodnotu 1 a nepodporovaný hodnotu

Tabulka 3.16: QRadar - podpora zdrojů událostí

Výrobce	**Platforma**	**Metoda integrace**	**P**	**V**	**I**
Cisco	ASA/PIX/FWSM	syslog	1	2	1
CheckPoint	GaiA / Security gateway	Syslog / LUA	1	2	1
Fortinet	FortiGate	syslog	1	2	1
Palo Alto	PAN-OS	syslog	1	2	1
Sophos	XG Firewall	syslog	0	2	1
Forcepoint	NGFW	syslog	1	2	1
Juniper	SRX	syslog	1	2	1
Cisco	IOS	syslog	1	1	1
Cisco	NXOS	syslog	1	1	1
Cisco	AireOS	syslog	1	2	1
Aruba (HPE)	ArubaOS	syslog	1	2	1
Extreme Networks	EX-OS	syslog	1	1	1
Pulse Secure	PCS	syslog	1	2	1
Microsoft	Windows	WinEvt 6.0 / WEC	1	2	1
RedHat	RHEL	syslog	1	2	1
Ubuntu	Ubuntu	syslog	1	2	1
Microsoft	Antimalware/Defender	WinEvt 6.0 / WEC	1	2	1
Symantec	Endpoint Protection	syslog	1	2	1
Sophos	Endpoint Protection	JDBC	1	2	1
Trend Micro	Apex One	SNMP	1	2	1
Crowd Strike	Falcon Endpoint Protection	API	1	2	1
Microsoft	DHCP server	WinEvt 6.0 / WEC	1	2	1
OSS	DHCP server	syslog	1	2	1
Google	Cloud Monitoring	n/a	0	2	1
Google	Cloud PubSub	n/a	0	2	1
Google	Cloud VPC	n/a	0	2	1
Amazon	AWS CloudTrail	API	1	2	1
Amazon	AWS VPC	API	1	2	1
Amazon	AWS GuardDuty	API	1	2	1
Amazon	AWS WAF	API	1	2	1
Microsoft	Azure EventHub	API	1	2	1
Microsoft	Azure AD	API	1	2	1
Microsoft	Azure NSG/VPC	API	1	2	1

0. Dále významnost [V], kdy významné zdroje nabývají hodnoty 2 a méně významné hodnoty 1. Dále bude bude hodnocena složitost integrace [I]. Ta může nabývat hodnot 1 = jednoduchá integrace, 2 = komplexní integrace. Za komplexní integraci jsou považovány takové integrace, které vyžadují přítomnost další úrovně mezi samotným zdrojem dat a platformou SIEM, které vyžadují netriviální konfiguraci. To může být například Windows

Tabulka 3.17: Splunk - podpora zdrojů událostí

Výrobce	**Platforma**	**Metoda integrace**	**P**	**V**	**I**
Cisco	ASA/PIX/FWSM	syslog	1	2	1
CheckPoint	GaiA / Security gateway	Syslog / LUA	1	2	1
Fortinet	FortiGate	syslog	1	2	1
Palo Alto	PAN-OS	syslog	1	2	1
Sophos	XG Firewall	syslog	1	2	1
Forcepoint	NGFW	–	0	2	1
Juniper	SRX	syslog	1	2	1
Cisco	IOS	syslog	1	1	1
Cisco	NXOS	syslog	1	1	1
Cisco	AireOS	syslog	1	2	1
Aruba (HPE)	ArubaOS	syslog	1	2	1
Extreme Networks	EX-OS	syslog	1	1	1
Pulse Secure	PCS	syslog	1	2	1
Microsoft	Windows	Win API	1	2	1
RedHat	RHEL	Syslog / file / journald	1	2	1
Ubuntu	Ubuntu	Syslog / file / journald	1	2	1
Microsoft	Antimalware/Defender	Win API	1	2	1
Symantec	Endpoint Protection	Syslog / file	1	2	1
Sophos	Endpoint Protection	Win API	1	2	1
Trend Micro	Apex One	syslog	1	2	1
Crowd Strike	Falcon Endpoint Protection	API	1	2	1
Microsoft	DHCP server	Win API	1	2	1
OSS	DHCP server	Syslog / file	1	2	1
Google	Cloud Monitoring	API	1	2	1
Google	Cloud PubSub	API	1	2	1
Google	Cloud VPC	API	1	2	1
Amazon	AWS CloudTrail	API	1	2	1
Amazon	AWS VPC	API	1	2	1
Amazon	AWS GuardDuty	API	1	2	1
Amazon	AWS WAF	API	1	2	1
Microsoft	Azure EventHub	API	1	2	1
Microsoft	Azure AD	API	1	2	1
Microsoft	Azure NSG/VPC	API	1	2	1

Event Collector u sběru událostí ze systémů Windows nebo komplexní API jako CheckPoint LUA, které je složité na integraci. Je-li k dispozici více metod sběru událostí, je pro účely porovnání upřednostněna ta jednodušší.

Obecný vzorec pro celkové hodnocení dané platformy je pak vypadá následovně:

$$H_x = \sum_{k=1}^{N} P * V * (1 - I/10)$$

Vzorec je navržen tak, aby mírně penalizoval příliš složitou integraci, avšak důraz je kladen na samotnou podporu různých typů zdrojů.

3.3.3 Výběr řešení

Aplikujeme-li navržený vzorec na jednotlivá řešení SIEM, dostaneme výsledky, jak je představuje tabulka 3.18.

Tabulka 3.18: Porovnání SIEM řešení

SIEM	**podporovaných zdrojů**	**celkové hodnocení**
Splunk	32	54.9
Qradar	29	49.5
ArcSight	28	47.6
LogRhythm	27	45.9

Z výsledků je patrné, že převážná většina identifikovaných typů zdrojů událostí je podporována všemi uvažovanými SIEM řešeními. Stávající řešení SIEM ArcSight obsadilo v hodnocení až třetí pozici, což je dáno především absencí podpory Google Cloud Platform a poměrně složitou integrací s některými zdroji Amazon AWS. Nejlepší výsledek v tomto hodnocení má Splunk, který podporuje všechny uvažované typy zdrojů vyjma Forcepoint NGFW firewallů. Také nabízí jednodušší možnosti integrace oproti konkurentům.

Z hlediska strategie společnosti v orientaci na cloud nabízejí IBM QRadar, Splunk i LogRhythm variantu SaaS, kdy jádro SIEM řešení je spravováno přímo výrobcem a je postaveno na cloudovém řešení. Microfocus ArcSight nabízí ArcSight pro AWS, nejedná se však o službu nýbrž pouze o standardní instanci přenesenou do cloudu, která je plně ve správě zákazníka.

Ideálním dalším hodnotícím kritériem by bylo TCO jednotlivých řešení. Zde je však řada překážek. Předně se obvykle liší licenční modely řešení provozovaného samotným zákazníkem od služby na bázi SaaS. Navíc se licenční model může lišit i u jednotlivých komponent, např. ArcSight ESM je licencovaná na základě EPS, ale ArcSight Data Platform (Loggery, SmartConnectory, atd) jsou licencovány na základě objemu dat za den. V případě řešení v režii zákazníka, je potřeba zakalkulovat veškeré náklady na technologie,

hosting, housing a podobně. Asi největším problémem je však cenová politika tvořená na míru, jelikož předpokládáme velkou společnost s řádově tisíci až deseti tisíci různými zdroji událostí. Nehledě na to, že s výjimkou Splunku žádný další dodavatel SIEM nepublikuje ceny, je zřejmé, že velká společnost je schopna vyjednat výrazné slevy.

Jelikož je celá tato práce zaměřena na technické aspekty řešení, bude i doporučení opřeno o výsledky srovnání z hlediska podpory. Dle něj je možné formulovat následující dvě možnosti. První možností je setrvání u stávajícího řešení MF ArcSight, jelikož do něj lze integrovat převážnou většinu uvažovaných zdrojů. Navíc lze předpokládat, že Micro Focus do budoucna nebude ignorovat třetího nejsilnějšího hráče na poli cloudových platforem a nabídne možnost integrace událostí z prostředí Google Cloud Platform. Jako dočasné řešení je možné uvažovat o integraci prostřednictvím FlexConnectoru.

Alternativou k zachování SIEM ArcSight je pak migrace na Splunk, který ze srovnání vyšel nejlépe a podporuje téměř všechny uvažované zdroje událostí. Navíc nabízí variantu SaaS, která je v souladu se strategií společnosti a umožňuje i velmi jednoduché a variabilní škálování. Splunk Cloud navíc nabízí tzv. HEC event kolektory, které umožňují doručení rozličně formátovaných událostí prostřednictvím protokolu HTTPS. Díky tomu lze např. události z cloudových služeb integrovat na přímo bez nutnosti instalace prostředníka, v případě Splunk např. Heavy Forwarderu.

Na základě vyslovených možností se z krátkodobého hlediska autor přiklání spíše k variantě zachování stávajícího řešení MF ArcSight a jeho optimalizaci. Pro volbu migrace na řešení Splunk Cloud by bylo nutné posoudit i ekonomický aspekt.

3.4 Identifikace slabých míst

Jelikož cílem práce je navrhnout nový design, který by lépe vyhovoval současnému vytížení celého řešení a vhodně vytěžil nové vlastnosti stávajících komponent či dokonce komponent nových, je potřeba předně definovat veškerá slabá místa, která budou předmětem optimalizace. V některých případech bude možné více alternativních řešení, z nichž bude na základě komparace vybráno nejvíce vyhovující.

Při návrhu úprav designu, optimalizací a testování budou uvažovány následující hodnoty, jak je uvádí tabulka 3.19. Jelikož počet událostí zasílaný Windows servery se značně

liší s ohledem na role daného serveru, byl uvažovaný vzorek stanoven na 100 serverů různých rolí, které zasílají události jednomu SmartConnectoru.

Tabulka 3.19: Uvažované toky dat

Zdroj	Události za vteřinu	Datový tok
Průměrné hodnoty na syslog koncentrátoru	27000	66 Mb/s
Hodnoty ve špičce na syslog koncentrátoru	38000	93 Mb/s
Průměrné hodnoty ze 100 serverů Windows	2300	56 Mb/s

3.4.1 Syslog

Události dopravované prostřednictvím syslog protokolu jsou procesovány řadou komponent, jak je detailně popsáno v kapitole 3.1.2. Jsou zde load balancery, kde je nakonfigurována virtuální IP adresa používaná pro doručování událostí. Dále jsou události distribuovány skupině Rsyslog serverů - syslog koncentrátorů, které zároveň nesou po jednom ArcSight Load Balanceru. Tomu jsou události předávány v rámci vnitřního (localhost) rozhraní. Pro správnou funkci balancování je využitý udpspoof modul Rsyslogu, který vkládá do odchozích UDP datagramů jako zdrojovou IP adresu adresu původního zařízení, od kterého událost byla přijata. Pokud by nebylo předávání ArcSight Load Balanceru řešeno v transparentním režimu a bylo použito běžné přeposílání, zdrojová IP adresa v datagramu by byla nastavena na adresu localhost, čili 127.0.0.1 a pro ArcSight Load Balancer by tak všechny události pocházely z jednoho zdroje. To by znemožnilo doručení událostí od jednoho původního zdroje jednomu SmartConnectoru a mělo tak v důsledku dopad na agregaci událostí, která právě na SmartConnectorech probíhá.

Problémy a rizika:

1. UDP spoof modul je dle dokumentace výrazně méně výkonný než běžný modul pro přeposílání.

2. Je zde podezření, že při příjmu či zpracování událostí Rsyslog servery dochází ke ztrátám událostí.

3. Rsyslog nedokáže monitorovat dostupnost ArcSight Load Balanceru, což může vést ke ztrátě událostí.

Cíle:

1. Otestovat výkonnost UDP spoof modulu, případně navrhnout alternativní řešení.
2. Otestovat propustnost Rsyslog serverů, identifikovat vhodné úpravy nastavení.
3. Navrhnout řešení zajišťující monitoring ArcSight Load Balancerů a běh v režimu vysoké dostupnosti.

Metodika testování: K testům toku syslog dat bude použitý anonymizovaný vzorek dat, který bude "přehráván" v laboratorním prostředí pomocí sady nástrojů tcpdump a tcpreplay. To umožní realizovat řadu měření, ověřit přítomnost a rozsah předpokládaných problémů a otestovat možná řešení. Jako pomocné nástroje pro monitoring stavu systému a některých ukazatelů budou využity např. nástroje "sar" nebo "netstat".

3.4.2 Windows

Pro sběr událostí ze systémů s OS Windows jsou využívány WiNC konektory. Jak je zmíněno v dřívější kapitole, došlo historicky k migraci z tzv. WUC konektorů na WiNC se zachováním celé logiky sběru událostí.

Problémy a rizika:

1. Při zpracování událostí WiNC konektory jsou některé události filtrovány, avšak filtrace je realizována až na výstupu.
2. Rostoucí počet zařízení a tedy i konektorů může do budoucna znamenat problém z hlediska využití velkého počtu spojení.

Cíle:

1. Navržení filtrace nežádoucích událostí s využitím MS XPath funkcionality na úrovni zdrojů událostí.
2. Identifikovat a otestovat možnosti agregace událostí.

Metodika testování: K testování filtrace i agregace bude využito testovací prostředí společnosti. Pomocí nástroje NetBalancer bude sledováno využití sítě jedním konkrétním SmartConnectorem před a po optimalizaci.

3.4.3 Single point of failure

Vedle již zmíněného problému s ArcSight Load Balancery lze identifikovat v současném návrhu i další místa, kde za určitých okolností hrozí ztráty událostí.

Problémy a rizika:

1. Jedno ESM v regionu lze označit za riziko, pakliže dojde k dlouhodobějšímu výpadku, což povede k zaplnění cache konektorů.

Cíle:

1. Navrhnout možná řešení k zajištění zpracování událostí i v případě výpadku ESM.

3.5 Validace a řešení slabých míst

3.5.1 Syslog a ArcLB

Aby bylo možné provést testování toku dat pomocí protokolu syslog, bylo připraveno laboratorní prostředí pro simulaci reálného stavu. Laboratorní prostředí bylo realizováno pomocí dvoumodulového serveru Super Micro. Detailní specifikace jsou k nalezení níže v tabulce 3.20. Na oba nezávislé moduly byl nainstalován operační systém Debian Linux 10.3, který sloužil jako hostitelské prostředí pro další virtualizované komponenty.

Tabulka 3.20: Parametry laboratorních serverů

CPU	AMD Opteron Processor 6140 2.6GHz
Počet jader	16
RAM	128 GB
HDD	300 GB VelociRaptop 10k
OS	Debian Linux 10.3 (Buster)
Jádro	4.19.0-8-amd64
KVM/Qemu	3.1+dfsg-8+deb10u4

Jako virtualizační platforma byla zvolena virtualizace KVM ve verzi 3.1, která má přímou podporu v Linuxovém jádře. Navíc jako opensource platforma nemá žádná omezení, pokud jde o alokaci dostupných zdrojů pro virtuální servery. Virtuální servery pak využívaly operační systém CentOS 7.7.1908. Jelikož se v případě systému CentOS jedná o komunitní build ze zdrojových balíčků Red Hat Enterprise Linux, lze předpokládat, že se oba systémy budou chovat podobně.

V případě produkčních serverů, které hostují Rsyslog a ArcSight Load Balancer se v testovaném případě jedná o nevirtualizované servery. Výhodou pro testování je, že využívají také procesory AMD Opteron, avšak s menším počtem jader, ale vyšší frekvencí. Osazení RAM je na produkčních serverech výrazně nižší, ale to bude kompenzováno na úrovni VM.

Tabulka 3.21: Parametry produkčních serverů

CPU	AMD Opteron Processor 4238 3.7 GHz
Počet jader	12
RAM	32 GB
HDD	2x 1000 GB 7.2k
OS	Red Hat Enterprise Linux 7.7
Jádro	3.10.0-1062
Rsyslog	7.6.7
ArcSight Load Balancer	1.4.0

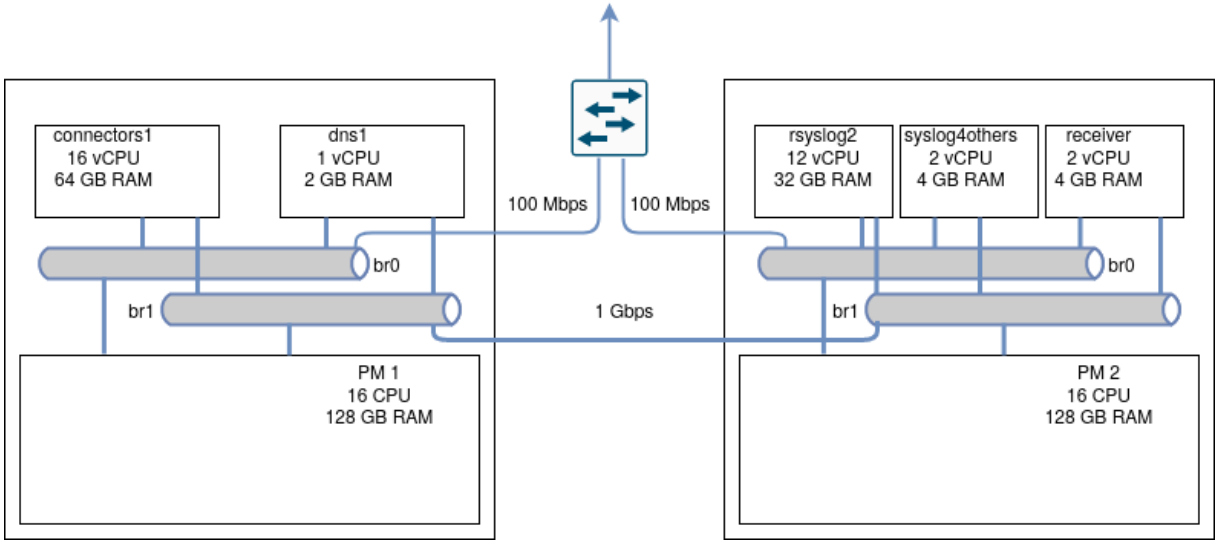
Pro přehlednost byly oba servery sloužící v laboratorním prostředí k hostování virtuálních serverů pojmenovány **pm1** a **pm2**. Oba fyzické servery disponují hned čtyřmi síťovými rozhraními. Jedno síťové rozhraní tedy bylo vždy využito pro připojení do lokální sítě, aby bylo možné servery jednoduše spravovat. Druhé rozhraní pak bylo určeno pro komunikaci mezi oběma hostitelskými servery a zároveň pro komunikaci s virtuálními servery. K tomu bylo využito rozhraní bridge. Externí rozhraní tak bylo označeno jako `br0` a interní rozhraní jako `br1`.

Jako interní síť byla zvolena 10.200.200.0/24, přičemž oba servery byly propojeny na přímo přes interní bridge rozhraní. To umožnilo komunikaci rychlostí až 1 Gbps. Výchozí gateway byla v případě všech serverů v externí síti připojené pouze rychlostí 100 Mbps.

Tabulka 3.22: Virtuální servery

Hostitel	Hostname	vCPU	RAM	role
PM1	connectors1	16	64GB	hostitel ArcSight SmartConnectorů
PM1	dns1	1	2GB	lokální DNS
PM2	rsyslog2	12	32GB	Rsyslog server a ArcLB
PM2	syslog4others	2	4GB	syslog třetích stran
PM2	receiver	2	4GB	alternativní syslog destinace

Z kapacitních důvodů byly testy prováděny jen s jedním Rsyslog serverem. Vzhledem k tomu, že v produkčním prostředí je Rsyslog serverům předřazen load balancer v režimu round robin, tento postup by neměl nikterak ovlivnit výsledky. Z hlediska výkonu by zvolený virtuální hardware měl plně postačovat, jelikož ani na produkčních systémech nedochází k přetížení všech 12 CPU a "load average", jakožto klíčový ukazatel nedostatku CPU dlouhodobě nedosahuje limitní hodnoty 12.



Obrázek 3.5: Diagram laboratorního prostředí (zdroj: vlastní)

Pro zjištění aktuálního stavu v produkčním prostředí byl několikrát proveden záznam síťové komunikace protokolu syslog pomocí nástroje "tcpdump". Bylo záměrně vybráno časové období lokální špičky, kdy se průměrná hodnota EPS blížila uvažovaným 38000 ve špičce. V rámci každého sběru dat byly paralelně prováděny hned tři záznamy - záznam příchozí komunikace (KP), záznam komunikace mezi Rsyslog procesem a ArcSight Load

Balancerem (KRA) a také záznam odchozí komunikace směrem od ArcSight Load Balanceru ke SmartConnectorům (KO). Jelikož veškeré toky událostí využívají protokol UDP, je každá událost nesena právě jedním UDP datagramem. Výstupy z nástroje tcpdump jsou tak jednoduše porovnatelné. Navíc byla vždy na začátku a konci testu odečtena hodnota "packet receive errors" dostupná pomocí příkazu "`netstat -su`". Jedná se o počítadlo UDP datagramů, které sice byly doručeny na síťové rozhraní, resp. do UDP bufferu, ale nebyly doručeny žádné cílové aplikaci z důvodu přetečení bufferu. To nastává zpravidla v situaci, kdy koncová aplikace není schopna dostatečně rychle načítat data z bufferu.

Tabulka 3.23: Syslog toky v produkčním prostředí

Test	KP [d]	Netstat [d]	Filtr [e]	KRA [d]	KO [d]	ZRS [%]	ZC [%]	EPS
1	7917680	1958409	517750	5551340	5546344	24.98	25.05	43987
2	6978780	990882	527283	5327034	5322240	17.43	17.5	38771
3	6496175	640616	525622	5177598	5173456	13.28	13.35	36090
4	6277500	437727	532434	5220142	5215444	9.14	9.22	34875
5	7074504	1155066	524079	5545502	5540511	15.34	15.42	39303
6	6451492	595408	528134	5212588	5211545	12	12.02	35842
7	6710284	755729	527321	5400895	5396034	12.65	12.73	37279
8	6527398	644882	516274	5355455	5354919	10.91	10.92	36263
9	6589911	619834	515088	5293378	5288614	12.86	12.94	36611
10	6833101	942662	517331	5375073	5374965	14.89	14.9	37962

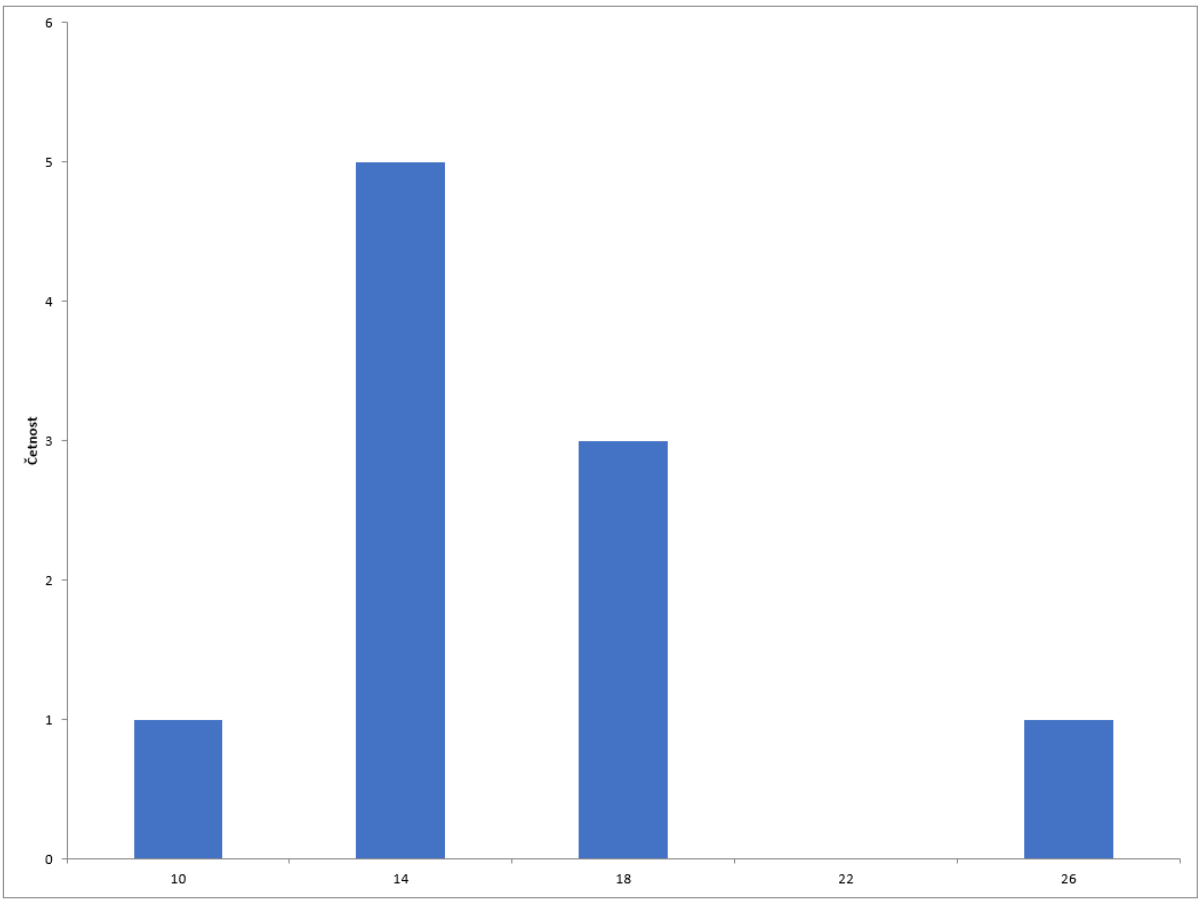
Jak je patrné z tabulky 3.23 stávající produkční prostředí vykazuje ztráty. Ve sloupci ZRS je vypočtena ztrátovost mezi objemem vstupních dat a objemem dat přeposlaných Rsyslogem na ArcSight Load Balancer. Procentuální hodnota zohledňuje události, které jsou Rsyslogem úmyslně zahozeny. Ve sloupci ZC je pak zachycena celková ztrátovost v rámci obou komponent. Poslední sloupec EPS udává průměrné množství událostí za vteřinu. Níže jsou oba vzorce dle kterých probíhal výpočet ztrát. Je však potřeba zdůraznit, že hodnoty ztrát jsou pouze orientační. Jelikož měření bylo prováděno na produkčním systému při plném provozu, je potřeba brát v úvahu řadu potenciálních problémů, jež mohou výsledky ovlivnit. Předně je to mírná odchylka ve startu jednotlivých instancí nástroje "tcpdump", byť byl sběr dat řešen scriptem. Také během samotného zpracovávání syslog událostí Rsyslogem a ArcSight Load Balancerem dochází ke zpoždění, kdy nějakou

chvíli trvá, než přijatá událost projde všemi pravidly a je následně odeslána na patřičný SmartConnector.

$$ZRS = 100 - (KRA/((KP - Netstat)/100))$$

$$ZC = 100 - (KO/((KP - Netstat)/100))$$

Rychlý statistickým ověřením hodnot pomocí normálního rozdělení je možné výsledky ztrát interpretovat tak, že v případě prvního měření se jedná o odlehlou hodnotu. Z toho důvodu budeme při dalších testech pracovat se zachycenými daty z měření vykazující druhou nejvyšší ztrátovost, v našem případě tedy s měřením číslo dvě.



Obrázek 3.6: Normální rozdělení Komunikace Odchozí (KO) (zdroj: vlastní)

V první fázi je potřeba analyzovat vstupní data. Záznam “pcap” z druhého meření má následující parametry:

Tabulka 3.24: Parametry vzorku č. 2

Celkový počet událostí	6978780
Počet událostí určených k zahození	990882
Počet událostí k předání ArcLB	6451497
Počet událostí k předání externímu systému č. 1	16483
Počet událostí k předání externímu systému č. 2	1262104

Z informací získaných na produkčním systému není možné přesně určit, kde dochází ke ztrátám. Počítadlo Nestat indikuje, že značná část ztrát se děje právě na síťové úrovni. Bohužel toto počítadlo pracuje se všemi UDP datagramy doručenými všem rozhraním na všech portech, proto jeho hodnoty mohou v nekontrolovaném prostředí, jakým je produkční systém, vnášet do výsledků značnou nepřesnost. Je tedy potřeba experimentálně ověřit, zda ke ztrátám nedochází i při samotném zpracování událostí Rsyslogem.

Jako metodika testování bylo zvoleno měření v laboratorním prostředí, kdy budou zachycená data ze vzorku číslo 2 přehrána pomocí nástroje "tcpreplay". Toto řešení je možné, jelikož se jedná o jednosměrný UDP provoz směrem od klientů k serveru bez session a bez potvrzování. Před samotným přehráním je nutné vstupní data upravit, aby MAC adresy a cílová IP adresa a případně i port reflektovaly laboratorní uspořádání. Jelikož přehrávání bude probíhat z hostitelského server **pm2**, bude zdrojová MAC adresa odpovídat MAC adrese virtuálního rozhraní "vnetX", ke kterému je připojen virtuální server **rsyslog2** na němž běží Rsyslog. Cílová MAC adresa stejně jako IP adresa budou potom odpovídat MAC a IP adrese na interním rozhraní server **rsyslog2**. Zde bude probíhat na vstupu kontrolní měření počtu doručených událostí za pomocí nástroje iptables. Dále budou periodicky Rsyslogem publikovány statistiky procesování přijatých událostí. Odchozí události z Rsyslogu budou měřeny pomocí nástroje iptables. Při tomto měření bude server **rsyslog2** nakonfigurován shodně se serverem v produkčním prostředí.

Jelikož doručování událostí destinaci SIEM je z hlediska objemu majoritní a s ohledem na to, že Rsyslog pro doručování využívá výstupní modul "omudpspoof", který dle dokumentace nedisponuje takovou rychlostí, jako konvenční modul "omfwd", byla provedena stejná sada testů také s využitím "omfwd" a se zápisem lokálně do souboru, aby bylo možné určit, zda má volba výstupní metody vliv na rychlost odebírání dat na vstupu. Následující tabulka obsahuje zprůměrované výsledky jednotlivých testových sad.

Tabulka 3.25: Ztráty na Rsyslogu

Test	Vstup IPT	Netstat	Vstup RS	Filtr	Výstup RS	ZIRS	ZRS
omudpspoof	6967434	1816718	5154105	380461	4773645	1813329	0
omfwd	6964948	1202740	5820221	425900	5394321	1144727	0
omfile	6966791	24797	6941994	500621	6441374	24797	0

Při vyhodnocení dat je zjevné, že veškeré ztráty v laboratorním prostředí vznikají na vstupu Rsyslog serveru, tedy během čtení dat z UDP bufferu. Ztráty při zpracování událostí Rsyslogem nebyly naměřeny. Za povšimnutí také stojí výrazný rozdíl ve výsledcích jednotlivých výstupních metod při jinak shodné konfiguraci. Modul "omudpspoof" v souladu s očekáváním vykázal nejhorší výsledek, avšak i běžné přeposílání za pomoci modulu "omfwd" vykazuje při srovnání se zápisem do souborů výrazně vyšší ztrátovost na vstupu.

S ohledem na zjištěné výsledky je vhodné soustředit se na možné optimalizace, které mohou zajistit rychlejší nebo efektivnější získávání UDP datagramů ze systémového bufferu. Na základě odborné literatury a dohledatelných informací byly identifikovány dvě hlavní oblasti pro zlepšení. Jedná se jednak o jaderné parametry, tedy nízkoúrovňová nastavení operačního systému a dále pak parametry aplikační dostupné v Rsyslogu. V tabulce 3.26 jsou uvedené všechny parametry i s výchozími hodnotami.

Parametry jádra OS	net.core.rmem_max	212992 [B]
	net.core.wmem_max	212992 [B]
	net.core.netdev_max_backlog	1000 [pkts]
Parametry Rsyslog	imudp:threads	1
	imudp:batchSize	32 [pkts]
	imudp:TimeRequery	2
	global:net.enableDNS	on

Tabulka 3.26: Identifikované optimalizační parametry

Zvolená metodika testování opět využívá "pcap" data ze vzorku číslo 2 zachyceného na produkčním serveru. Data jsou přehrávána ze serveru **pm2** oproti serveru **rsyslog2**, přičemž je opět prováděno měření za pomocí nástrojů "iptables" a "netstat", jsou sbírány statistiky z Rsyslog procesu. Server **rsyslog2** je nakonfigurován v souladu s produkčním serverem, kdy jsou definovány tři destinace - SIEM, externí destinace číslo 1 a externí

destinace číslo 2. Jako SIEM destinace slouží lokálně nainstalovaný ArcSight Load Balancer, na který jsou události přeposílány prostřednictvím "omudpspoof" výstupu. Ten následně přeposílá události na deset SmartConnectorů běžících na serveru **connector1**. Další dvě destinace slouží pro simulaci dalších cílů jako v produkčním prostředí. V případě tohoto laboratorního prostředí jsou události pro obě destinace směrovány na server **syslog4others** a doručovány prostřednictvím běžného omfwd výstupu zde běžícímu procesu Rsyslog. Pro odlišení využívá každá s destinací jiný UDP port. Data jsou na tomto serveru ukládána na disk. Počet odeslaných událostí všem destinacím je také sledován na serveru **rsyslog2** pomocí "iptables". Nad to je ještě celý systém monitorován pomocí nástroje "sar" a ve vteřinových intervalech jsou sbírány informace o stavu UDP bufferu z ukazatele `/proc/net/udp`.

Pro testování byl zvolen postup, kdy se nejprve testovaly různé varianty parametrů jádra. Po nalezení vhodné sady parametrů pak došlo na optimalizaci parametrů aplikace. Cílem procesu optimalizace bylo dosáhnout na průměrnou ztrátovost 5% nebo lepší při obvyklém počtu událostí za vteřinu (EPS). Tabulka 3.27 zobrazuje již výsledné zprůměrované hodnoty ztrát vyčíslené v procentech. Všechna data lze pak najít v přílohách.

Jako optimální se podle testů jeví následující sada parametrů, při jejichž užití se podařilo stáhnout průměrnou ztrátovost na 4.7% při přibližné rychlosti 38000 EPS. Ani v laboratorních podmínkách nedocházelo k přetížení CPU, pakliže byly sledovány hodnoty "load average", které se pohybovaly pod kritickou hranicí 12.

Vzhledem k získaným hodnotám a k tomu, že se potvrdilo, že modul "omudpspoof" není ideální co do výkonu, byly zvažovány některé další alternativy. První uvažovanou možností bylo nahrazení Rsyslogu alternativním řešením. V tomto směru byly otestovány další dva produkty sloužící pro příjem syslog událostí s možností jejich přeposílání pomocí UDP spoof metody. Jednalo se o open source systém Syslog-ng a komerční software NXlog. Oba produkty jsou dostupné ve volné verzi - Syslog-ng OSE a NXlog CE (community edition). Pro otestování byla z hlediska funcionality verze Syslog-ng OSE dostačující, ale v případě NXlog bylo nutné použít zkušení verzi komeční implementace NXlog Enterprise Edition, jelikož NXlog CE nedisponuje UDP spoof výstupem.

Tabulka 3.27: Optimalizace parametrů

Test	Ztráta rsyslog	Ztráta ArcLB	Ztráta celkem
1	20.5	0.3	20.7
2	16.7	0	16.7
3	16.7	0	16.7
4	16.4	0	16.4
5	16	0	16
6	9.8	0.2	10
7	9.5	0	9.5
8	9.5	0	9.5
9	9.3	0	9.3
10	7	0.1	7.1
11	7.3	0.1	7.3
12	7.7	0	7.7
13	9.5	0	9.6
14	6.3	0	6.3
15	6.8	0	6.8
16	6.5	0	6.5
17	7.2	0	7.2
18	7.3	0.1	7.4
19	7.6	0	7.6
20	4.6	0.1	4.7
21	10.1	0.1	10.1

Metodika testování byla podobná předchozím testům, avšak vzhledem k tomu, že bylo primárním cílem otestovat výstup metodou UDP spoof, byla konfigurace maximálně zjednodušena co se pravidel či filtrů týče. To především z toho důvodu, že každý z produktů nabízí více možností jak k pravidlům a filtrům přistoupit, přičemž jednotlivé přístupy mohou mít odlišný dopad na celkový výkon při zpracování událostí. Implementace produkčních filtrů a pravidel by tak vnášela do výsledku testů nepredikovatelné vlivy. Opět byl využit "pcap" záznam jako v předchozích testech přehrávaný rychlostí 38000 EPS, přičemž každý syslog server byl nakonfigurován tak, aby vše na vstupu předal na výstup. Při testování bylo zvoleno několik typů výstupů. Testoval se zápis do souboru, přeposílání běžným způsobem (UDP) v rámci localhostu, přeposílání běžným způsobem na vzdálenou destinaci, přeposílání pomocí UDP spoof v rámci localhostu a přeposílání pomocí UDP spoof na vzdálenou destinaci.

Parametry jádra OS	net.core.rmem_max	16777216 [B]
	net.core.wmem_max	16777216 [B]
	net.core.netdev_max_backlog	1000 [pkts]
Parametry Rsyslog	imudp:threads	3
	imudp:batchSize	128 [pkts]
	imudp:TimeRequery	12
	global:net.enableDNS	off

Tabulka 3.28: Optimální parametry Rsyslogu a OS

Tabulka 3.29: Verze testovaných syslog produktů

NXlog	4.5.4503
Rsyslog	8.2002.0
Syslog-ng	3.25.1

Ke sledování byly opět použity nástroje “iptables” a “netstat”, v případě zápisu do souboru i nástroj “wc” a pakliže byly dostupné i statistiky o průběhu zpracování v syslog programu. Ty bohužel nebyly dostupné v případě NXlogu. Při testech byly použity stejné jaderné parametry, které byly vyhodnoceny jako nejvhodnější v předchozích testech Rsyslogu. Všechny syslog produkty byly z důvodu objektivnosti testování hostovány na stejném virtuálním serveru **rsyslog2**. Byly však využity nejnovější dostupné varianty syslog produktů jak ukazuje tabulka 3.29. Konfigurace všech tří syslog produktů byla optimalizována co nejvhodněji pro testovaný případ - konfigurace lze najít v přílohách. Jako síťové destinace byly pak využity ArcSight Load Balancer běžící lokálně na server **rsyslog2** a jako vzdálená destinace byl použit server **receiver**, kde byl nakonfigurován Rsyslog pro příjem událostí a zápis do souboru.

Výsledky testů nejsou zcela očekávané. Zatím co dokumentace Rsyslogu vysloveně varuje, že výkonnost modulu omudpspoof je nižší než výkonnost běžného přeposílání, dopadly výsledky pro Rsyslog velmi dobře. Naopak výsledky Syslog-ng při použití UDP spoofingu byly o poznání horší, zejména, pokud jde o zasílání v rámci localhostu. NXlog vykazoval mírou ztrátovost při všech testech, ale nepodařilo se identifikovat, čím je toto chování způsobeno. Úpravy konfigurace pro zlepšení výkonu při zpracování UDP, které jsou doporučeny v dokumentaci vedly k opačnému efektu a výsledné hodnoty výrazně zhoršily.

Tabulka 3.30: Naměřené ztráty při testu udp spoof modulu

Produkt	**Metoda**	**ztráty** [%]
Nxlog	file	0.29
Nxlog	udp local	4.15
Nxlog	udp remote	6.1
Nxlog	udpspoof local	4.13
Nxlog	udpspoof remote	4.92
Rsyslog	file	0
Rsyslog	udp local	0
Rsyslog	udp remote	0
Rsyslog	udpspoof local	0
Rsyslog	udpspoof remote	0
Syslog-ng	file	0
Syslog-ng	udp local	0
Syslog-ng	udp remote	0
Syslog-ng	udpspoof local	18.34
Syslog-ng	udpspoof remote	2.99

V případě Rsyslogu byly provedeny doplňující testy s postupným zvyšováním EPS. Nulové ztráty se podařilo ověřit do přibližně 43000 EPS. S vyššími hodnotami se začal projevovat problém, že část přehrávaných datagramů vůbec nebyla virtuálnímu serveru doručena. Původ problému není jasný a nepodařilo se dohledat ani žádné počítadlo, které by reflektovalo nějaké ztráty či chyby na síťové vrstvě. Nicméně z hlediska otestování výkonu byla dosažená hodnota vyhovující.

Výrazně dobrý výsledek Rsyslogu v tomto testu indikoval, že zde možná existuje problém ve stávající konfiguraci, který zapřičiňuje ztráty. Jak již bylo udeveno, Rsyslog jednak přeposílá téměř všechny události ArcSight Load Balanceru, ale zároveň posílá vybrané události dalším systémům. Při dalších testech tedy byla přidána konfigurace zajišťující přeposílání do SIEM, ale přeposílání dalším systémům bylo vynecháno. I tyto testy vykázaly nulové ztráty.

Dle provedených testů se jako problematická jevila část konfigurace zajišťující přeposílání dalším systémům. Jednalo se o dva typy destinací používající běžné přeposílání pomocí omfwd modulu protokolem UDP. Obě části konfigurace byly implementovány za pomocí asynchronních front, což v praxi znamená, že každá syslog událost byla do takové fronty zduplikována a zpracovávána nezávisle na hlavní frontě, která se starala o doručování do SIEM. Důvodem pro tuto implementaci byly historické problémy s nedostupností

destinací. Byť se jedná o posílání dat pomocí UDP, čili negarantovaného spojení, může nastat problém, pakliže destinace je ve stejné síti (broadcast domain) a neodpovídá. V ten moment není možné získat ARP záznam a Rsyslog zastaví doručování dané destinaci a začne ukládat události do cache, přičemž se stále snaží získat ARP pro danou destinaci. Jelikož je ale zpracování hlavní fronty sekvenční, dojde tím k ovlivnění i doručování dalším destinacím. Jako prevence takové situaci bylo doručování odděleno do samostatných asynchronně zpracovávaných front, což ale v kontextu postupného nárůstu událostí přicházejících do systému vedlo nakonec k problémům se ztrátami. Po úpravě konfigurace Rsyslogu a začlenění doručování dalším systémům do hlavní fronty se podařilo docílit nulové ztrátovosti.

Posledním problémem zpracovávání syslog událostí byla neschopnost Rsyslogu detekovat dostupnost ArcSight Load Balanceru. Bohužel Rsyslog nedisponuje žádnou možností jak realizovat takovou detekci při využití UDP protokolu. ArcSight Load Balancer sice nabízí možnost běhu v režimu vysoké dostupnosti, ale jak již bylo zmíněno, jedná se pouze o režim aktivní/pasivní. Sice by bylo možné provozovat jeden či více clusterů ArcSight Load Balancerů v takové konfiguraci na dedikovaných serverech, ale škálovatelnost takového řešení není zrovna ideální. Zásadní nevýhodou se také jeví nutnost restartovat při změně konfigurace celý cluster.

Jako vhodnější řešení se jevila možnost přesunout ArcSight Load Balancery na dedikované servery, dál je provozovat v samostatném režimu a mezi ně a Rsyslog servery přidat další vrstvu load balanceru. Toto řešení sice vnáší vyšší komplexnost do celé soustavy, na druhou stranu přináší benefit v podobě jednoduchého horizontálního škálování ArcSight Load Balancerů za současného zajištění vysoké dostupnosti.

Prvotní testy byly v laboratorním prostředí prováděné s Pulse Secure Virtual Traffic Managerem, který je dostupný jako softwarový produkt a lze jej provozovat omezeně v režimu vývojáře zdarma. Z důvodu výkonu byl ale nahrazen jaderným load balancerem IPVS, který je nativně dostupný v Linuxovém jádře. Díky tomu nemusí být pakety předávány do userspace a výkonnost takového řešení dramaticky roste. Při testech byl pro hostování IPVS load balanceru plně postačující server s dvěma vCPU a nedocházelo k žádným ztrátám. Pro zajištění vysoké dostupnosti IPVS load balanceru je zapotřebí doplnit jej o službu Keepalived, která standardně součástí Red Hat Enterprise Linuxu a dalších

Linuxových distribucí. Jedná se o routovací daemon, který využívá protokol VRRP pro zajištění vysoké dostupnosti sdílené IP adresy. Zároveň umožnuje dynamicky obsluhovat IPVS load balancer na základě stavu koncových (real) serverů. Testování dostupnosti koncových serverů je definovatelné a nezávislé na protokolu a portu, který je balancovaný. Lze tedy provádět detekci stavu ArcSight Load Balanceru pomocí TCP a v případě jeho vypnutí či pádu je datový tok automaticky přesunut na jiný dostupný ArcSight Load Balancer. Detaily konfigurace celého řešení lze vidět v příloze.

3.5.2 Windows

Pro optimalizaci sběru událostí z Windows serverů byl na testovací server s OS Windows 2016 nainstalován WiNC SmartConnector a nástroj NetBalancer, který umožňuje sledování datového toku k jednotlivým procesům. Jelikož WiNC SmartConnectory bývají obvykle nakonfigurovány na sběr událostí ze 100 serverů, byl pro testování jako reprezentativní vzorek identifikován produkční SmartConnector s průměrnou příchozí hodnotou EPS, která činila 2300 EPS, přičemž SmartConnector byl nakonfigurován na sběr událostí právě ze 100 Windows serverů různých rolí. Jeho konfigurace tedy byla přenesena na testovací SmartConnector, aby nedošlo během testů k ovlivnění sběru událostí v produkčním prostředí.

Jak již bylo zmíněno, WiNC SmartConnectory jsou na výstupu vybaveny filtrem, který zahazuje některé typy událostí podle položky EventID. Jedná se o události, které nemají zásadní význam z hlediska bezpečnosti a neexistuje pro ně žádné pravidlo v SIEM. Předně jde o události z kanálu Systém, který zahrnuje například i události antivirového programu MS Antimalware / Defender nebo vybrané události z kanálu Security. Konkrétní odfiltrované události vycházejí ze standardu společnosti. WiNC SmartConnector na rozdíl od staršího typu WUC SmartConnectoru nabízí možnost definice filtru aplikovatelného na straně zdrojového serveru.

Při převádění filtru ze stávající podoby na podobu kompatibilní s Windows Eventing API vyvstaly dva problémy související s možnostmi filtrování v rámci WiNC SmartConnectoru. MS Windows standardně nabízí přístup k událostem prostřednictvím aplikace

Tabulka 3.31: Filtrování Windows událostí

Kanál	**Typ filtru**	**Podmínky**
System	inkluzivní	pouze události Microsoft Antimalware
Security	exkluzivní	vybrané události dle EventID
Windows Defender	inkluzivní	všechny události Windows Defender

Event Viewer. Zde je možné jednak prohlížet události a také definovat filtry. Aplikace nabízí jednoduchý způsob definice filtru pomocí zadání např. vybraných EventID, které má nebo naopak nemá zobrazit. Pakliže navolíme filtr, je možné následně zobrazit jej v XML kódu, který využívá API. Filtr může vypadat například následovně:

```
<QueryList>
  <Query Id="0" Path="Application">
    <Select Path="Application">*[System[(EventID=63 or EventID=16394)]]</Select>
    <Select Path="System">*[System[(EventID=63 or EventID=16394)]]</Select>
    <Suppress Path="Application">*[System[(EventID=210)]]</Suppress>
    <Suppress Path="System">*[System[(EventID=210)]]</Suppress>
  </Query>
</QueryList>
```

V tomto případě je požadováno zobrazení událostí s EventID 63 a 16394 z kanálů Aplikace i Systém a zároveň je požadováno nezobrazení události s EventID 210. Jelikož oba filtry inkluzivní i exkluzivní jsou volány nad stejnými kanály, je pochopitelně exkluzivní filtr nadbytečný. GUI rozhraní Event Vieweru nedisponuje logikou pro definici složitějších pravidel. To je možné napřímo pomocí XML. Lehkou úpravou lze docílit toho, že filtr bude inkluzivní např. pro kanál Aplikace a naopak exkluzivní pro kanál Systém. Dva zmiňované problémy tkví v tom, že WiNC SmartConnector nedisponuje možností definovat "Suppress Path" a umožňuje definovat pouze jednu "Select Path". Je tedy nezbytné definovat požadované chování jedním výrazem. Výsledný filtr pro WiNC SmartConnector má tedy podobu:

```
*[System[Provider[@Name='Microsoft-Windows-Security-Auditing']
         and (EventID!=0001 and EventID!=0002 and EventID!=0003)]]
or *[System[Provider[@Name='Microsoft Antimalware'] and Channel='System']]
```

```
or *[System[Provider[@Name='Microsoft-Windows-Windows Defender']]])
```

Výsledný filtr jednak získává všechny události z kanálu Security s výjimkou událostí se specifikovanými EventID. Dále zahrnuje všechny události od Microsoft Antimalware, které jsou v kanálu System a také události Windows Defenderu v samostatné skupině. Po aplikaci tohoto filtru v testovacím prostředí byly zaznamenány následující změny ve sledovaných průměrných hodnotách:

Tabulka 3.32: Porovnání toku dat Windows

Ukazatel	**Před filtrování na zdroji**	**Po filtrování na zdroji**
EPS	2300	870
Datový tok	56 Mb/s	24 Mb/s

Naměřené údaje vypovídají zhruba 60% snížení počtu událostí stejně jako využitého datového pásma. Jedná se však pouze o snížení u vybraných 100 zdrojů a ve sledovaném období. Skladba a množství událostí se v čase mění stejně jako se výrazně liší dle rolí serverů. Takto dramatické snížení lze připisovat tomu, že v rámci testování byl prováděn monitoring během pracovního dne, kdy je počet událostí na vrcholu. Software NetBalancer navíc nedisponuje možností dlouhodobého měření průměrného datového toku na úrovni jednotlivých procesů, takže údaje o datovém toku nejsou zcela přesné, ale spíše indikativní. Hodnotu EPS bude možné sledovat dlouhodobě právě v systému SIEM a následně ji vyhodnotit.

Aplikace filtru na straně zdrojových serverů umožní díky nižší EPS i snížení počtu WiNC SmartConnectorů a tím dojde i k úspoře co do počtu paralelních připojení na další komponenty SIEM. Další možností je pak rozšíření stávajícího modelu ArcSight SmartConnector -> ArcSight ESM o další úroveň, tedy o jakési koncentrátory. Zde je možné využít toho, že ArcSight SmartConnector nabízí jako jednu z možný destinací CEF Syslog. Lze tedy vložit další, tentokrát Syslog SmartConnector mezi WiNC SmartConnector a ESM. CEF syslog destinace podporuje i přenos pomocí TCP, případně i TLS. CEF syslog destinace disponuje volbou "Forwarder", která zajistí, že v těle CEF události bude uchována informace o původním SmartConnectoru, který provedl parsování, byť bude událost doručena SmartConnectorem koncentrující události z více WiNC SmartConnectorů.

3.5.3 Single point of failure

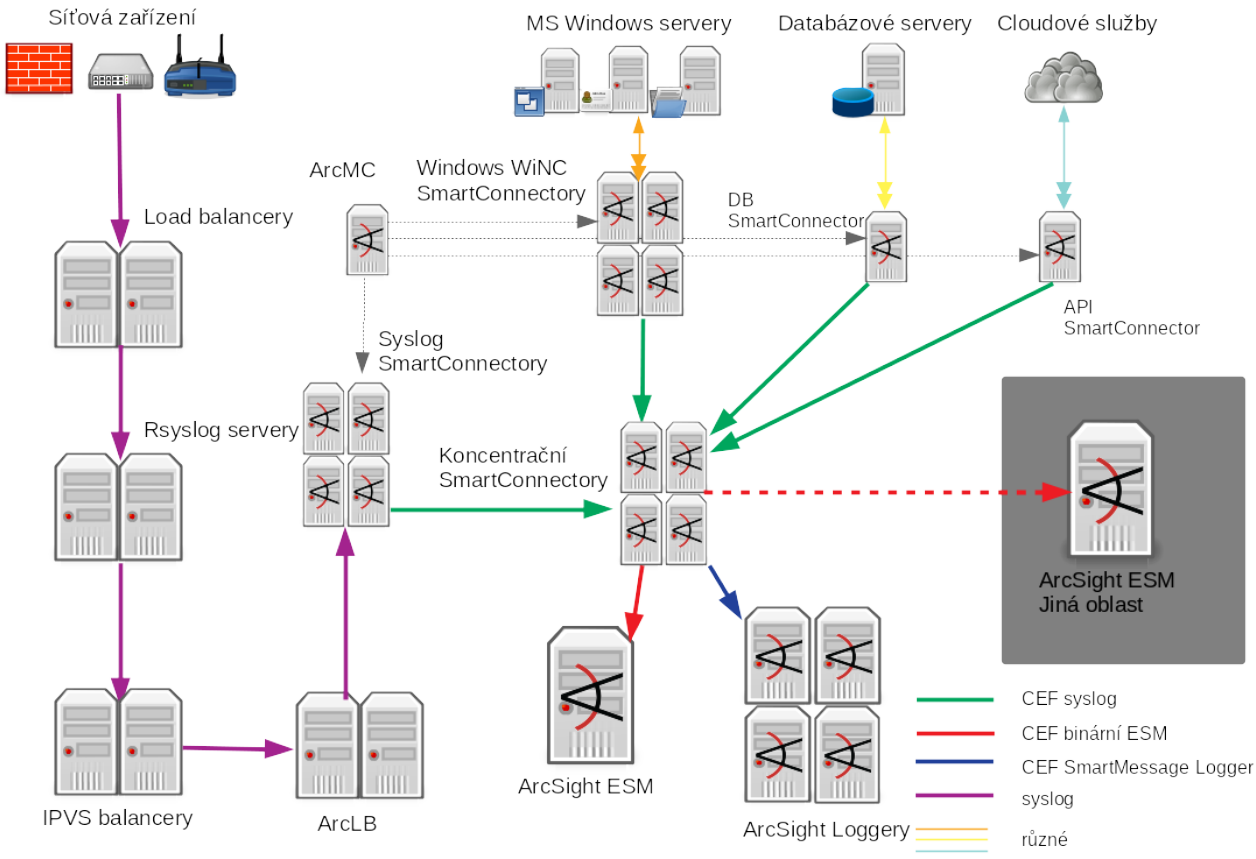
Nedostupnost oblastního ESM není takovým rizikem z hlediska ztráty událostí, jelikož SmartConnector dokáže po čas nedostupnosti ESM ukládat události do cache a doručit je později. Je to ale problém z hlediska bezpečnosti, jelikož události nejsou vyhodnocovány v reálném čase a výpadek dostupnosti ESM může v nejhorším případě být i symptomem probíhajícího útoku. Z toho důvodu by bylo optimální mít záložní řešení, jak zajistit zpracování událostí v reálném čase ať už při plánovaném či neplánovaném výpadku ESM.

Jednou z možností je provoz ESM v režimu HA, čili vysoké dostupnosti. Tato funkcionalita není standardně součástí a je nutné ji v případě zájmu zalicencovat a dokoupit. Další možností jak tento problém řešit navazuje na koncept zmíněný v části o zpracování událostí ze systémů Windows. Jedná se o využití koncentračních SmartConnectorů. S jejich implementací se totiž zmenšuje celkový počet konektorů, které komunikují s ESM. SmartConnector disponuje možností definovat pro destinaci i záložní destinaci, kam lze události předávat v případě nedostupnosti primární destinace. S menším počtem koncentračních SmartConnectorů lze tedy snadněji realizovat záložní konektivitu k ESM v jiné oblasti.

Pakliže na vrstvě koncentračních SmartConnectorů definujeme záložní ESM destinaci, budou v případě výpadku primárního ESM všechny události doručované záložnímu ESM. Ve výchozím stavu bude ale SmartConnector zároveň ukládat všechny události do cache pro pozdější doručení primárnímu ESM. To by ale znamenalo, že následně dojde k duplicitnímu zpracování událostí a znovu dostupné ESM vygeneruje alerty, které již byly vygenerovány ESM v záložní oblasti. Tomu lze předejít nastavením módu cache na variantu “Drop if Dest Down”. V takovém případě nebudou události pro primární ESM drženy v cache a budou pouze posílány na záložní ESM. Tento přístup lze v případě potřeby zkombinovat s ESM běžící v HA režimu.

3.5.4 Návrh nového designu

Při postupné analýze celého prostředí a jednotlivých komponent vycházelo najevo, že nový design rozhodně nebude jednodušší, jak autor původně doufal. Naopak bude robustnější, ale měl by postihnout většinu stávajících problémů a slabin.



Obrázek 3.7: Nový diagram oblasti (zdroj: vlastní)

Diagram reflektuje navržené změny při zpracování toku dat typu syslog. Díky tomu je celá soustava odolnější vůči výpadkům. Mezi zdrojové SmartConnectory a Loggery a ESM je vložena další úroveň koncentračních SmartConnectorů. Díky tomu je možné snížit objemy dat z jednotlivých lokalit, jelikož data nejsou posílána duplicitně ESM a Loggerům, jak tomu bylo v případě původního řešení. Koncentrační SmartConnectory by měli být umístěny v uzlech s dobrou síťovou konektivitou, avšak ne nutně všechny v jedné lokaci. Jejich rozvržení by mělo být odvozeno od geografického rozložení zdrojů v dané oblasti a dostupné konektivitě včetně např. záložních tras. S ohledem na nutnou konektivitu s ESM v jiné oblasti sloužící jako záložní by ale nemělo být rozložení koncentračních SmartConnectorů příliš fragmentované.

Vhodný postup realizace změn vychází z jejich kritičnosti. Na prvním místě je tak realizace úprav na soustavě komponent zpracovávajících syslog události, aby se tak předešlo dalším ztrátám událostí a případným problémům při výpadku nějaké komponenty,

která není dostupná v režimu vysoké dostupnosti. Dalším krokem by měly být úpravy při sběru dat ze zařízení s OS Windows z důvodu snížení zátěže, počtu SmartConnectorů a datového toku nutného pro jejich získání. Na to navazuje vytvoření vrstvy koncentračních SmartConnectorů. Poslední fází je pak zajištění zasílání událostí záložnímu ESM v jiné oblasti v případě výpadku primárního ESM.

4 Závěr

Hlavním cílem práce bylo zhodnocení stávající platformy SIEM z pohledu správce řešení s ohledem na aktuální potřeby společnosti a navržení vhodné architektury celého řešení. V rámci této analýzy byly vyhodnoceny možnosti integrace různých typů zařízení, která aktuálně jsou, nebo by vzhledem k pozici na trhu do budoucna mohla být společností užívána a tudíž lze předpokládat jejich integraci s řešením SIEM. Bylo vytvořeno portfolio zařízení a produktů různého zaměření, u nichž byly zkoumány možnosti integrace s nejrozšířenějšími produkty v segmentu SIEM.

V rámci porovnávání možností integrace byla vyhodnocena i její komplexnost, která má vliv na budoucí správu a byla uvažována i škálovatelnost řešení. Na základě zjištěných poznatků proběhlo vyhodnocení různých SIEM platforem a to především s ohledem na technické aspekty řešení. Jako nejlépe vyhovující platformou s ohledem na rozsah podpory různých typů produktů a zařízení z portfolia a s ohledem na celkovou strategii společnosti byla vyhodnocena platforma Splunk, přesněji ve variantě Splunk Cloud. Stávající platforma ArcSight sice zaujala v hodnocení až třetí pozici, ale rozdíly mezi porovnávanými platformami z hlediska sledovaných parametrů byly minimální. Z toho důvodu bylo doporučeno setrvání u současného řešení, jelikož náklady nutné na změnu řešení by nejspíše převyšovaly benefity získané z přechodu na novou platformu. Toto doporučení se opírá hlavně o technologický aspekt, protože cenotvorba v případě velkých společností je především o dlouhém vyjednávání. Navíc v případě segmentu SIEM je i ze strany výrobců řešena individuálně.

V souladu s vyhodnocením byly dále řešeny nedostatky současné architektury. Zde bylo identifikováno několik problémů. Za hlavní problém lze označit náchylnost původního řešení pro zpracování událostí přicházejících prostřednictvím protokolu syslog na ztráty při vyšší zátěži. Dalším identifikovaným problémem byly komponenty v řetězci zpracovávající syslog zprávy, které nebyly provozovány v režimu vysoké dostupnosti. Jako další problém byla identifikována nevhodná metoda přístupu k filtrování událostí pocházejících ze systémů s OS Windows, kde docházelo k filtrování až na úrovni SmartConnectorů. To mělo za následek zbytečně vysoký počet událostí posílaných ze serverů směrem ke SmartConnectorům.

Díky testování a prověřování všech aspektů konfigurace Rsyslog koncentrátorů a jejich alternativ se povedlo odhalit výkonnostní problém zavlečený s dobrým úmyslem do konfigurace mnoho let nazpět, který byl zodpovědný za ztráty UDP datagramů. V rámci toho byly otestovány další produkty s obdobnou funkcionalitou, nicméně použitý Rsyslog se zdá být v tomto směru optimální volbou. Díky testování byl identifikován i optimální způsob řešení vysoké dostupnosti za pomocí souboru Keepalived/IPVS jako mezivrstvy mezi Rsyslog koncentrátory a ArcSight Load Balancery.

Na základě všech informací, výsledků testů a doporučení byl vytvořen nový diagram architektury celého řešení, který reflektuje všechny odhalené problémy, slabiny či nedostatky. V souladu s tím byl navržen i postup realizace jednotlivých opatření, který zohledňuje jejich logické provázání.

Použité zdroje

Tištěné zdroje

Gordon, A. *Official (ISC)2 Guide to the CISSP CBK, Fourth edition*. New York: CRC Press, 2015. ISBN 978-1-4822-6275-9.

Stewart, J. M., Chapple, M., Gibson, D. *CISSP Study Guide*. Indianapolis: John Willey & Sons, Inc., 2015. ISBN : 978-1-119-04271-6.

McClure, S., Scambray, J., Kurtz, G. *Hacking Exposed 7: Network Security Secrets & Solutions*. New York: McGraw-Hill, 2012. ISBN 978-0-07-178028-5.

Miller, D. R., Harris, S., Harper, A. A., Vandyke, S., Blask, C. *Security Information and Event Management (SIEM) Implementation*. New York: McGraw-Hill/Osborne, 2011. ISBN 978-0071701099.

Conklin, W. A., White, G., Cothren, C., Davis, R. L., Williams, D. *CompTIA Security+ All-in-One Exam Guide, Fifth Edition*. New York: McGraw-Hill/Osborne, 2018. ISBN 978-1260019322.

Dostálek, L. a kolektiv *Velký průvodce protokoly TCP/IP: Bezpečnost 2. aktualizované vydání*. Praha: Computer Press, 2003. ISBN 80-7226-849-X.

Dye, M., McDonald, R., Rufi, A. *Network Fundamentals, CCNA Exploration Companion Guide*. Indianapolis: Cisco Press, 2007. 560 s. ISBN-10: 1-58713-208-7.

McQuery, S. *CCNA Preparation Librarty 7th edition*. Indianapolis: Cisco Press, 2008. ISBN-10: 1-58705-462-0.

Paquet, C. *Implementing Cisco IOS Network Security (IINS): (CCNA Security exam 640-553) (Authorized Self-Study Guide)*. Indianapolis: Cisco Press, 2009. 624 s. ISBN-10: 1-58705-815-4

Strebe, M., Perkins, Ch. *Firewally a proxy-servery Praktický průvodce*. Brno: Computer Press, 2003. 450 s. ISBN 80-7226-983-6.

Elektronické zdroje

Gemalto NV *Breach Level Index* [online]. [cit. 2018-08-26]. URL: <https://www6.gemalto.com/e/51442/ort-h1-2017-gemalto-report-pdf/93zlf9/729228999>

Gartner Inc. *Gartner Says Worldwide Security Software Market Grew 3.7 Percent in 2015* [online]. [cit. 2018-08-26]. URL: <https://www.gartner.com/newsroom/id/3377618>

Gartner Inc. *Gartner Magic Quadrant for SIEM Products (2017, 2016, 2015, 2014, 2013, 2012, 2011, 2010)* [online]. [cit. 2018-09-02]. URL: <https://www.51sec.org/2017/07/gartner-magic-quadrant-for-siem-products-2016-2015-2014-2013-2012-2011-2010/>

Hewlett Packard Enterprise Development *HPE Security ArcSight ESM 101* [online]. [cit. 2018-10-15]. URL: <https://community.softwaregrp.com/t5/ESM-and-ESM-Express-Previous/ESM-101-ESM-7-0/ta-p/1641759?attachment-id=67087>

Rainer Gerhards and Others *omudpspoof: UDP spoofing output module* [online]. [cit. 2018-12-12]. URL: <https://www.rsyslog.com/doc/v8-stable/configuration/modules/omudpspoof.html>

Microsoft *Windows Events* [online]. [cit. 2018-12-13]. URL: <https://docs.microsoft.com/en-us/windows/desktop/events/windows-events>

Micro Focus *WiNC using native Windows Eventing API* [online]. [cit. 2018-12-14]. URL: <https://community.softwaregrp.com/t5/Archive-Discussion-Board/WinC-We-switched-from-WUC-We-know-WinC-uses-native-Windows/m-p/1562238#64903>

Micro Focus *Micro Focus Security ArcSight Connectors: SmartConnector User Guide* [online]. [cit. 2019-02-15]. URL: <https://community.softwaregrp.com/t5/ArcSight-Connectors/ArcSight-SmartConnector-User-Guide-7-11-0/ta-p/1586784?attachment-id=70024>

Hewlett Packard Enterprise *Announcing HPE Security ArcSight Data Platform solution* [online]. [cit. 2019-02-15]. URL: <http://www.wit.co.th/hp/resources/HPE%20Security%20ArcSight%20Data%20Platform%20sol

Micro Focus *Micro Focus Security ArcSight Event Broker: Administrator's Guide* [online]. [cit. 2019-02-15]. URL: <https://community.softwaregrp.com/t5/Event-Broker/Event-Broker-2-21-Administrator-s-Guide-updated-9-27-18/ta-p/1660015?attachment-id=69034>

Micro Focus *ArcSight Data Platform Data Sheet* [online]. [cit. 2019-02-16]. URL: <https://www.microfocus.com/media/data-sheet/arcsight_data_platform_ds.pdf>

O'Reilly *Eric Allman* [online]. [cit. 2019-04-14]. URL: <https://www.oreilly.com/pub/au/359>

Gartner Inc. *Palo Alto Networks an Eight-Time Gartner Magic Quadrant Leader* [online]. [cit. 2019-12-02]. URL: <https://blog.paloaltonetworks.com/2019/09/network-gartner-magic-quadrant-leader/>

Cisco Systems, Inc. *Cisco ASA Series Syslog Messages* [online]. [cit. 2019-12-08]. URL: <https://www.cisco.com/en/us/td/docs/security/asa/syslog/b_syslog/about.html>

Lonvick, C. *The BSD syslog Protocol* [online]. [cit. 2018-12-11]. URL: <https://tools.ietf.org/html/rfc3164>

Gerhards, R. *The Syslog Protocol* [online]. [cit. 2018-12-11]. URL: <https://tools.ietf.org/html/rfc5424>

Ma, Y., Miao, F., Salowey, J. *Transport Layer Security (TLS) Transport Mapping for Syslog* [online]. [cit. 2018-12-11]. URL: <https://tools.ietf.org/html/rfc5425>

New, D., Rose, M. *Reliable Delivery for syslog* [online]. [cit. 2018-12-11]. URL: <https://tools.ietf.org/html/rfc3195>

Gerhards, R., Lonvick, C. *Transmission of Syslog Messages over TCP* [online]. [cit. 2018-12-11]. URL: <https://tools.ietf.org/html/rfc6587>

The kernel development community *The Linux Kernel documentation* [online]. [cit. 2019-11-11]. URL: <https://www.kernel.org/doc/html/latest/>

Fortinet, Inc. *FortiOS 6.0 Online Help* [online]. [cit. 2019-11-16]. URL: <https://help.fortinet.com/fos60hlp/60/Content/FortiOS/fortiOS-HTML5-v2/Home.htm>

Forcepoint LLC *How to forward SMC log and audit data to external syslog or SIEM servers* [online]. [cit. 2019-11-17]. URL: <https://support.forcepoint.com/KBArticle?id=000015002>

Cisco Systems, Inc. *Cisco ASA Series CLI Configuration Guide, 9.0* [online]. [cit. 2019-11-19]. URL: <https://www.cisco.com/c/en/us/td/docs/security/asa/asa90/configuration/guide/asa_90_cli_

Palo Alto Networks, Inc. *Forward Logs to an HTTP/S Destination* [online]. [cit. 2019-11-19]. URL: <https://docs.paloaltonetworks.com/pan-os/9-1/pan-os-admin/monitoring/forward-logs-to-an-https-destination>

Palo Alto Networks, Inc. *Configure Log Forwarding from Panorama to External Destinations* [online]. [cit. 2019-11-19]. URL: <https://docs.paloaltonetworks.com/panorama/9-1/panorama-admin/manage-log-collection/configure-log-forwarding-from-panorama-to-external-destinations.html>

Juniper Networks, Inc. *Configuring Routing Rules to Forward Data* [online]. [cit. 2019-11-20]. URL: <https://www.juniper.net/documentation/en_US/jsa7.3.1/jsa-administration-guide/topics/task/operational/jsa-admin-configuring-routing-rules-for-bulk.html>

Sophos Ltd. *Sophos Central APIs: How to send alert and event data to your SIEM* [online]. [cit. 2019-11-21]. URL: <https://community.sophos.com/kb/en-us/125169>

IT Central Station *Security Information and Event Management Buyer's Guide and Reviews January 2020* [online]. [cit. 2020-02-11]. URL: <https://www.itcentralstation.com/landing/qlfd-report-security-information-and-event-management-siem>

Přílohy

Konfigurace Rsyslog pro testování UDP spoof

```
#### RSYSLOG v7 CONFIGURATION FILE ####

# For more information see /usr/share/doc/rsyslog-*/rsyslog_conf.html
# If you experience problems, see http://www.rsyslog.com/doc/troubleshoot.html

### DEBUG ###

$DebugFile /opt/logs/rsyslog-debug.log
$DebugLevel 1

#### MODULES ####

module(load="impstats"
        interval="10"
        severity="7"
        resetCounters="off"
        log.syslog="off"
        log.file="/opt/logs/rsyslog-stats.log")

global(net.aclResolveHostname="off")
global(net.enableDNS="off")

# Provides UDP syslog reception with multi-thread

module(load="imudp"
        TimeRequery="12"
        threads="3"
```

```
        batchSize="128") # needs to be done just once

input(type="imudp" Address="10.200.200.120" port="514")

# UDP spoofing output module to keep original IP when forwarded to ArcLB
module(load="omudpspoof")

# Set the default permission for all log files
$Umask 0000
$FileCreateMode 0644
$DirCreateMode 0755

$WorkDirectory /opt/rsyslog-cache

$OMFileAsyncWriting on
$OMFileIOBufferSize 1000k

main_queue(
    queue.size="600000"   # capacity of the main queue
    queue.type="FixedArray"
    queue.filename="00mainq"
    queue.dequeueBatchSize="10000"
    queue.highwatermark="550000"
    queue.lowwatermark="350000"
    queue.maxdiskspace="10g"
    queue.saveonshutdown="on"
    queue.workerThreads="12"
)

#### GLOBAL DIRECTIVES ####
# Include rules from rules file.
```

```
#$IncludeConfig /etc/rsyslog.d/rules-rsyslog-Group00.conf

#*.* action (name="siemfile" type="omfile" file="/opt/logs/siem-rsyslog.log")

#*.* action(name="siemlocal" type="omfwd" target="127.0.0.1" port="5140" template="

#*.* action(name="siemremote" type="omfwd" target="10.200.200.200" port="5140" temp

#*.* action(name="siemspooflocal" type="omudpspoof" target="127.0.0.1" port="5140"

*.* action(name="siemspoofremote" type="omudpspoof" target="10.200.200.200" port="5
```

Konfigurace Syslog-ng pro testování UDP spoof

```
@version: 3.25
@include "scl.conf"

# Syslog-ng configuration file, compatible with default Debian syslogd
# installation.

# First, set some global options.
options { chain_hostnames(off);
          flush_lines(0);
          use_dns(no);
          use_fqdn(no);
          dns_cache(no);
          owner("root");
          group("adm"); perm(0640);
          stats_freq(0);
          bad_hostname("^gconfd$");
          threaded(yes) ;
```

```
};

source s_udp {
    network(
        port("514")
        transport("udp")
    );
};

destination d_siem_udp {
    network(
        "127.0.0.1"
        transport("udp")
        port(5140)
        spoof-source(no)
    );
};

destination d_siem_udp_remote {
    network(
        "10.200.200.200"
        transport("udp")
        port(5140)
        spoof-source(no)
    );
};

destination d_siem_udp_spoof {
    network(
        "127.0.0.1"
        transport("udp")
```

```
        port(5140)
        spoof-source(yes)
    );
};

destination d_siem_udp_spoof_remote {
    network(
        "10.200.200.200"
        transport("udp")
        port(5140)
        spoof-source(yes)
    );
};

destination d_siem_udp_spoof_lb {
    network(
        "10.200.200.10"
        transport("udp")
        port(5140)
        spoof-source(yes)
    );
};

destination d_siem_file { file("/opt/logs/siem-syslogng.log"); };

#log { source(s_udp); destination(d_siem_file); };
#log { source(s_udp); destination(d_siem_udp); };
log { source(s_udp); destination(d_siem_udp_spoof); };
#log { source(s_udp); destination(d_siem_udp_remote); };
#log { source(s_udp); destination(d_siem_udp_spoof_remote); };
#log { source(s_udp); destination(d_siem_udp_spoof_lb); };
```

Konfigurace NXlog pro testování UDP spoof

```
User nxlog
Group nxlog
Panic Soft

# default values:
# PidFile   /opt/nxlog/var/run/nxlog/nxlog.pid
# CacheDir  /opt/nxlog/var/spool/nxlog
# ModuleDir /opt/nxlog/lib/nxlog/modules
# SpoolDir /opt/nxlog/var/spool/nxlog

define CERTDIR /opt/nxlog/var/lib/nxlog/cert
define CONFDIR /opt/nxlog/var/lib/nxlog

# Note that these two lines define constants only; the log file location
# is ultimately set by the ‘LogFile‘ directive (see below). The
# ‘MYLOGFILE‘ define is also used to rotate the log file automatically
# (see the ‘_fileop‘ block).
define LOGDIR /opt/nxlog/var/log/nxlog
define MYLOGFILE %LOGDIR%/nxlog.log

# By default, ‘LogFile %MYLOGFILE%‘ is set in log4ensics.conf. This
# allows the log file location to be modified via NXLog Manager. If you
# are not using NXLog Manager, you can instead set ‘LogFile‘ below and
# disable the ‘include‘ line.
LogFile %MYLOGFILE%
#include %CONFDIR%/log4ensics.conf
```

```
# This block rotates ‘%MYLOGFILE%‘ on a schedule. Note that if ‘LogFile‘
# is changed in log4ensics.conf via NXLog Manager, rotation of the new
# file should also be configured there.
<Extension _fileop>
    Module  xm_fileop

    # Check the size of our log file hourly, rotate if larger than 5MB
    <Schedule>
        Every   1 hour
        <Exec>
            if ( file_exists('%MYLOGFILE%') and
                 (file_size('%MYLOGFILE%') >= 5M) )
            {
                 file_cycle('%MYLOGFILE%', 8);
            }
        </Exec>
    </Schedule>

    # Rotate our log file every week on Sunday at midnight
    <Schedule>
        When    @weekly
        Exec    if file_exists('%MYLOGFILE%') file_cycle('%MYLOGFILE%', 8);
    </Schedule>
</Extension>

<Extension _syslog>
    Module      xm_syslog
</Extension>

<Input syslog>
    Module      im_udp
```

```
    Port        514
    Host        10.200.200.120
</Input>

#<Output syslog_siem>
#    Module  om_udpspoof
#    Host    127.0.0.1
#    Port    5140
#</Output>

#<Output syslog_siem>
#    Module  om_udp
#    Host    127.0.0.1
#    Port    5140
#</Output>

<Output syslog_siem>
    Module  om_udp
    Host    10.200.200.200
    Port    5140
</Output>

#<Output syslog_siem_file>
#    Module  om_file
#    File    "/opt/logs/siem-nxlog.log"
#</Output>

#<Processor buffer>
#    Module      pm_buffer
#    # 100 MB buffer
#    MaxSize     102400
```

```
#    Type        Mem
#    # warn at 80MB
#    WarnLimit   80000
#</Processor>

#<Route syslog_to_files>
#    Path    syslog  => syslog_siem_file
#</Route>

<Route syslog_to_udp>
    Path    syslog  => syslog_siem
</Route>

#<Route syslog_to_files>
#    Path    syslog => buffer => syslog_siem_file
#</Route>
```

Konfigurace Keepalived s obsluhou IPVS

```
global_defs {
   notification_email {
     root@localhost
   }
   notification_email_from keepalived
   smtp_server 127.0.0.1
   smtp_connect_timeout 30
   router_id SIEMLB
   vrrp_skip_check_adv_addr
   vrrp_garp_interval 0
   vrrp_gna_interval 0
}
```

```
vrrp_instance ARCLB {
    state MASTER
    interface eth1
    virtual_router_id 11
    priority 200
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass tajneheslo
    }
    virtual_ipaddress {
        10.200.200.10
    }
}

virtual_server 10.200.200.10 5140 {
    delay_loop 15
    alpha
    retry 2
    delay_before_retry 5
    lb_algo sh
    lb_kind NAT
    sh-fallback
    persistence_timeout 1
    protocol UDP

    real_server 10.200.200.130 5140 {
        weight 1
        TCP_CHECK {                    # TCP healthchecker
            connect_port 8443     # TCP port to connect
```

```
            connect_timeout 4    # Timeout connection
        }
    }
    real_server 10.200.200.150 5140 {
        weight 1
        TCP_CHECK {                     # TCP healthchecker
            connect_port 8443     # TCP port to connect
            connect_timeout 4    # Timeout connection
        }
    }
}
```

Seznam tabulek

Seznam obrázků

Seznam zkratek

API Application Programming Interface

ArcMC ArcSight Management Console

BSD Berkeley Software Distribution

CEB Connectors in Event Broker

CEF Common Event Format

CERT Computer Emergency Response Team

CRE Custom Rules Engine

CSIRT Computer Security Incident Response Team

CSV Comma Separated Values

DDoS Distributed Denial of Service

DNAT Destination Network Address Translation

DoS Denial of Service

DSM Device Support Module

ESM Enterprise Security Manager

GNULinux GNUś Not Unix

HTTP Hyper-Text Transfer Protocol

IDS Intrusion Detection System

IETF Internet Engineering Task Force

IOPS Input Output Operations per Second

IPS Intrusion Prevention System

JCIFS Java Common Internet File System

JSON JavaScript Object Notation

LEEF Log Event Extended Format

MDI Machine Data Intelligence

OS Operating System

REST Representational State Transfer

RFC Request for Comments

SEM Security Event Management

SIEM Security Information and Event Management

SIM Security Information Management

SMB Server Message Block

SOC Security Operation Center

TCP Transmission Control Protocol

TLS Transport Layer Security

UDP User Datagram Protocol

WEC Windows Event Collector

WEF Windows Event Forwarding

WELF WebTrends Enhanced Log file Format

WiNC Windows Native Connector

WUC Windows Unified Connector

XML Extensible Markup Language