



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

METODIKA NASTAVENÍ PRAVIDEL FIREWALLU

METHODOLOGY FOR SETTING FIREWALL RULES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Michal Vagner

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Petr Sedlák

BRNO 2024

Zadání diplomové práce

Ústav: Ústav informatiky
Student: **Bc. Michal Vagner**
Vedoucí práce: **Ing. Petr Sedlák**
Akademický rok: 2023/24
Studijní program: Informační management

Garant studijního programu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Metodika nastavení pravidel firewallu

Charakteristika problematiky úkolu:

Úvod
Teoretická východiska práce
Analýza současného stavu
Vlastní návrhy řešení
Závěr

Cíle, kterých má být dosaženo:

Cílem této diplomové práce je migrace firewallových pravidel na nový firewall ve firmě.

Základní literární prameny:

ČSN EN ISO/IEC 27001 Informační bezpečnost, kybernetická bezpečnost a ochrana soukromí - Systémy managementu informační bezpečnosti - Požadavky, 2023. [Praha]: Česká agentura pro standardizaci.

ČSN EN ISO/IEC 27002 Informační bezpečnost, kybernetická bezpečnost a ochrana soukromí - Opatření informační bezpečnosti, 2023. [Praha]: Česká agentura pro standardizaci.

DOUCEK, Petr; KONEČNÝ, Martin a NOVÁK, Luděk, 2019. Řízení kybernetické bezpečnosti a bezpečnosti informací. Praha: Professional Publishing. ISBN isbn978-80-88260-39-4.

SEDLÁK, Petr a KONEČNÝ, Martin, 2023. Přeměna ISMS v manažerské informatice. Brno: CERM, akademické nakladatelství. ISBN isbn978-80-7623-110-8.

SEDLÁK, Petr a KONEČNÝ, Martin, 2021. Kybernetická (ne)bezpečnost: problematika bezpečnosti v kyberprostoru. Brno: CERM, akademické nakladatelství. ISBN isbn978-80-7623-068-2.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2023/24

V Brně dne 4.2.2024

L. S.

doc. Ing. Miloš Koch, CSc.
garant

doc. Ing. Vojtěch Bartoš, Ph.D.
děkan

Abstrakt

Diplomová práce se zaměřuje na vysvětlení problematiky firewallů, a především firewallů nové generace. Seznamuje s jejich konkrétním zástupcem Barracuda CloudGen Firewall, projektovým řízením a technologiemi jako je Python a RESTful API. Na základě těchto poznatků staví návrh implementace pro migrování a nastavení firewallových politik na Barracuda CloudGen Firewall ve společnosti COMGUARD a.s.

Abstract

This diploma thesis focuses on explaining the concept of firewalls, and especially next-generation firewalls. It introduces their specific representative Barracuda CloudGen Firewall, project management and technologies such as Python and RESTful API. Based on these findings it creates an implementation proposal for migrating and creating firewall policies on Barracuda CloudGen Firewall in COMGUARD a.s.

Klíčová slova

firewall, objekt, pravidlo, migrace, riziko, rest api, python, next generation, Barracuda, TCO, projektový management

Key words

firewall, object, rule, migration, risk, rest api, python, next generation, Barracuda, TCO, project management

Bibliografická citace

Citace tištěné práce:

VAGNER, Michal. Metodika nastavení pravidel firewallu. Brno, 2024. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/158753>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Sedlák.

Citace elektronického zdroje:

VAGNER, Michal. Metodika nastavení pravidel firewallu [online]. Brno, 2024 [cit. 2024-05-09]. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/158753>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Sedlák.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 13. května 2024

.....

podpis autora

Poděkování

Chtěl bych tímto poděkovat vedoucímu své diplomové práce panu Ing. Petru Sedlákoví za odbornou pomoc, ochotu a trpělivost během zpracování této práce. Dále bych chtěl poděkovat svým přátelům a rodině za podporu během těchto několika měsíců. V neposlední řadě bych chtěl poděkovat svým kolegům a společnosti COMGUARD a.s. za poskytnutí potřebných podkladů, podporu, odbornou konzultaci a celou tuto příležitost.

OBSAH

ÚVOD.....	11
CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ	13
1 TEORETICKÁ VÝCHODISKA PRÁCE.....	14
1.1 Firewall	14
1.2 Typy firewallů	16
1.1.1 Funkcionalita a způsob filtrování dat	16
1.1.2 Způsob nasazení	18
1.2 Next-generation firewall.....	20
1.2.1 Bezpečnostní funkce NGFW	21
1.3 Barracuda CloudGen Firewall	22
1.4 Navigace v Barracuda CloudGen Firewall	23
1.4.1 Konfigurační karta	25
1.4.2 Ovládací prvky konfiguračních stránek	26
1.5 Forwarding Rules	26
1.5.1 Access Rules	26
1.5.2 Firewallové objekty	28
1.6 Analýza rizik	33
1.6.1 RIPRAN.....	34
1.7 Časová analýza.....	41
1.7.1 CPM	41
1.7.2 Síťové grafy.....	43
1.8 RACI matice.....	45
1.9 Total Cost of Ownership	46
1.10 REST API	48
1.10.1 Stavové kódy	51
1.10.2 Barracuda Firewall REST API	52
1.11 Python.....	53
1.11.1 Vestavěné prvky.....	54
1.11.2 Moduly a knihovny.....	55
2 ANALÝZA SOUČASNÉHO STAVU.....	57
2.1 Základní charakteristika společnosti.....	57
2.1.1 Organizační struktura	59
2.2 Důvod migrace.....	60

2.2.1	Struktura uvnitř Forcepoint Sidewinder	62
2.2.2	Zadání projektu migrace	65
3	VLASTNÍ NÁVRHY ŘEŠENÍ.....	66
3.1	Rizika spojená s migrací	66
3.1.1	Identifikace rizik	66
3.1.2	Kvantifikace rizik	67
3.1.3	Návrhy opatření na snížení rizik	69
3.1.4	Celkové posouzení rizik	71
3.2	Popis vyskytnutého rizika a využití nápravného opatření	72
3.2.1	Obnovení primárního firewallu	73
3.2.2	Obnovení sekundárního firewallu	78
3.3	Časová analýza	79
3.3.1	Identifikace činností	79
3.3.2	RACI matice	80
3.3.3	CPM	81
3.3.4	Síťový graf	82
3.4	Migrace objektů	84
3.4.1	Ruční migrace objektů	84
3.4.2	Migrace objektů pomocí skriptu	88
3.4.3	Zhodnocení migrace objektů	107
3.5	Migrace Access Rules	107
3.5.1	Postup přidání nového pravidla	108
3.5.2	Závěr k vkládání pravidel	122
3.6	Finanční zhodnocení	123
3.6.1	Pořizovací náklady	123
3.6.2	Provozní náklady	125
3.6.3	Náklady na obnovu	127
3.6.4	Celkové náklady na vlastnictví	128
3.6.5	Porovnání nákladů	129
	ZÁVĚR	132
	SEZNAM POUŽITÝCH ZDROJŮ	134
	SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ	141
	SEZNAM POUŽITÝCH OBRÁZKŮ	145
	SEZNAM POUŽITÝCH TABULEK.....	150

SEZNAM POUŽITÝCH GRAFŮ	151
SEZNAM PŘÍLOH.....	152
PŘÍLOHY	153

ÚVOD

V dnešním digitálně propojeném světě je ochrana informací a systémů před neoprávněným přístupem, útoky a kybernetickou kriminalitou, která roste na velikosti a sofistikovanosti každým dnem, zásadní. Firewally představují první obrannou linii v boji proti těmto hrozbám tím, že monitorují a regulují provoz mezi různými sítěmi a zařízeními. Firewally nové generace navíc mohou sloužit k více účelům, než je jenom monitorování provozu, proto jsou u společností stále oblíbenější.

Ovšem ne vše musí zůstat neměnné, protože ve společnosti může dojít ke změně v její bezpečnostní strategii, která nemusí pokaždé přijít plánovaně, avšak jakákoliv společnost by měla být schopna ji bezpečně zvládnout, tak aby zamezila možným ztrátám. To vyžaduje, aby manažer kybernetické bezpečnosti rychle reagoval na tyto podněty a připravil s předstihem kvalitní plán, jak se s touto změnou vypořádat.

Tato práce se zabývá jedním z takovýchto možných scénářů, kdy ve společnosti došlo k nutné změně vlivem vnějších faktorů a je potřeba migrovat firewallové politiky na nový firewall zavčas, bezpečně a bezproblémově. Je proto potřeba pochopit problematiku především firewallů nové generace a využít dostupné technologie a znalosti projektového managementu, aby bylo dosaženo požadovaného stavu.

Teoretická část práce se zaměřuje na vysvětlení podstaty firewallů, jejich dělení a dále se soustředí na firewally nové generace, konkrétně jeden z jejich zástupců Barracuda CloudGen Firewall, na který bude prováděna migrace. Je zde představeno jeho uživatelské rozhraní, konfigurace a funkce potřebné pro plánovaný projekt. Krom toho se práce zabývá důležitými aspekty projektového řízení, díky kterému je možné takovéto změny bezproblémově zvládnout. V neposlední řadě jsou zde uvedeny technologie usnadňující prováděnou migraci a předcházející chybám, jako je Python a RESTful API.

Druhou částí práce je analýza současného stavu, kde bude představena společnost COMGUARD a.s., ve které je potřeba provést tuto změnu, její předmět podnikání, nabízené produkty a služby a její organizační struktura. Dále zde bude více přiblížena struktura pravidel a objektů stávajícího firewallu Forcepoint Sidewinder a příčina jeho nutné výměny a migrace stávajících politik a podpůrných konfigurací na Barracuda

CloudGen Firewall. Na základě těchto informací bude vytvořeno zadání migrace, která bude řešena v návrhové části práce.

Konkrétní návrh řešení bude uveden ve třetí části práce, kde za pomoci jazyku Python bude vytvořen skript v kombinaci s technologií RESTful API ke komunikaci s firewallem, aby bylo možné rychle a bezchybně migrovat objekty, které usnadňují práci při následné migraci samotných pravidel. Před celým projektem budou vyhodnocena rizika, vytvořena časová analýza a přiděleny odpovědnosti za jednotlivé aktivity. Celá migrace bude navíc zhodnocena z finančního hlediska a míry úspěchu migrace.

CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Cílem této závěrečné práce je seznámení čtenáře s problematikou firewallů nové generace a obeznámit je s tím, jaké postupy mohou zvolit, jestliže se vyskytnou v situaci, kdy je potřeba provést migraci firewallových politik na nové zařízení, které vyžaduje jiný syntax. Tato problematika nese značná rizika a může být také velmi časově náročná, proto se tato práce zaměřuje na popis metod, které zajistí snížení faktorů, které by mohly mít negativní dopad. Mezi tyto postupy patří analýza rizik, časová analýza a také využití technologií jako je Python nebo RESTful API. V neposlední řadě mohou být náklady spojené s vlastnictvím konkrétního produktu často podceňovány. Z tohoto důvodu je na závěr práce provedena podrobná analýza TCO.

Práce se skládá ze tří částí. V části Teoretická východiska práce jsou zmíněné metody představeny, tak aby je mohl pochopit i relativní začátečník. Následně pro lepší pochopení kontextu situace, ve které je tato problematika uplatňována, je vytvořena část Analýza současného stavu. V této části je především představena společnost COMGUARD a.s. a důvod, proč musí migrovat na nový firewall. Teoretické poznatky a výstup analýzy slouží k praktickému využití v části práce Vlastní návrhy řešení.

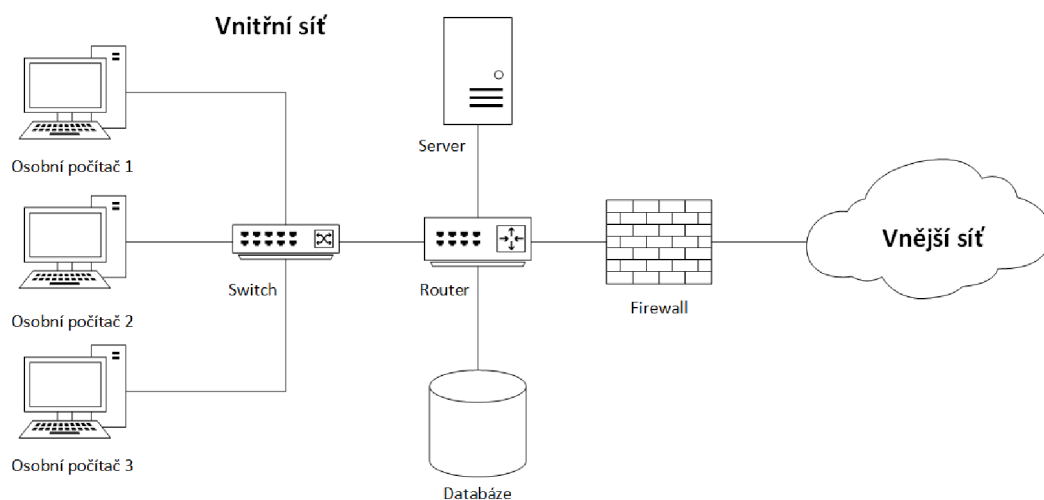
Dílčím cílem práce je také umožnění replikace stejného nebo obdobného postupu čtenářem, který se nachází v podobné situaci. Je proto nutné dbát na stanovení jasných a snadno interpretovatelných postupů. Využité metody by navíc měly být zvoleny na základě přínosů pro celou práci a dané téma.

1 TEORETICKÁ VÝCHODISKA PRÁCE

1.1 Firewall

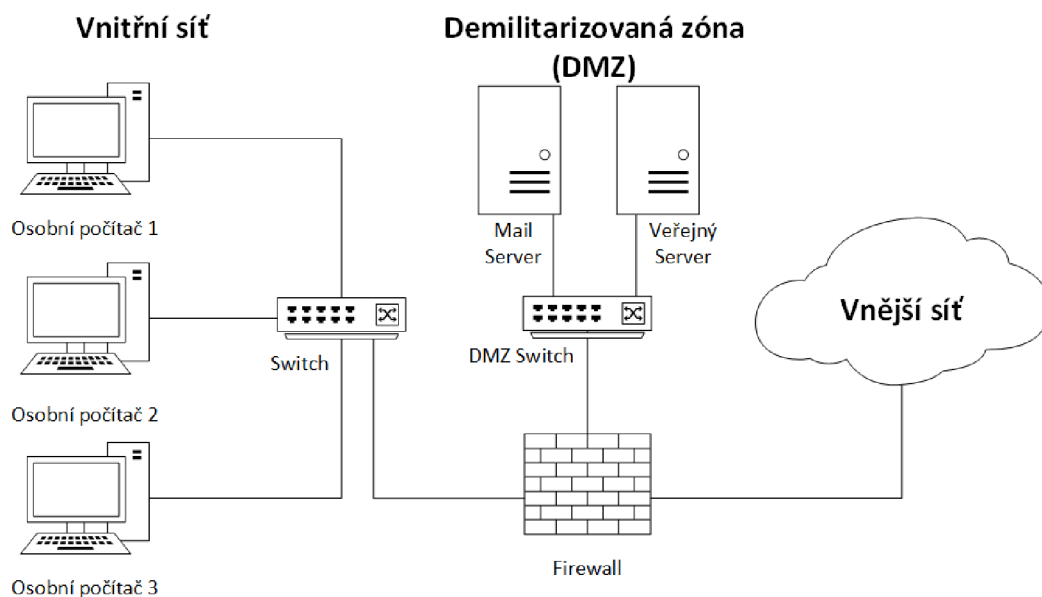
Abychom pochopili, co je firewall, můžeme použít analogii ze středověku, když ještě vládli králové a lordi a stavěly se hrady. Takovýto hrad představoval sídlo onoho vládce nebo alespoň strategicky významné místo, které nesmělo být ztraceno nebo infiltrováno. Z tohoto důvodu zde byly stavěny zdi, které podobně jako firewall bránily ve vstupu nepovolaným osobám a odrážely případné útoky. Avšak osoby s povoleným vstupem byly vpuštěny hlavní bránou po kontrole strážným, stejně jako když provoz splňuje firewallové pravidlo, je vpuštěn dovnitř do sítě. (1; 2)

Firewall (FW) je tedy mechanismus používaný k řízení a monitorování provozu do a ze sítě za účelem ochrany zařízení v síti. Porovnává příchozí a odchozí provoz s předdefinovanými bezpečnostními pravidly a provoz nebo zpráva, které nesplní kritéria zahodí. Firewall tak může blokovat pokusy o neoprávněný přístup z vnějších sítí a zabráňuje šíření malwaru. Je zásadní součástí podnikové a osobní bezpečnostní strategie, protože poskytuje první obrannou linii v ochraně dat a infrastruktury. (1; 3)



Obrázek 1: Zjednodušený diagram firewallu chránící vnitřní síť (Zdroj: Vlastní zpracování dle: (1))

Firewall dále slouží také jako nástroj pro správu a kontrolu interního provozu v rámci sítě. Některé nástroje pro tuto správu zahrnují segmentaci sítě, tvorbu demilitarizovaných zón, kontrolu přístupů, monitoring a logování. Jde tedy rychle pochopit, proč se jedná o jeden z nejdůležitějších bezpečnostních prvků sítě. (4; 5)



Obrázek 2: Zjednodušený diagram vytvoření DMZ pomocí firewallu (Zdroj: Vlastní zpracování dle: (1))

Nastavení firewallových pravidel a firewally obecně míří k zachování CIA triády napříč všemi aktivy organizace, které jsou připojeny k síti. Zde jsou základní principy CIA:

- **Důvěrnost (Confidentiality):** Jedná se o zabránění neoprávněnému zveřejnění citlivých informací, zabránění přístupu ke zdrojům neautorizovaným uživatelům a zajištění toho, aby byla ve všech fázích zpracování údajů vynucována odpovídající úroveň utajení. Šifrování a tvorba seznamů pro řízení přístupů jsou klasickým příkladem technologie zaměřené na poskytování důvěrnosti.
- **Integrita (Integrity):** Integrita se zaměřuje na prevenci neoprávněné úpravy dat a zajištění jejich přesnosti, což mohou plnit funkce firewallů nové generace. Firewally nové generace jsou například schopny provádět hloubkovou kontrolu paketů (*DPI*), aby analyzovaly jejich obsah a zajistily, zda neobsahují malware nebo jiný škodlivý obsah, který by mohl pozměnit nebo poškodit data.
- **Dostupnost (Availability):** Dostupnost zajišťuje, že data a zdroje jsou v případě potřeby přístupné oprávněným uživatelům. K tomuto firewally přispívají tím, že chrání síť před útoky, jako jsou například útoky *DDoS* (*Distributed Denial of Service*), které mohou zahltnout síť provozem a učinit ji nedostupnou pro legitimní uživatele. V některých případech se mohou firewally také podílet na řízení kvality provozu a optimalizovat tak chod sítě. (3; 6)



Obrázek 3: CIA triáda (Zdroj: Vlastní zpracování dle: (6))

1.2 Typy firewallů

Firewallly lze dělit dle několika různých kategorií. Mezi ty nejdůležitější patří funkcionality a způsob filtrování dat a způsob nasazení. (1; 5)

1.1.1 Funkcionality a způsob filtrování dat

Klasifikace založená na funkčnosti se zabývá tím, co firewall dělá a jakým způsobem zpracovává, filtruje a reguluje síťový provoz. Pochopení této klasifikace umožňuje organizaci vybrat vhodné řešení pro své potřeby. (1; 7)

Mezi základní typy patří packet-filtering firewall, stateful firewall, proxy firewall a next-generation firewall. Jejich hlavní rozdíly jsou v nabízených funkcích, náročnosti na výkon a správu a úrovni zabezpečení. (1; 7)

Packet-Filtering firewall

Firewallly pro filtrování paketů jsou nejjednodušším a nejzákladnějším typem firewallů. Pracují na třetí vrstvě ISO/OSI modelu a rozhodují se na základě hlaviček paketů. Zaměřují se na kontrolu zdrojové a cílové IP adresy každého paketu, čísel portů

a používané protokoly (TCP, UDP, ICMP atd.). Zda firewall nechá provoz projít nebo ne je rozhodnuto na základě předdefinovaných pravidel, která obsahují již zmíněné parametry. (1; 5)

Jejich hlavní výhodou je jejich minimální dopad na výkon systému, což jim umožňuje zvládat velké objemy provozu s relativně nízkou latencí. Avšak v porovnání s pokročilými typy firewallů, které umožňují i kontrolu stavů, jsou méně zabezpečené. Nedokážou totiž zjistit, zda je paket součástí existující autorizované relace nebo nového, potenciálně škodlivého připojení. Jsou tedy náchylnější k určitým typům útoků, jako je IP spoofing a TCP session hijacking. (1; 3)

Stateful firewall

Stateful firewall (stavový firewall), také známý jako firewall s dynamickým filtrováním paketů, přidává důležitou úroveň zabezpečení udržováním tabulky stavů, která sleduje všechna aktivní připojení. Každý přichodzí nebo odchodzí paket je totiž kontrolován nejen podle definovaných pravidel, ale také podle kontextu jeho toku provozu nebo relace. Tato stavová kontrola umožňuje firewallu rozlišovat mezi legitimními pakety pro různé typy připojení a škodlivými pakety, které se pokoušejí zneužít síť. (1; 3; 5)

Tyto firewally jsou zabezpečenější než jednoduché firewally pro filtrování paketů, protože pracují se stavem síťových připojení (např. zda se TCP spojení právě vytváří, je navazováno nebo je uzavřeno). Tato schopnost pomáhá předcházet útokům, které zneužívají existující připojení, jako je například TCP/IP session hijacking. Nevýhodou je vyšší nárok na výpočetní výkon, což může vést k větší délce doby průchodu paketů a delší odezvě. Zvýšení úrovně zabezpečení, v porovnání s tímto relativně malým dopadem, se v naprosté většině případů vyplatí. (1; 3)

Proxy firewall

Proxy firewally fungují na aplikační vrstvě ISO/OSI modelu, proto se jim říká také *application-level gateways*. Fungují jako prostředník mezi uživatelem a cílovým serverem a účinně izolují vnitřní síť od přímého externího přístupu. Když síťový uživatel požaduje přístup k externí službě, proxy firewall naváže spojení jménem uživatele, načte

požadovaná data a poté je odešle zpět uživateli. Díky tomu mezi interním uživatelem a externími servery nikdy nedojde k přímému spojení. (1; 2)

Jejich silnou stránkou je kontrola obsahu jednotlivých zpráv na aplikační úrovni, než dojde k jejich předání uživateli. Do této kontroly patří podrobné filtrování obsahu, včetně blokování konkrétních typů obsahu, škodlivého obsahu a známého malwaru. Proxy firewall může výrazně navýšit zabezpečení sítě, a proto se velmi hodí do podnikových sítí. Avšak jeho nevýhodou může být vyšší dopad na výkon sítě, a tedy i odezvu a také odezva od uživatelů sítě, protože nemohou přistupovat k určitému typu obsahu. (1; 2)

Next-generation firewall

Next-generation firewalls (NGFW), česky firewally nové generace, představují nutnou evoluci tradičního konceptu firewallu. Spojují funkce tradičního firewallu s moderními a sofistikovanými funkcemi, jako jsou intrusion prevention system (IPS), intrusion detection system (IDS), deep packet inspection (DPI), application control jako je SSL inspection, správa přístupů a identit a mnohé další. Cílem NGFW je poskytnout komplexní bezpečnostní řešení, které dokáže vynutit bezpečnostní zásady na nejnižší úrovni. (1; 4)

NGFW vyžadují ze všech zde zmíněných typů pravděpodobněji nejvyšší výpočetní výkon, ale zato poskytují nejrozsáhlejší bezpečnostní funkce a míru konfigurace. Tato konfigurace však vyžaduje školený personál, protože může být složitá a znalost všech pokročilých funkcí zamezí chybovosti. Nicméně rozvoj technologií a sofistikovanosti hrozeb klade velký důraz na vývoj a posun technologií v oblasti kybernetické bezpečnosti, proto se NGFW budou stávat stále větší normou pro zabezpečení organizace. NGFW budou rozvedeny více v samostatné kapitole. (1; 4)

1.1.2 Způsob nasazení

Způsob nasazení řeší, jak jsou firewally technologicky implementovány a kde jsou nasazeny. Toto dělení opět pomáhá organizaci vybrat řešení, které bude dobře fungovat s její činností, architekturou sítě a dalšími požadavky. (1; 7)

Firewall je možné nasadit v různém provedení či konfiguraci. Existuje možnost klasického hardware zařízení, softwarového řešení, cloudového řešení nebo virtuálního firewallu. Všechny tyto typy nabízí různé výhody, způsob využití, míru přizpůsobení, úroveň ochrany a náročnost údržby. (1; 8)

Hardware zařízení

Samostatné hardwarové nebo hardwarové/softwarové jednotky se často označují jako síťové firewally a obvykle poskytují nejbezpečnější řešení pro soukromé sítě bez vytváření zátěže na jiné zařízení v síti. Tato zařízení jsou navržena pro různé velikosti sítí a poskytují různé funkce na základě konkrétního modelu a zvolené konfigurace. Obvykle jsou používány tam, kde je kladen vysoký důraz na bezpečnost a výkon, protože se jedná o spolehlivé řešení se schopností zpracovat velké objemy datového provozu. (8)

Softwarové řešení

Softwarové firewally jsou nainstalované přímo na pracovní stanici nebo serveru. Mohou být tedy součástí operačního systému nebo nainstalovány později jako doplňkový bezpečnostní software. Hlavním úkolem softwarových firewallů je ochrana individuálních zařízení především na základě monitorování a řízení interakcí aplikací s internetem nebo jinými sítěmi. Jejich hlavní výhodou jsou nízké náklady a jednoduchost aktualizace. (8)

Firewall as a Service

Cloudové firewally jsou poskytovány jako služba, Firewall as a Service (FWaaS), a jsou hostovány a řízeny přímo v cloudovém prostředí. Nabízí značnou škálovatelnost, jednoduchost nasazení a vzdálenou správu. Jejich využití je výhodné především pro společnosti, které již ve vysoké míře využívají cloudové služby, díky jednoduchosti integrace nebo pro společnosti se značnou mírou mobility svých zaměstnanců při výkonu práce, což umožňuje bezpečný přístup ke zdrojům společnosti. Míra škálovatelnosti navíc nabízí dobrý poměr ceny a výkonu. (9)

Virtuální firewall

Virtuální firewally jsou softwarová zařízení implementované ve virtualizovaných prostředích a poskytují zabezpečení napříč virtuálními stroji a sítěmi. Jsou kritické v prostředích, kde se intenzivně využívá virtualizace, jako jsou datová centra a cloudové platformy třeba pro segmentaci provozu. (10)

1.2 Next-generation firewall

Obrana proti kybernetickým útokům je každým dnem náročnější a už to nebude snazší. Oblast hrozeb se vyvíjí směrem k rychlejší, účinnější a efektivnější kriminální ekonomice těžící z útoků proti uživatelům, podnikům, poskytovatelům služeb a vládám. Organizovaný kybernetický zločin a výměna exploitů zažívají boom a podporují velmi lukrativní ekonomiku. Útočníci dnes jasně chápou základní bezpečnostní technologie a jejich zranitelnosti, navíc skupiny hackerů nyní začali pracovat jako podniky a sledují životní cykly vývoje svého softwaru, provádějí testování kvality na bezpečnostních produktech, než je uvolní do světa a neustále hledají způsoby, jak uniknout bezpečnostní obraně. (2; 4)

Tradiční firewally se soustředily na poskytování vysoké úrovně viditelnosti toho, co se děje v síti, a zakazování provozu v místě vstupu při neshodě s pravidly. Útočníci již však neprovádějí pokročilé útoky v jednom okamžiku, jejich útočná schémata a kampaně využívají sofistikované metodologie, jako je šifrovaný provoz, zero-day útoky, vyhýbání se detekci, command and control (C&C) a různé jiné maskovací techniky. Naučili se také tyto vektory útoku kombinovat do sofistikovanějších útoků, složitějších na odhalení a provádějící více než jeden úkol. (4; 11)

Next-generation firewally představují novou vysokou úroveň síťového zabezpečení díky budování ochranných vrstev přes které se musí případný útočník dostat. Tyto vrstvy jsou tvořeny moderními funkcemi vestavěnými nad klasickou kontrolou portů a protokolů. Útočník tak musí přijít s mnohem sofistikovanějším útokem, který musí překonat ne jeden, dva, ale třeba i několik bezpečnostních mechanismů vytvořených pro optimalizaci zabezpečení sítě. Této formě ochrany se říká cibulový model a je většinou tvořen několika

nezávislými opatřeními, NGFW tuto ochranu ale nabízí uvnitř jednoho řešení pomocí již zmíněných IPS, IDS, DPI, SSL inspekce a mnohých dalších funkcí. (4; 12)

NGFW tak řeší řadu problémů spojených s novodobými hrozbami a útoky tím, že vynucuje bezpečnostní politiky na úrovni aplikací, a nejen na úrovni portů, protokolů nebo IP adres, což zajišťuje přístup pouze bezpečným a autorizovaným aplikacím. Nabízí také detailní kontrolu nad sítí na základě uživatelských identit a skupinových zásad, což může usnadnit správu a zavést větší kontrolu v podnikovém prostředí. Součástí ochrany bývají také často aktualizované databáze hrozeb na základě, kterých detekční algoritmy monitorují provoz a mohou v některých případech zabránit bezpečnostním incidentům. V neposlední řadě přináší SSL inspekce nepostradatelnou kontrolu nad šifrovaným provozem jeho dešifrováním a inspekci. (2; 4)

1.2.1 Bezpečnostní funkce NGFW

IPS – Intrusion Prevention System (Systém prevence narušení) - IPS je technologie zabezpečení sítě, která zkoumá toky síťového provozu, aby zjistila a zabránila zneužití zranitelnosti. Systémy IPS mohou provádět detekci anomálií a mohou blokovat hrozby, které se mohou vyhnout jednoduché detekci založené na signaturách. (13)

ATP – Advanced Threat Protection (Pokročilá ochrana proti hrozbám) – ATP zahrnuje kombinaci nástrojů a technik používaných k ochraně před sofistikovanými, často utajenými pokusy o narušení sítě, například pomocí sandboxingu, kdy jsou soubory nebo programy spouštěny ve virtuálním prostředí za účelem kontroly potenciálních hrozeb, analýzy chování k identifikaci neobvyklé aktivity v síti a databáze známých hrozeb. ATP pomáhá při identifikaci a reakci na pokročilé přetrvávající hrozby (APT), které tradiční bezpečnostní opatření mohou minout. (14)

URL Filtering (Filtrování URL) – Jedná se o funkci, která uživatelům blokuje přístup ke konkrétním webovým stránkám za účelem vynucování bezpečnostních zásad, ale i podnikových politik a udržování produktivity. Blokové mohou být například známé webové stránky obsahující malware, phishingové stránky, stránky s nevhodným obsahem pro pracovní prostředí a další. Jedná se o klíčový nástroj, který zabraňuje bezpečnostním incidentům zaviněných lidskou chybou. (15)

Antivirová ochrana – Antiviry uvnitř NGFW kontrolují příchozí soubory a přenosy a zjišťují, jestli se jedná o známé viry nebo signatury malwaru. Tato funkce je zásadní pro zachycení malwaru před tím, než vstoupí do sítě. (16)

Traffic shaping (Řízení provozu) – Pro zajištění optimálního chodu a kvality spojení některých aplikací nebo provozu mohou NGFW vynucovat tzv. Traffic shaping neboli Quality of Service (Kvalitu služeb), což je správa šířek pásma, síťových spojení a dalších fungujících na základě nastavených priorit. Tato funkce je obzvlášť důležitá v prostředích, kde jsou síťové zdroje omezené a je třeba je alokovat dle priorit společnosti. (17)

User Awareness (Povědomí o uživateli) – Jedná se o funkci v NGFW umožňující firewallu identifikovat a řídit přístup uživatele nebo skupiny uživatelů. Je integrována s adresářovými službami (jako je MS Active Directory) a aplikuje bezpečnostní zásady na základě identity uživatele, nikoli pouze podle IP adresy, což se může hodit při prosazování zásad, které se liší podle role uživatele v rámci organizace. (18)

SSL inspection (Inspekce SSL) – metoda dešifrování šifrovaného provozu, kontrole dešifrovaného obsahu na skryté hrozby a následnému opětovnému zašifrování provozu před jeho odesláním na místo určení. Krom toho může firewall blokovat také zastaralé formy šifrování, které nejsou tak bezpečné. (5; 19)

Mail security (Mailové zabezpečení) – Spadá sem filtrování spamu, antivirové skenování příloh emailů a ochrana proti phishingu. Cílem je opět předejít lidské chybě. (20)

1.3 Barracuda CloudGen Firewall

Jedno z předních řešení firewallů nové generace na trhu je Barracuda CloudGen Firewall (CGFW) nabízející pokročilou ochranu a centrální správu celé sítě ve formě hardware, cloud ale i virtuálních zařízení. Barracuda nabízí řadu modelů, kde každý model je přizpůsoben tak, aby splňoval specifické potřeby síťové infrastruktury, což zajišťuje, že organizace všech velikostí a s různými technologiemi jej mohou využívat. Barracuda CloudGen Firewall také poskytuje plnou a snadnou integraci s cloudovými službami, jako jsou AWS, Azure a Google Cloud Platform, což je obzvlášť přitažlivé pro podniky,

které dávají přednost cloudovým službám, nebo pro ty, které právě přecházejí do cloudu. (21; 22)

Mezi jeho přední vlastnosti patří bezpečné připojení na virtuální síť pomocí technologie VPN s vlastním proprietárním protokolem TINA založeným na protokolu IPsec. Navíc má CGFW podporu chytré správy sítě, což znamená, že krom samotné ochrany sítě zajišťuje CGFW také nepřetržitost síťového provozu díky efektivní správě, která chrání síť před výpadky a zhoršením kvality spojení. Mimo to disponuje všemi pokročilými bezpečnostními funkcemi firewallů nové generace, včetně ATP, IPS, filtrování URL, antiviru a SSL inspekce. Jedná se tedy o ideální řešení do dynamických a náročných podnikových sítí s potřebou vysoké kvality zabezpečení. (23)

Správa nastavení politik, klientů a zařízení je prováděna centrálně pomocí Barracuda Firewall Admin nebo Barracuda Firewall Control Center, dále už pouze Firewall CC, kde je také možné sledovat aktuální síťový provoz a provádět tak aktuální diagnostiku sítě. Centralizovaná správa prostřednictvím Firewall CC tak zjednodušuje správu zabezpečení sítě, zejména pro organizace s více pobočkami nebo rozsáhlou cloudovou infrastrukturou, ve kterých mohou správci dohlížet, konfigurovat a spravovat všechny firewally z jediného jednotného rozhraní, které usnadňuje úlohy, jako jsou aktualizace, opravy a vynucování zásad. (22)

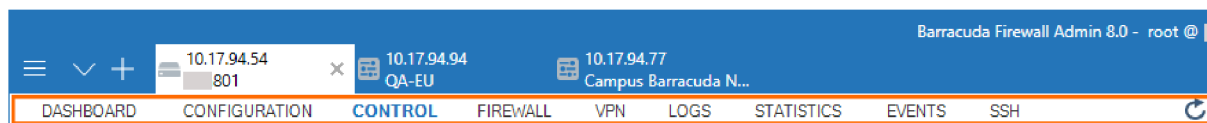


Obrázek 4: Logo společnosti Barracuda (Zdroj: (23))

1.4 Navigace v Barracuda CloudGen Firewall

Servisní lišta (*Service tab*) je hlavní navigační a provozní nástroj uživatelského rozhraní a poskytuje kartu pro každou hlavní sekci Firewall Admin nebo Firewall CC. Další služby představené na CloudGen Firewallu, jako je VPN, přidávají na tuto lištu další karty,

odkud je možné přistupovat k nastavení a podsekcím v závislosti na nakonfigurované službě. (24)



Obrázek 5: Servisní lišta v Barracuda CGFW (Zdroj: (24))

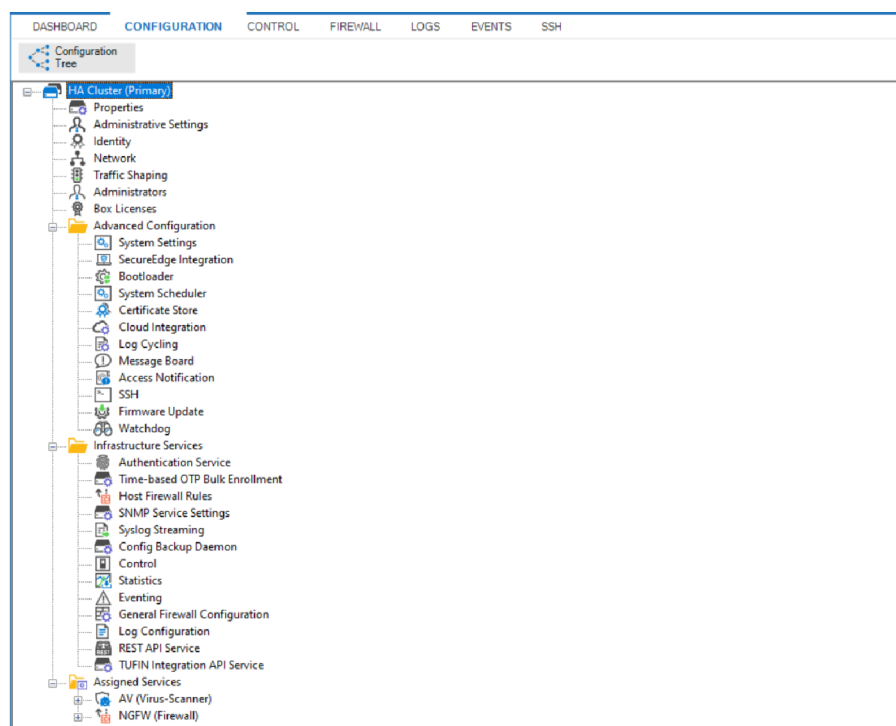
Servisní lišta v rozhraní Firewall Admin obsahuje následující karty:

- **DASHBOARD** – Poskytuje obecný systémový přehled ohledně CloudGen Firewallu nebo Firewall CC. Tato stránka se otevře po prvním přihlášení do Barracuda CloudGen Firewallu.
 - **CONFIGURATION** – Obsahuje konfigurační strom pro CloudGen Firewall nebo Firewall CC. Z tohoto stromu probíhá většina konfiguračních nastavení. Veškeré prvky jsou uspořádány hierarchicky.
 - **CONTROL** – Zobrazuje informace o službách, aktuálním stavu sítě, běžících procesech, stavu systému a licencích.
 - **FIREWALL** – Poskytuje informace v reálném čase, a i historické informace o síťovém provozu a provozu aplikací procházejících CloudGen Firewalllem.
 - **DHCP** – Zobrazuje přidělené a rezervované IP adresy a rozsahy. Umožňuje také jejich správu.
 - **VPN** – Poskytuje přístup k přehledu VPN site-to-site a client-to-site spojení v reálném čase, jestliže je nakonfigurováno.
 - **LOGS** – Záložka LOGS zobrazuje systémové události procesů na vrstvě boxu a pro každou nakonfigurovanou službu.
 - **STATISTICS** – Poskytuje informace o všech připojeních vytvořených z firewallu.
 - **EVENTS** – Obsahuje informace související s událostmi, které jsou vytvořeny na Barracuda CloudGen Firewall.
 - **SSH** – Poskytuje přihlášení k rozhraní příkazového řádku CloudGen Firewallu.
- (24)

1.4.1 Konfigurační karta

Konfigurační změny uvnitř Firewall Admin jsou prováděny v kartě *CONFIGURATION* odkud je možné se navigovat na jednotlivé konfigurační stránky. Nachází se zde konfigurační strom, který obsahuje veškeré konfigurační prvky firewallu v hierarchickém pořadí, poklepáním na konfigurační uzel se otevře rozpad všech jeho stránek. Každá konfigurační stránka může být zobrazena v základním nebo pokročilém zobrazení. Základní zobrazení obsahuje nejčastěji používané konfigurace, zatímco rozšířené zobrazení zobrazuje všechna dostupná konfigurační nastavení. (25)

Na obrázku níže je možné vidět rozbalený konfigurační strom. Mezi nejdůležitější konfigurace, které budou zároveň použity při migraci jsou *Administrators*, *REST API Service* a především *NGFW (Firewall)*, kde se nachází *Forwarding Rules*.

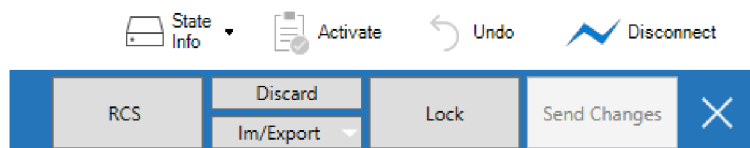


Obrázek 6: Konfigurační strom Barracuda CGFW (Zdroj: Vlastní zpracování)

1.4.2 Ovládací prvky konfiguračních stránek

Po otevření konfigurační stránky z konfiguračního stromu je možné najít možnosti pro úpravu nastavení, tak jako na obrázku níže. Zde jsou ty nejčastěji používané:

- **Uzamknout (Lock)** – Před každou úpravou je potřeba stránku zamknout, jelikož ke konfiguraci může přistupovat více administrátorů a mohlo by dojít ke konfliktům při provádění změn. Zámek uzamkne konfigurační stránku, abyste měli přístup k zápisu, a změni stav tlačítka na Odemknout, které stránku odemkne.
- **Odeslat změny (Send Changes)** – Po jakékoliv úpravě nastavení je nutné změny odeslat na firewallu nebo CC kliknutím na tlačítko Odeslat změny. Po úspěšném odeslání změn se v horní části zobrazí nápis Čeká na aktivaci.
- **Zahodit (Discard)** – Tato možnost umožňuje zahodit změny po jejich odeslání.
- **Aktivovat (Activate)** – Chcete-li po úpravě a odeslání změn aplikovat konfiguraci na firewall, klikněte na tlačítko Aktivovat. (25)



Obrázek 7: Ovládací prvky konfiguračních stránek (Zdroj: (25))

1.5 Forwarding Rules

V této části konfigurace probíhá nastavení základních firewallových politik, které řídí provoz dovnitř a ze sítě. Hlavní prioritou zde budou *Access Rules* a *Firewall Objects*. (26)

1.5.1 Access Rules

Jádro celého firewallu je tvořeno pravidly, které řídí provoz na základě shody. Pravidla se vyhodnocují shora dolů. První pravidlo, které se shoduje bude provedeno, pokud ale komunikace neodpovídá prvnímu pravidlu, vyhodnotí se další pravidlo a pokračuje se

tímto způsobem, dokud není nalezeno odpovídající pravidlo. Pokud žádné z pravidel neodpovídá, spojení se zablokuje. (26)

Action	Name	Features	Policy Usage	NAT Mode	Service	Source	Destination	User
Pass	allow-admins-2-external		None	Explicit NAT	Any	192.168.1.201, 192.168.1.2...	Any	Any
Deny NAT	allow-webservices-from...		All Policy Pr...	N.A.	TCP 8010-8080	0.0.0.0/0	83.22.64.25	Any
App Redirect	RestAPI-Access		N.A.	N.A.	TCP 8080	Trusted LAN	192.168.200.0/24	Any
Pass	BOX-BARRACUDA-CUD...		None	Explicit NAT	ScreenConnect	Ref: Trusted LAN, Ref: WI...	192.168.200.200	Any
Block	BlockATDQuarantine		N.A.	N.A.	Any	ATD	Any	Any
Pass	BOX-EVAL-MODE-BRIDGE		All Policy Pr...	N.A.	Any	Eval Mode Bridged Ports	Eval Mode Bridged Ports	Any
Primary Management IP Ok...	BOX-SERVICEIP-MGMT...		None	Explicit NAT	Ref: HTTPS, Ref: NGFW-MG...	0.0.0.0/0 dev MGMT, 0.0.0...	Service IPs	Any
App Redirect	BOX-SETUP-MGMT-ACC...		N.A.	N.A.	ECHO, TCP 22, TCP 443	Private IP+4 Addresses	DHCP1	Any
App Redirect	BOX-LOCALDNSCACHE		N.A.	N.A.	DNS	Trusted LAN	Local IP	Any
Basic LAN and VPN Rules (6)								
Pass	BOX-LAN-2-INTERNET		All Policy Pr...	N.A.	Any	Ref: Trusted LAN	Ref: Internet	Any
Pass	BOX-LAN-2-LAN		All Policy Pr...	N.A.	Any	Trusted LAN	Trusted LAN	Any
Pass	BOX-LAN-2-VPN-SITE		All Policy Pr...	N.A.	Any	Trusted LAN	VPN-Networks	Any
Pass	BOX-VPN-SITE-2-SITE		All Policy Pr...	N.A.	Any	Ref: VPN-Local-Networks	Ref: VPN-Remote-Networks	Any
Pass	BOX-VPNCLIENTS-2-LAN		All Policy Pr...	N.A.	Any	Any	Trusted LAN	Any
Block	BLOCKALL		N.A.	N.A.	Any	Any	Any	Any

Obrázek 8: Barracuda CGFW – Forwarding Rules – Access Rules (Zdroj: Vlastní zpracování)

Pravidla jsou vyhodnocována na základě nastavených kritérií. Je možné nastavit následující kritéria:

- **Služba (Service)** – protokol a rozsah protokolů/portů odpovídající přenosu. Pro pravidlo je možné definovat jednu a více služeb. Je zde možné použít předdefinovaný objekt (*Service Object*) nebo vytvořit vlastní.
- **Zdroj (Source)** – Zdrojová IP adresa nebo maska sítě, která má být zpracována pravidlem. Může zde být vybrán síťový objekt (*Network Object*), explicitně zadána konkrétní IP adresa nebo maska sítě.
- **Cíl (Destination)** – Cílová IP adresa nebo maska sítě, která má být zpracována pravidlem. Může být opět vybrán síťový objekt nebo explicitně zadaná hodnota.
- **Plán/Čas (Schedule/Time)** – U pravidel je možné dodatečně použít rozvrhový objekt (*Schedule Object*), který udává, kdy bude pravidlo aktivní. Toto kritérium je volitelné.
- **Uživatel (User)** – Dalším volitelným kritériem je uživatel, což omezí pravidla na specifického uživatele nebo skupiny uživatelů. Jsou zde používány uživatelské objekty (*User Objects*). (26)

Jakmile je pravidlo vyhodnoceno na základě stanovených kritérií může nastat jedna z akcí, které se volí při tvorbě pravidla. Mezi tyto akce patří:

- **Povolit (Pass)** – Veškerý provoz odpovídající pravidlu je přeposlán dál.
- **Blokovat (Block)** – Veškerý provoz odpovídající pravidlu je zablokován. Odesílatel není informován, že přenos byl zablokován.
- **Odmítnout (Deny)** – Veškerý provoz odpovídající tomuto pravidlu přístupu bude odmítnut. Po odmítnutí je odeslána zpráva TCP-RST pro požadavky TCP, ICMP Port Unreachable pro požadavky UDP a ICMP Denied by Filter pro ostatní IP protokoly.
- **Destination NAT (Dst NAT)** – Firewall přepíše cílovou IP adresu, síť nebo port na předdefinovanou síťovou adresu (např. přesměrování příchozích paketů s cílovou veřejnou adresou nebo portem na privátní IP adresu nebo port uvnitř sítě).
- **Mapa (Map)** – Firewall přepíše příchozí rozsahy nebo IP adresy na cílové síť nebo IP rozsahy, podobně jako pravidlo Dst NAT dělá pro jednu IP adresu.
- **Přesměrování aplikace (App Redirect)** – Firewall lokálně přesměruje provoz na jednu ze služeb běžících na CloudGen Firewall.
- **Broad Multicast** – broadcast vyhovující tomuto pravidlu je propagován dál. Používá se pro síť s mostem (*bridging*).
- **Kaskáda (Cascade)** – Provoz je přesměrován na podmnožinu pravidel.
- **Kaskáda zpět (Cascade Back)** – Součástí podmnožin pravidel, navrátí se na seznam globálních pravidel a obnoví vyhodnocování pravidel pod pravidlem kaskády. (26)

1.5.2 Firewallové objekty

Správa a udržování firewallů může být náročná, obzvláště ve velkých podnikových sítích, kde můžou existovat stovky nebo i tisíce pravidel. Když dojde ke změně v síti, je potřeba tato pravidla změnit a ruční měnění každého pravidla vede k častým a nežádoucím

chybám, které mohou mít různé následky, od nemožnosti přístupu k nějaké službě až po otevření možnosti napadení sítě nějakým útočníkem. Z tohoto důvodu je nejen v Barracuda CloudGen Firewallu možné definovat firewallové objekty (*Firewall Objects*), které předchází těmto chybám a ulehčují správu. (27)

Objekt funguje jako zástupce, který je použit v místě, kde by mělo být vyplněno třeba nějaké kritérium jako je IP adresa, uživatel apod. Tímto způsobem může být použit neomezeně a když je poté potřeba udělat změnu, stačí změnit daný objekt a ten se sám upraví u všech pravidel, což eliminuje výše zmíněné problémy. Další výhodou je snazší zapamatování si názvu vytvořeného objektu než třeba celé IP adresy. V Barracuda CloudGen Firewallu se může setkat s následujícími objekty:

- **Objekty připojení (Connection Objects)** – Objekty připojení definují výstupní rozhraní a zdrojovou (NAT) IP adresu.
- **Proxy ARP** – Překládá MAC adresy, které nejsou fyzicky na CloudGen Firewallu, na odpovídající IP adresy.
- **Sít'ové objekty (Network Objects)** – Odkazuje na sítě, IPv4 a IPv6 adresy, názvy hostitelů, geolokační objekty, adresy MAC nebo rozhraní.
- **Pojmenované sítě (Named Networks)** – Přenáší informace o podsítích a rezervované IP adresy do konfigurace firewallu v podobě čitelné pro člověka. Pojmenované sítě lze použít jak pro vyhodnocení sady pravidel, tak pro vizualizaci.
- **Objekt služby (Service Objects)** – Definuje, které cílové a klientské TCP/UDP porty a/nebo IP protokoly může služba aplikovaná na pravidlo používat.
- **Uživatelské objekty (User Objects)** – Uživatelské objekty se odkazují na konkrétní uživatele a skupiny uživatelů.
- **Rozvrhové objekty (Schedule Objects)** – Pro omezení pravidel na konkrétní časy a intervaly.
- **Skupiny rozhraní (Interface Groups)** – Skupina rozhraní určuje rozhraní, které může zdrojová adresa používat.

- **Aplikace (Applications)** – Kombinace předdefinovaných aplikací a vlastních aplikací.
- **URL Filter** – Omezení přístupu na určité webové stránky. CloudGen Firewall poskytuje předdefinovaný seznam URL kategorií, které jsou k dispozici pro blokování a povolování.
- **Zásady obsahu souborů (File Content Policies)** – Filtruje stažené soubory nebo přílohy e-mailů podle typu souboru, názvu nebo typu MIME.
- **Zásady uživatelského agenta (User Agent Policies)** – Filtrujte webový provoz na základě informací obsažených v řetězci uživatelského agenta. (27)

Network Objects

Network Objects neboli síťové objekty jsou používány při tvorbě pravidel, když je potřeba se odkázat na konkrétní *síť*, *IP rozsah*, *IPv4 nebo IPv6 adresu*, *hostname*, *geolokační objekt*, *MAC adresu* nebo *rozhraní*. MAC adresa a rozhraní jsou volitelné parametry, které se používají pouze, když je síťový objekt na straně zdroje (*Source*) a v jiných případech jsou ignorovány. Síťový objekt se navíc může odkazovat na další síťové objekty. (28)

Name	References	Entries	Comment
DYNAMIC			
STATIC			
admin1	1	192.168.1.201	
admin2	1	192.168.1.221	
admin3	1	Ref: admin1, Ref: admin2, Ref: Google-DNS	
Any	42	0.0.0.0/0	All IPv4 addresses
barracuda_FWs	0	10.10.11.41-45	
connect.barracuda.com	1	DNS ()	
dmz	1	172.16.1.54	
Eval Mode Bridged Ports	2	0.0.0.0/0 port1, 0.0.0.0/0 port2, 0.0.0.0/0 MGMT, 0.0.0.0/0 portA1, 0.0.0.0/0 p1, 0.0.0.0/0 p2	
google-172.217.32.0	0	172.217.32.0/20	hello
google-173.194.0.0	0	173.194.0.0/16	
Google-DNS	1	8.8.8.8	
Internet	1	Ref: Any, NOT 10.0.0.0/8, NOT 172.16.0.0/12, NOT 169.254.0.0/16, NOT 192.168.0.0/16	All routed IPv4 addresses
IP_83.22.64.25	1	83.22.64.25	webserver
pc-net	0	10.10.100.10-20	
Private 10	1	10.0.0.0/8	Private class A network

Obrázek 9: Barracuda CGFW - Network Objects (Zdroj: Vlastní zpracování)

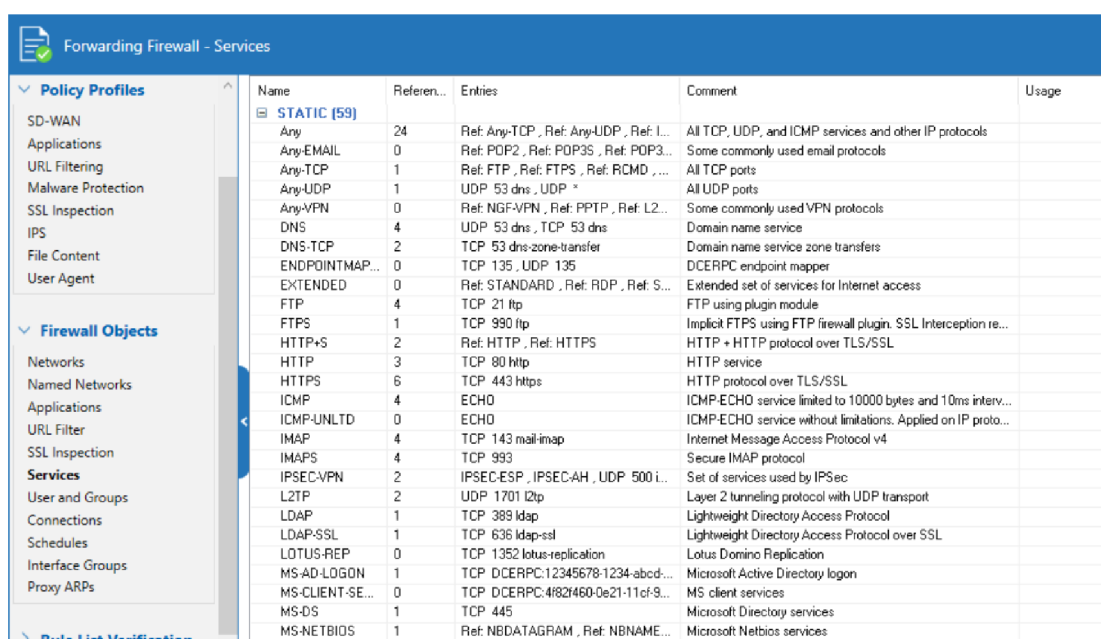
Využití těchto objektů zjednodušuje správu pravidel, například při změně IP adresy, kdy stačí změnit adresu v objektu a všechna pravidla se upraví sama, jestliže se na tento objekt odkazují. Avšak pokud je objekt někde použit, nemůže být odstraněn, dokud nebude odstraněn tento odkaz. (28)

Zde je seznam všech typů síťových objektů:

- **Obecné síťové objekty (Generic Network Objects)** – Mohou obsahovat síťové adresy všech typů (IP adresa, IP rozsah apod.). Obecné síťové objekty mohou obsahovat síťové objekty IPv4 i IPv6.
- **Jedna IPv4 adresa (Single IPv4 Address)** – Jedna IP adresa formátu IPv4.
- **Seznam IPv4 adres (List of IPv4 Addresses)** – Vícero jednotlivých IP adres a/nebo odkazy na jiné Single IPv4 Address objekty. Například: 10.0.10.1, 10.0.10.10, 10.0.10.127
- **Jedna IPv4 síť (Single IPv4 Network Address)** – Jedna síť ve formátu IPv4. Například: 10.0.10.0/25
- **Seznam IPv4 sítí (List of IPv4 Network Addresses)** – Jakákoli kombinace více sítí IPv4, IPv4 adres a/nebo odkazů na jiné objekty síťových adres. Například: 10.0.10.0/25, 172.16.0.10
- **Hostname (DNS Resolved)** – Název hostitele, který lze přeložit pomocí DNS. Například: myhost.test.com
- **Jedna IPv6 adresa (Single IPv6 Address)** – Jedna IP adresa IPv6.
- **Seznam IPv6 adres (List of IPv6 Addresses)** – Více adres IPv6 a/nebo odkazy na jiné Single IPv6 Address objekty.
- **Jedna IPv6 síť (Single IPv6 Network)** – Jedna síť IPv6.
- **Seznam IPv6 sítí (List of IPv6 Networks)** – Jakákoli kombinace více sítí IPv6, IPv6 adres a/nebo odkazů na jiné objekty síťových adres typu IPv6.
- **Sdružovací objekt (Pool Object)** – Pool Object poskytuje způsob, jak získat automaticky přidělené IP adresy z rozsahu sítě. Tento rozsah sítě lze volně konfigurovat. Pool Object je dostupný pouze pro Barracuda CC.
- **Vyloučené položky (Excluded Entries)** – konkrétní síť, které jsou vyloučeny ze síťového objektu. Pro transparentnost a konzistentnost zde nemohou být použity reference na jiné objekty. (28)

Service Objects

Service Objects neboli objekty služby definují, které cílové a klientské porty TCP/UDP nebo protokoly IP může služba aplikovaná na pravidlo používat. Ve výchozím nastavení obsahuje Barracuda CloudGen Firewall sadu předdefinovaných servisních objektů, které mohou být snadno upraveny nebo si administrátoři mohou vytvořit své vlastní podle svých potřeb. Mohou se navíc také odkazovat na již existující objekt služby. Během tvorby nových objektů by se měli administrátoři řídit pravidlem minimálních oprávnění, tedy specifikovat pouze potřebné porty v daném objektu, namísto třeba velkého rozsahu, který by mohl vytvořit větší útočnou plochu pro útočníka. (29)



Name	Referen...	Entries	Comment	Usage
STATIC [59]				
Any	24	Ref: Any-TCP , Ref: Any-UDP , Ref: I...	All TCP, UDP, and ICMP services and other IP protocols	
Any-EMAIL	0	Ref: POP2 , Ref: POP3S , Ref: POP3...	Some commonly used email protocols	
Any-TCP	1	Ref: FTP , Ref: FTPS , Ref: RCMD , ...	All TCP ports	
Any-UDP	1	UDP 53 dns , UDP *	All UDP ports	
Any-VPN	0	Ref: NGF-VPN , Ref: PPTP , Ref: L2...	Some commonly used VPN protocols	
DNS	4	UDP 53 dns , TCP 53 dns	Domain name service	
DNS-TCP	2	TCP 53 dns-zone-transfer	Domain name service zone transfers	
ENDPOINTMAP...	0	TCP 135 , UDP 135	DCERPC endpoint mapper	
EXTENDED	0	Ref: STANDARD , Ref: RDP , Ref: S...	Extended set of services for Internet access	
FTP	4	TCP 21 ftp	FTP using plugin module	
FTPS	1	TCP 990 ftp	Implicit FTPS using FTP firewall plugin. SSL Interception re...	
HTTP+S	2	Ref: HTTP , Ref: HTTPS	HTTP + HTTP protocol over TLS/SSL	
HTTP	3	TCP 80 http	HTTP service	
HTTPS	6	TCP 443 https	HTTP protocol over TLS/SSL	
ICMP	4	ECHO	ICMP-ECHO service limited to 10000 bytes and 10ms interv...	
ICMP-UNLTD	0	ECHO	ICMP-ECHO service without limitations. Applied on IP proto...	
IMAP	4	TCP 143 mail-imap	Internet Message Access Protocol v4	
IMAPS	4	TCP 993	Secure IMAP protocol	
IPSEC-VPN	2	IPSEC:ESP , IPSEC:AH , UDP 500 i...	Set of services used by IPSec	
L2TP	2	UDP 1701 l2tp	Layer 2 tunneling protocol with UDP transport	
LDAP	1	TCP 389 ldap	Lightweight Directory Access Protocol	
LDAP-SSL	1	TCP 636 ldap-ssl	Lightweight Directory Access Protocol over SSL	
LOTUS-REP	0	TCP 1352 lotus-replication	Lotus Domino Replication	
MS-AD-LOGON	1	TCP DCERPC:12345678-1234-abcd...	Microsoft Active Directory logon	
MS-CLIENT-SE...	0	TCP DCERPC:4f82460-0e21-11cf-9...	MS client services	
MS-DS	1	TCP 445	Microsoft Directory services	
MS-NETBIOS	1	Ref: NBDAAGRAM , Ref: NBNAME...	Microsoft Netbios services	

Obrázek 10: Barracuda CGFW – Service Objects (Zdroj: Vlastní zpracování)

Servisní objekt se může skládat z:

- **IP protokol (IP Protocol)** – Požadovaný protokol (např. TCP, UDP) pro službu používanou pravidlem.
- **Porty a rozsahy portů (Ports and Port Ranges)** – Porty nebo rozsahy portů, které může služba použít pro protokol.
- **Dynamické služby (Dynamic Services)** – Dynamické služby (např. dynamický směrovací protokol)

- **Zásuvné moduly (Plugin Modules)** – Zásuvné moduly pro sdílené objekt služby, které dynamicky přidělují porty.
- **Ochrana protokolů portů (Port Protocol Protection)** – *Port Protocol Protection* používá DPI pro vynucení použití zvoleného protokolu na portu. Spojení, které použijí nepovolený protokol mohou být nahlášeny nebo dokonce ukončeny. (29; 30)

1.6 Analýza rizik

Riziko má řadu definic, každopádně většinou se jedná o nějakou nepředvídanou či předvídanou situaci, kdy dojde k odchýlení se od požadovaných výsledků. Toto odchýlení může být ve formě ztráty, zpoždění, snížení tržeb, poškození aktiva a mnohé další. Důležitým faktem ale zůstává, že rizika jsou součástí podnikání, ať už se týkají projektů, financí nebo kybernetické bezpečnosti, proto by měla být řízena a minimalizována. (31; 32)

Analýza rizik je jednou z prvních analýz, která probíhá na začátku projektu. Díky identifikovaným rizikům je možné zhodnotit, zda je projekt vhodné vůbec realizovat, odsunout jej do budoucna, kdy pro něj budou lepší podmínky nebo se do něj vůbec nepouštět. Cílem je také umožnit řízení těchto rizik, jako je třeba minimalizace dopadů nebo pravděpodobnosti výskytu a umožnit tak úspěšné dokončení projektu a šetření finančních a jiných zdrojů. (31, 33)

Množství rizik se mění s podstatou projektu, jeho délkou, komplexností, proměnlivostí jeho prostředí a také zkušenostmi členů projektového týmu. Rizika jsou navíc během doby trvání projektu proměnlivá, protože se mohou vyskytovat nová rizika, rizika s nízkým dopadem se mohou proměnit na rizika s vysokým dopadem a pravděpodobnost jejich výskytu se může také změnit, proto je důležité rizika sledovat i po samotné analýze, volit nová opatření, když si to situace žádá a nenechat se ničím překvapit. (31, 33)

Z výše uvedeného vyplývá, že řízení rizik se skládá ze dvou částí, a to analýzy rizik a sledování rizik. Analýza rizik se skládá z identifikace rizik, ohodnocení a posouzení rizik projektu a následné reakce na identifikovaná rizika. Pravděpodobně nejdůležitější částí je identifikace rizik, protože závisí hodně na zkušenostech projektového týmu a také

na množství a kvalitě informací, které má k dispozici. Jednou z možných metod pro analýzu rizik je metoda *RIPRAN*. (33)

1.6.1 RIPRAN

Zmíněná metoda RIPRAN (**RI**sks **PR**ojekt **AN**alysis) byla vyvinuta pracovníkem VUT Brno, doc. Ing. Branislavem Lacko CSc. a je nyní běžně používaná v praxi během řízení rizik. Skládá z několika fází, které zahrnují identifikaci rizik, kvantifikaci rizik, reakci na rizika a celkové posouzení rizik. Před těmito čtyřmi fázemi probíhá také příprava na samotnou analýzu, jako například stanovení plánu a postupu a zajištění potřebných podkladů. Výsledkem je tabulka s identifikovanými ohodnocenými riziky, na které jsou vytvořena opatření. (31; 34)

Identifikace rizik projektu

V první fázi jsou identifikována všechna možná rizika, která mohou ohrozit projekt nebo organizaci jako celek. Rizika mohou být identifikována pomocí brainstormingu, z historických dat minulých projektů, z registru rizik společnosti nebo pouze na základě zkušeností a odhadu. Jak již bylo zmíněno, nikdy nebudeme schopni identifikovat všechna rizika, musíme však identifikovat ty nejrelevantnější a nejzávažnější, proto je během této fáze snaha nalézt co největší množství rizik. (35)

Je možné postupovat dvěma základními způsoby. Prvním způsobem je hledání scénáře k hrozbě a druhým je hledání hrozby ke scénáři. Hrozba představuje projev konkrétního nebezpečí a scénář je důsledek tohoto projevu, takže se můžeme ptát na otázky typu „*Co se stane, když?*“ nebo „*Aby se něco stalo, co musí předcházet?*“ Výstupem je však tabulka, ve které bude zapsané obojí, společně s identifikačním číslem rizika (ID) a případné poznámky. (35)

Tabulka 1: Tabulka RIPRAN (Zdroj: Vlastní zpracování dle: (35))

ID	Hrozba	Scénář	Poznámky
1	Slabě nastavené heslo	Přístup k aplikaci/účtu bude odcizen a může dojít ke zneužití	Zneužití může zahrnovat: ztrátu údajů, pošpinění jména, další šíření malwaru, ...
2	Nedostatečné chlazení serverů	Servery se budou přehřívat, zpomalovat a v horším případě vznikne požár	...
...			

Kvantifikace rizik

Nyní je potřeba stanovit hodnotu identifikovaných rizik, proto je nutné určit pravděpodobnost výskytu a velikost dopadu. Pravděpodobnost výskytu nám říká, jaká je šance, že se hrozba realizuje a tím pádem i možný scénář. Dopad poté udává ujmu způsobenou projektem nebo organizací, jestliže dojde k naplnění rizika. Újmou se nejčastěji rozumí finanční škoda, ale může sem patřit i ekologický dopad nebo újma na zdraví a další možné negativní projevy. (31; 36)

Pro hodnocení pravděpodobnosti a dopadu je možné použít jak číselné, tak i verbální hodnocení pomocí tabulek. Styl a škálu si volí projektový tým sám podle potřeby projektu. Pro lepší výsledky navíc každý člen provede vlastní zhodnocení rizik a následně jsou hodnocení porovnávána, diskutována a následně upravena na všemi akceptovanou hodnotu. Níže jsou uvedeny nejčastěji používané škály pro hodnocení rizik, avšak v metodě RIPRAN existují i jiné škály pro podrobnější analýzu. (31; 36)

Tabulka 2: Verbální ohodnocení pravděpodobnosti výskytu rizika (Zdroj: Vlastní zpracování dle: (37))

Pravděpodobnost výskytu rizika	
Vysoká pravděpodobnost – VP	Nad 66 %
Střední pravděpodobnost – SP	33 až 66 %
Nízká pravděpodobnost – NP	Do 33 %

Tabulka 3: Verbální ohodnocení dopadu rizika (Zdroj: Vlastní zpracování dle: (37))

Velikost dopadu rizika	
Velký dopad – VD	• Ztráta nad 20 % hodnoty projektu • Ohrožení cíle projektu • Ohrožení termínu projektu
Střední dopad – SD	• Ztráta mezi 1 % až 20 % hodnoty projektu • Ohrožení některých částí projektů • Ohrožení některým termínů činností
Nízký dopad – ND	• Ztráta do 1 % hodnoty projektu • Nutné menší úpravy v plánu projektu

Jestliže rizika ohodnotíme z pohledu pravděpodobnosti výskytu a velikosti dopadu, je možné vypočítat hodnotu rizika. V případě verbálního hodnocení je potřeba použít jednu z příslušných matic. Pro aktuální škály se používá matice viz níže.

- VHR – Vysoká hodnota rizika
- SHR – Střední hodnota rizika
- NHR – Nízká hodnota rizika (36, 37)

Tabulka 4: Matice pro přiřazení třídy hodnoty rizika (Zdroj: Vlastní zpracování dle: (37))

		Pravděpodobnost výskytu rizika		
		VP	SP	NP
Velikost dopadu rizika	VD	VHR	VHR	SHR
	SD	VHR	SHR	NHR
	ND	SHR	NHR	NHR

Tabulka 5: Tabulka RIPRAN – Ohodnocená (Zdroj: Vlastní zpracování dle: (35))

ID	Hrozba	Scénář	Pravděpodobnost	Dopad	Hodnota rizika	Poznámky
1	Slabě nastavené heslo	Přístup k aplikaci/účtu bude odcizen a může dojít ke zneužití	SP	VD	VHR	Zneužití může zahrnovat: ztrátu údajů, pošpinění jména, další šíření malwaru, ...
2	Nedostatečné chlazení serverů	Servery se budou přehřívat, zpomalovat a v horším případě vznikne požár	NP	VD	SHR	...
...						

Reakce na rizika

Po provedení hodnocení rizik je nutné vytvořit opatření pro jednotlivá rizika, aby došlo ke snížení pravděpodobnosti jejich výskytu či alespoň zmírnění dopadu. Některá rizika je možné pouze snížit, jiná úplně eliminovat a někdy není možné riziko, jakkoliv ovlivnit, v takových případech je nejlepší se riziku vyhnout nebo jej podstoupit, jestliže je to

přijatelné. Kombinace opatření může také přinést lepší efekt než jedno samostatné opatření. Výběr opatření závisí především na kreativitě projektového týmu, jeho zkušenostech, dostupných zdrojích a možných technických alternativách. Po návrhu opatření se občas také určuje nová hodnota rizika. (31; 38)

Během výběru opatření je několik zásad, které pomáhají při výběru. První zásadou je fakt, že náklady na opatření by nikdy neměly převýšit újmu, kterou by riziko mohlo způsobit, v takovém případě by nedávalo takovéto řešení smysl, i kdyby kompletně eliminovalo riziko. Další zásadou je prioritizace rizik s vysokou hodnotou rizika, protože představují největší hrozbu pro projekt a společnost. V neposlední řadě je důležité pravidelné kontrolování a úpravy rizik podle změn v projektu, protože se mohou měnit, objevovat a zvyšovat. Jedna věc je také navrhnout opatření na riziko, druhá je potom samotná implementace opatření, jestliže se opatření neaplikuje, hodnota rizika se nezmění, ať už je na něj vymyšleno sebelepší opatření. (31; 38)

Vysoká hodnota rizika

Jak již bylo zmíněno při výběru opatření by rizika s vysokou hodnotou měla mít nejvyšší prioritu, protože mají největší vliv na výsledek a životaschopnost projektu nebo organizace. Takové riziko je potřeba snížit na dostatečně nízkou úroveň nebo se mu vyhnout. V krajních případech je možné takové riziko podstoupit, což ale musí rozhodnout vedení společnosti. (31; 33)

Střední hodnota rizika

Na rizika se střední hodnotou se většinou vytváří opatření, která nejsou příliš finančně a ani časově náročná. Hraje zde tedy roli poměr nákladů na opatření a výše škody při realizaci rizika. Tato rizika je také možné akceptovat, pokud není žádné možné řešení. Sledování těchto rizik by nemělo být podceňováno, protože jsou pouze jeden stupeň od vysoké hodnoty a mohou se tam záhy dostat. (31; 33)

Nízká hodnota rizika

Nízká hodnota rizika je v mnoha případech považována za příliš nepodstatnou a jsou tedy akceptována. V případě výběru opatření, se volí taková opatření, která nestojí téměř žádné finanční zdroje nebo čas. To jsou většinou organizační zásady, zásady práce

a chování a podobně. Pořád je však doporučeno sledovat vývoj rizika během projektu. (31; 33)

Tabulka 6: Tabulka RIPRAN – Návrh na opatření (Zdroj: Vlastní zpracování dle: (35))

ID	Hrozba	Pravděpo- dobnost	Dopad	Hodnota rizika	Návrh opatření	Nová hodnota rizika
1	Slabě nastavené heslo	SP	VD	VHR	Vytvoření politiky pro vytváření bezpečných hesel, dvoufaktorová autentizace, minimalizace práv	NHR
2	Nedostatečné chlazení serverů	NP	VD	SHR	Nákup kvalitních chladících jednotek, instalace teploměru a sledování ukazatelů, záložní PSU i pro chlazení	NHR
...						

Celkové posouzení rizik

Závěrečnou fází analýzy rizik je celkové posouzení identifikovaných rizik. Žádné z rizik by nemělo překročit akceptovatelnou úroveň, vycházející z vnitřní politiky firmy a správně by nemělo být žádné riziko s vysokou hodnotou po aplikaci opatření. V takovém případě je další rozhodnutí podstoupeno vedení společnosti. (31; 39)

Z analýzy rizik by měla být vytvořena závěrečná zpráva, která dává všechny její fáze dohromady. Obsahující také datum vytvoření a data další možné revize, zodpovědné osoby za implementace opatření a kompletní rejstřík nebo tabulku identifikovaných rizik s jejich charakteristikami. Projektový manažer nebo odpovědná osoby by měla nadále sledovat vývoj rizik. (33; 39)

Tabulka 7: Kompletní tabulka RIPRAN (Zdroj: Vlastní zpracování dle: (35))

ID	Hrozba	Scénář	Pravděpo- dobnost	Dopad	Hodnota rizika	Návrh opatření	Nová hodnota rizika	Poznámky
1	Slabě nastavené heslo	Přístup k aplikaci/úctu bude odcizen a může dojít ke zneužití	SP	VD	VHR	Vytvoření politiky pro vytváření bezpečných hesel, dvoufaktorová autentizace, minimalizace práv	NHR	Zneužití může zahrnovat: ztrátu údajů, pošpinění jména, další šíření malwaru, ...
2	Nedostatečné chlazení serverů	Servery se budou přehřívat, zpomalovat a v horším případě vznikne požár	NP	VD	SHR	Nákup kvalitních chladících jednotek, instalace teploměru a sledování ukazatelů, záložní PSU i pro chlazení	NHR	...
...								

1.7 Časová analýza

Časová analýza je nedílnou součástí řízení projektů a projektů obecně, protože jejím cílem je zajistit nepřerušenu kontinuitu činností a jejich včasné dokončení v souladu s plánem, a tedy i celého projektu, aby mohl být považován za úspěšný. Do časové analýzy spadá řada metod, nástrojů a pomůcek, které je možné při její tvorbě použít, jako například CPM, PERT, ganttovy diagramy, síťové grafy aj. (33)

Jako první část časové analýzy většinou bývá stanovení jednotlivých aktivit, úkolů a činností, které je nutné provést pro naplnění vyšších celků. K jednotlivým činnostem jsou následně přiřazeny další charakteristiky jako je jejich trvání, závislost na dalších aktivitách a požadavky na zdroje. Nakonec vznikne detailní plán, který říká, jak dlouho přibližně bude projekt trvat, v jaké posloupnosti budou muset být činnosti realizovány a jaké budou přibližné náklady. (31; 33)

Mezi použité metody a nástroje během projektu byla využita metoda CPM (Critical Path Method) a síťové grafy. Metoda CPM pomáhá zjistit, které činnosti projektu leží na takzvané kritické cestě. Jestliže se činnosti na kritické cestě zpozdí, tak se zpozdí i dokončení celého projektu. Na druhé straně síťové grafy jsou grafický nástroj pro vyjádření vztahů, návaznosti a toku činností. Pomáhají lépe vizualizovat nejen tyto vztahy, ale i kritickou cestu. (33; 40)

Ve finále můžeme říct, že časová analýza pomáhá průběžně sledovat a kontrolovat stav plnění naplánovaných činností a tím i míru plnění projektu. Toto sledování umožňuje včas identifikovat možné zpoždění a učinit nápravné opatření a zajistit tak dokončení projektu v daném časovém rámci a splnění cíle projektu v požadované kvalitě. (33)

1.7.1 CPM

Metoda kritické cesty (Critical Path Method) je jedním z velmi používaných nástrojů pro projektové řízení a plánování. Pomáhá stanovit kritickou cestu, tedy nejkratší možnou dobu, za kterou lze projekt dokončit a stanovit tak činnosti, které se nemohou zpozdít, aby nedošlo ke zpoždění celého projektu. Kritická cesta je zároveň nejdelší cestou

v projektu, tedy nejdelší sled po sobě jdoucích činností bez přerušení od začátku až do konce. (40)

CPM lze řešit tabulkově nebo v kombinaci se síťovými grafy, ale základ zůstává stejný. Jde o postupný výpočet časových charakteristik jednotlivých činností na základě jejich vzájemných vztahů. Mezi tyto charakteristiky patří *ZM (začátek možný)*, tedy nejdřívější možný termín, kdy může daná činnost začít, *KM (konec možný)*, což je nejdřívější možný termín dokončení činnosti, tedy bez jakéhokoliv zpoždění, *ZP (začátek přípustný)* je naopak nejpozdější termín, kdy může činnost začít, aby nedošlo ke zpoždění projektu a *KP (konec přípustný)* je nejpozdější termín, kdy činnost může být dokončena a nedojde ke zpoždění projektu. Postup výpočtu těchto charakteristik probíhá ve dvou průchodech činností. (33; 40)

V první řadě jsou činnosti procházeny od začátku do konce a jsou zde vypočítány hodnoty *ZM (začátek možný)* a *KM (konec možný)*. Začátek možný je stanoven jako konec možný předchozí činnosti. U první činnosti projektu je výjimka, protože je *ZM* roven nule, jelikož této činnosti nepředchází žádná jiná činnost, a tedy nemusí čekat na své zahájení. Jestliže některé činnosti předchází více jak jedna činnost, je jako *ZM* zvolena vyšší hodnota *KM* z předchozích činností. Konec možný je stanoven součtem činnosti a její délkou trvání. Hodnota *KM* poslední činnosti je také délka trvání celého projektu. (40)

$$KM = ZM + t$$

Rovnice 1: Vzorec - Konec možný (Zdroj: Vlastní zpracování dle: (40))

- t – délka trvání činnosti

Druhý průchod je zároveň kontrolní, protože procházíme od konce činností zpátky na začátek a výsledná hodnota *ZP* by měla být rovna nule. Začneme tím, že určíme *KP* poslední činnosti. Ten je roven *KM* a poté od něj odečteme délku trvání činnosti a získáme *ZP*. *KP* předposlední činnosti je potom *ZP* činnosti poslední a tak dále a tak dále. Jestliže se opět dostaneme do situace, kde bude na výběr z více hodnot, zvolíme nižší *ZP*. (40)

$$ZP = KP - t$$

Rovnice 2: Vzorec - Začátek přípustný (Zdroj: Vlastní zpracování dle: (40))

Po druhém průchodu je ještě potřeba vypočítat *RC* (*rezerva celková*), což uděláme odečtem *KM* od *KP* dané činnosti nebo *ZM* od *ZP*. Tato rezerva udává o kolik je možné, aby se konkrétní činnost zpozdila a nedošlo tak ještě ke zpoždění celého projektu. Činnosti s nulovou rezervou leží na již zmíněné kritické cestě a musí být kladen důraz na jejich včasné splnění, protože jakékoliv zpoždění těchto činností povede ke zpoždění celého projektu. (40)

$$RC = KP - KM = ZP - ZM$$

Rovnice 3: Vzorec - Rezerva celková (Zdroj: Vlastní zpracování dle: (40))

1.7.2 Síťové grafy

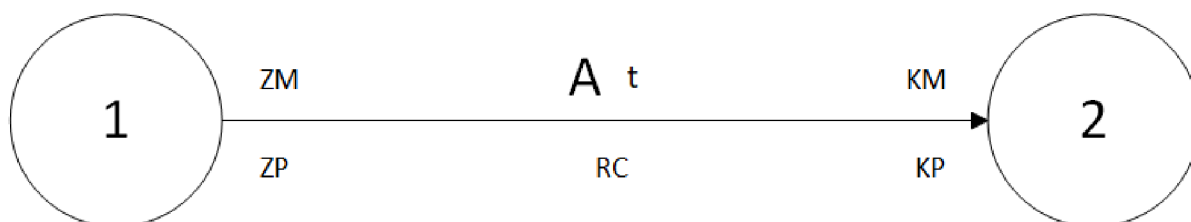
Síťové grafy jsou grafická pomůcka, která se používá v kombinaci s metodou CPM. Samotná metoda CPM ve formě tabulky může být značně složitá pro monitorování a jediné, co z ní bez většího úsilí získáme, je kritická cesta. Obtížnost sledování závislosti a návaznosti činností v tabulce roste exponenciálně se složitostí projektu. Přesně tyto obtíže a problémy řeší síťové grafy. (31; 40)

Síťové grafy jsou tvořeny ze dvou základních prvků, *uzlů* a *hran*. Uzly nebo hrany mohou znázorňovat činnosti nebo propojení mezi nimi, to je dané zvoleným typem síťového grafu. V praxi se můžeme setkat se dvěma typy síťových grafů, avšak nezávisle na typu grafu musí splňovat tyto podmínky:

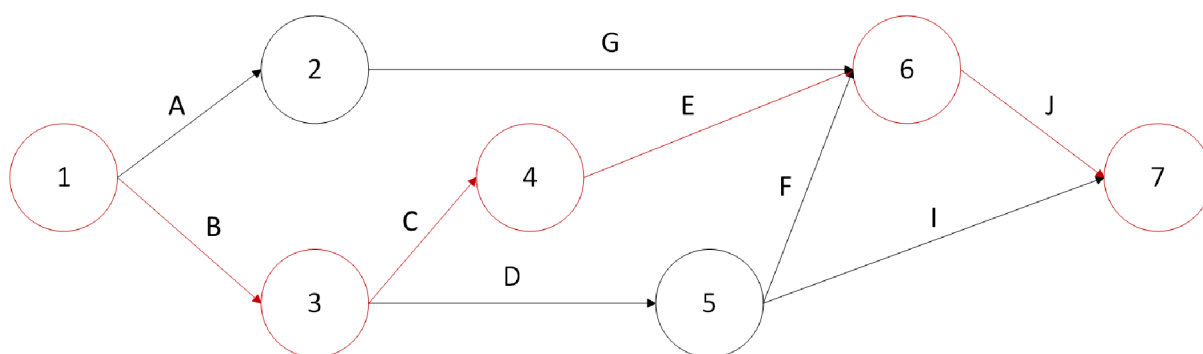
1. Síťový graf má právě jeden začátek a jeden konec
2. Síťový graf je konečný (má konečný počet vrcholů a hran)
3. Síťový graf je acyklický (žádná činnost se nesmí vracet do předešlé)
4. Síťový graf je souvislý (neobsahuje izolované uzly nebo hrany)
5. Síťový graf je orientovaný (vazby jsou dané šipkami)
6. Síťový graf je jednoduchý (mezi dvěma uzly nemůže vzniknout dvojité hrany)
7. Síťový graf je ohodnocený (činnosti jsou ohodnoceny například časem) (31; 40)

Hranově definovaný graf

Jeden z typů síťových grafů je hranově definovaný graf. Tato forma síťového grafu znázorňuje činnosti jako hrany, které jsou opatřeny časovými charakteristikami a kde jednotlivé uzly značí začátek a konec činnosti. Během jeho tvorby může často dojít k situacím, kdy je obtížné propojit jednotlivé činnosti a je proto vytvořena fiktivní činnost, která má nulovou délku trvání, což může vést k větší nepřehlednosti. Hranově definované grafy jsou v praxi již méně používané. (31; 40)



Obrázek 11: Síťový graf – hrana (Zdroj: Vlastní zpracování dle: (40))



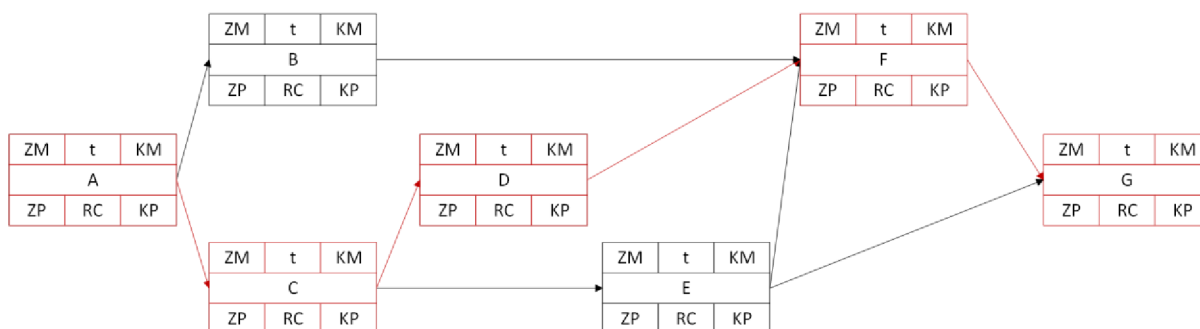
Obrázek 12: Příklad hranově definovaného síťového grafu bez hodnot (Zdroj: Vlastní zpracování dle: (40))

Uzlově definovaný graf

V praxi používanější typ síťového grafu je uzlově definovaný graf. Jak název napovídá jednotlivé činnosti jsou znázorněny uzly a hrany tvoří vazby mezi těmito činnostmi. Uzly jsou v tomto případě opatřeny časovými charakteristikami, na rozdíl od hran v hranově definovaném grafu. Nevznikají tedy také situace, kdy je potřeba propojit činnosti pomocí fiktivní činnosti. (31; 40)



Obrázek 13: Síťový graf – uzel (Zdroj: Vlastní zpracování dle: (40))



Obrázek 14: Příklad uzlově definovaného síťového grafu bez hodnot (Zdroj: Vlastní zpracování dle: (40))

1.8 RACI matice

Aby bylo jasné, kdo odpovídá za vypracování a splnění jednotlivých činností projektu, je možné vytvořit RACI matici. RACI matice vznikne propojením stanovených činností projektu a členů podílejících se na jeho uskutečnění. Níže jsou definované typy odpovědností, se kterými se můžeme při tvorbě RACI matice setkat. (31)

- **R – Responsible (Realizuje):** Osoba, která je zodpovědná za fyzické provedení činnosti nebo činností spojené s projektem. Těchto osob může být u dané činnosti i více, jestliže to je potřeba.
- **A – Accountable (Zodpovídá):** Je nutné určit osobu zodpovědnou za správné a včasné dokončení jednotlivých činností, a to při zachování stanovených nákladů. Tato osoba se nemusí fyzicky podílet na realizaci činnosti, avšak zodpovídá za stav plnění. U každé činnosti může být pouze jedna takováto osoba, aby bylo jasné, kdo je zodpovědný.
- **C – Consulted (Konzultuje):** Jedná se většinou o osobu, která se nemusí podílet na realizaci činnosti nebo celého projektu, ale má značné zkušenosti a znalosti v dané problematice, a proto může při jejím plnění poskytnout jinou cennou

pomoc, jako je například odborná rada. Těchto osob může být u jedné činnosti i více.

- **I – Informed (Informován):** Informované osoby se na realizaci činnosti nepodílejí, ale jsou informované o jejím průběhu, stavu plnění a jiných důležitých detailech, které by tyto osoby měly znát. Často bývá informováno o stavu projektu třeba vedení společnosti. (31; 40)

Tabulka 8: RACI matice – příklad (Zdroj: Vlastní zpracování dle: (31))

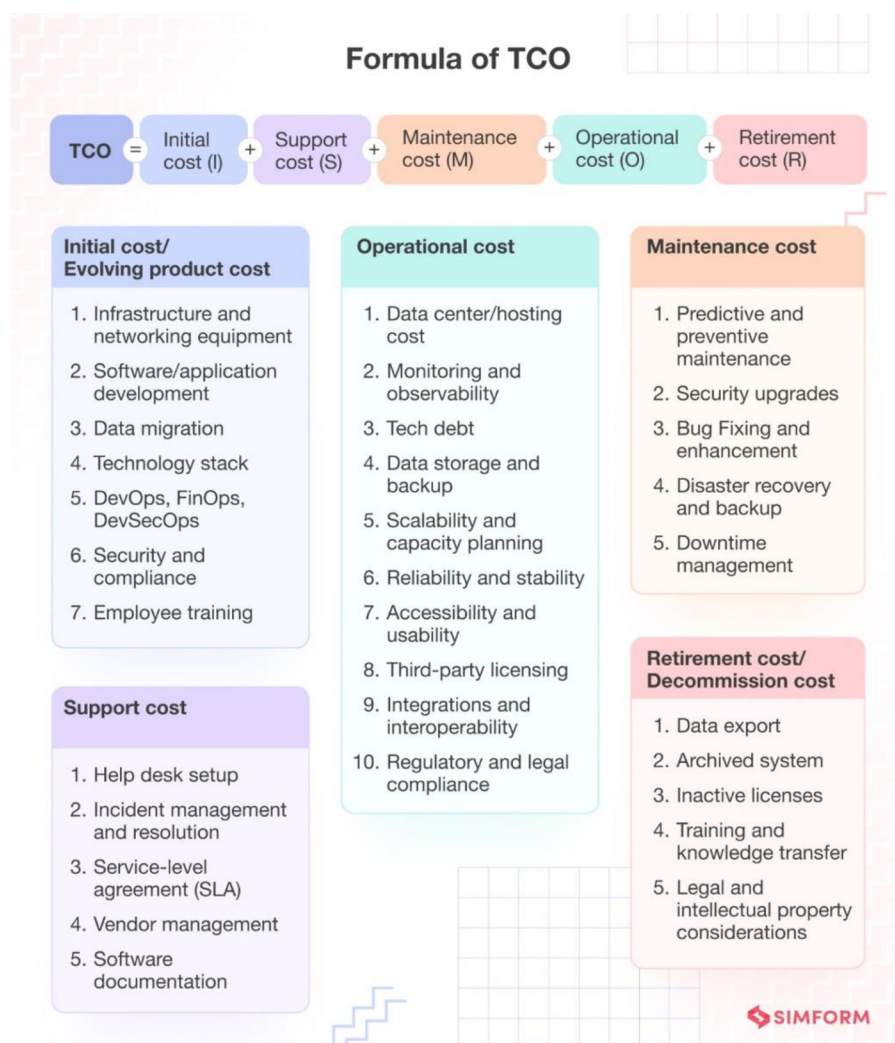
Činnost	Vedoucí pracovník	Technik	Odborný konzultant	Majitel společnosti
Příprava technického řešení	A/C	R	C	I
Instalace a konfigurace systému	I	R/A	C	I
Testování a validace systému	C	R/A	C	I
Předání systému	R/A	I		R
Školení uživatelů	I	R/A		I

1.9 Total Cost of Ownership

Celkové náklady na vlastnictví produktu, z angličtiny Total Cost of Ownership (TCO), představují celkové náklady spojené s vlastnictvím a provozem produktu po celou dobu jeho životního cyklu. Spadají sem počáteční náklady na nákup nebo vývoj a průběžné výdaje, jako je údržba, upgrady, licenční poplatky, školení a podpora. (41; 42)

Pro výpočet celkových nákladů na vlastnictví by se mělo vyhodnotit více než počáteční náklady jako jsou tvrdé náklady spojené s pořízením softwaru a hardwaru, a měkké náklady, jako je školení a údržba. Tato metoda výpočtu bere v úvahu jak přímé, tak i nepřímé náklady, které mohou být známe jako *skryté* náklady, a které bývají často opomíjeny, což vede k situacím, kdy se společnosti musí vzdát svých plánů, protože nepočítaly se všemi možnými náklady. (41; 42)

Pro výpočet TCO byste měli nejprve definovat rozsah a časový rámec pro které je potřeba tyto náklady kalkulovat, což může být očekávaná životnost zařízení, konkrétní smluvní období nebo doba trvání projektu. (41; 42)



Obrázek 15: Typy nákladů, ze kterých se skládá TCO (Zdroj: (43))

Poté byste se měli podívat na každou kategorii výdajů a určit všechny odpovídající náklady. Různé druhy nákladů lze identifikovat jako:

- **Počáteční náklady (Initial Costs)** – Počáteční náklady zahrnují náklady na software a hardware, vývoj produktu, implementaci, migraci dat, uživatelské licence, školení, externí systémová rozhraní a přizpůsobení.
- **Provozní náklady (Operational Costs)** – Provozní náklady vznikají během celé doby používání produktu. Zahrnují mimo jiné obnovování licencí, ukládání dat a školení uživatelů.

- **Náklady na údržbu (Maintenance Costs)** – Jak je produkt používán, objevují se náklady na průběžnou údržbu a podporu, které mohou zahrnovat náklady související s úpravou kódu, upgrady infrastruktury a opravami chyb nebo havárií. Odstávky jsou další aspekt, který se obvykle nebere v úvahu, ale může mít velký dopad.
- **Náklady na podporu (Support Costs)** – Jsou to náklady související s upgrady softwaru, vysokou dostupností, zabezpečením a školením zaměstnanců. Migrace uživatelů na nové systémy také vytváří dodatečné náklady především během přechodného období.
- **Náklady na vyřazení (Retirement Costs)** – Na konci životního cyklu se často objevují náklady spojené s přechodem na jiný produkt, jako je migrace dat, archivace, školení a přenos znalostí a právní náklady. (41; 42)

Na závěr je nutné dodat, že TCO lze vypočítat několika způsoby v závislosti na typu daného produktu nebo služby a především odvětví. Neexistuje žádné absolutní řešení pro stanovení TCO a u každého řešení je mnohem lepší zvážit specifika každého oboru podnikání. Pro zjednodušení a větší přehlednost se náklady na údržbu a podporu často přidávají k provozním nákladům, je však důležité na žádné z těchto částí nezapomenout. (41; 42)

1.10 REST API

API v originále anglicky Application Programming Interface je sada pravidel a protokolů tvořící způsob ovládání a interakce se softwarovými aplikacemi. REST API, anglicky REpresentational State Transfer API, je jeden z jeho typů umožňující komunikaci po síti prostřednictvím protokolu HTTP/S a jeho metod jako je *POST*, *GET*, *PUT*, *DELETE* a dalších ve formě požadavků (*requests*). Hlavní podstatou REST API je *bezstavová* komunikace mezi klientem a serverem, což znamená, že žádný předchozí požadavek neovlivní jakýkoliv následující, jelikož o něm není uchováván žádný kontext, který by to umožňoval a je tedy nutné při každém novém požadavku zadat všechny potřebné parametry. Každý požadavek se skládá z použité metody, cílového URL, hlaviček a těla. (44; 45)

Základní HTTP metody REST API jsou:

- **GET** – navrátí požadovaná data ze serveru
- **POST** – odešle data serveru pro zpracování (např. vytvoření nového zdroje/objektu)
- **PUT** – aktualizuje zdroj/objekt na serveru pomocí nových dat
- **DELETE** – smaže zdroj/objekt ze serveru (44; 45)

Po zvolení metody je nutné specifikovat cestu URL (Uniform Resource Locator) k požadovanému zdroji na serveru, což zahrnuje protokol HTTP/S, hostname, port a samotnou cestu. Dále je potřeba specifikovat hlavičky (*headers*), obsahující důležitá metadata, jako například formát (*content type*), způsob kódování (např. *UTF-8*), jazykové preference a autentizaci. Autentizace může probíhat pomocí klasického jména a hesla, tokenů nebo API klíčů. (44; 45)

Tělo (*body*) je tvořeno parametry, do kterých klient vkládá požadované hodnoty. Tyto parametry, jejich uspořádání a hloubka se liší dle cílové aplikace a podporovaného formátu, kde jako příklad můžeme uvést formát JSON (JavaScript Object Notation), který je jednoduchý na pochopení a velmi rozšířený. Na obrázku níže je uveden příklad této skladby parametrů pro vytváření síťových objektů v rámci REST API Barracuda CGFW využívající formát JSON. (44; 45)

```

{
  "color": "#ff0000",
  "comment": "This is my object.",
  "dnslifetime": 600,
  "excluded": [
    {
      "entry": {
        "comment": "My Comment",
        "interface": "eth0",
        "ip": "127.0.0.1",
        "mac": "11:22:33:44:55:66"
      }
    }
  ],
  "geo": {
    "excluded": [
      "Austria",
      "Europe"
    ],
    "included": [
      "Austria",
      "Europe"
    ]
  },
  "included": [
    {
      "entry": {
        "comment": "My Comment",
        "interface": "eth0",
        "ip": "127.0.0.1",
        "mac": "11:22:33:44:55:66"
      },
      "references": "MyOtherObject"
    }
  ],
  "name": "MyObject",
  "type": "generic"
}

```

Obrázek 16: REST API - Požadavek (Zdroj: (46))

Na obrázku č. 16 můžeme vidět, že JSON se skládá z objektů a polí. *Objekty* tvoří páry klíč/hodnota (*key/value*), které jsou ohraničeny uvozovkami a odděleny dvojtečkou, celý objekt je potom ohraničen složenými závorkami, zde můžeme uvést jako příklad obrázek č. 17. *Pole* tvoří seznam hodnot uvnitř hranatých závorek, jako příklad může sloužit obrázek č. 18. (45; 46)

```
"comment": "My Comment",
```

Obrázek 17: REST API – Objekt (Zdroj: (46))

```

"excluded": [
  "Austria",
  "Europe"
],

```

Obrázek 18: REST API – Pole (Zdroj: (46))

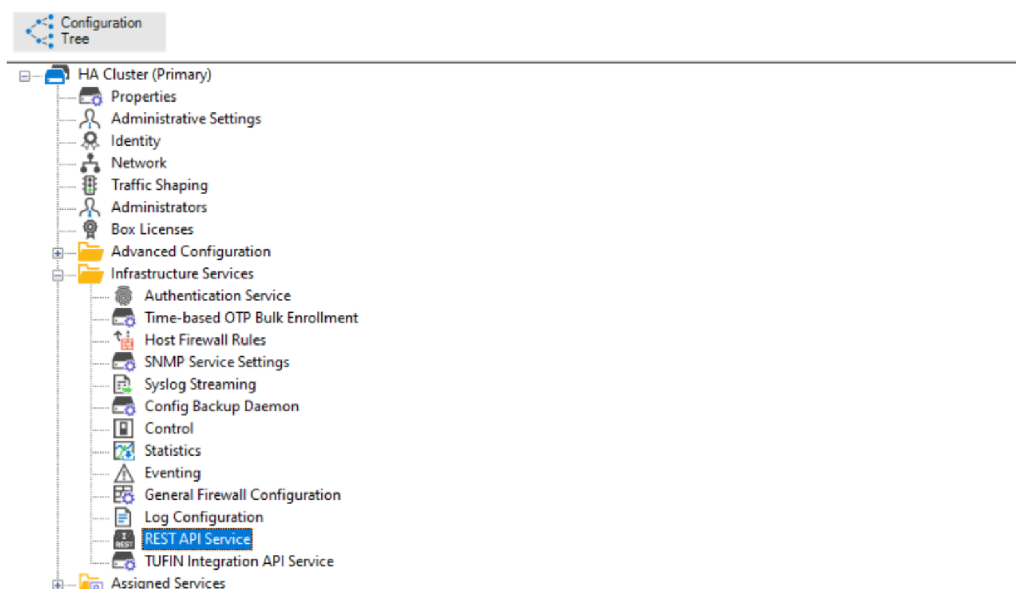
1.10.1 Stavové kódy

REST API používá standardní HTTP stavové kódy pro sdělení výsledků požadavků. Tyto kódy nám říkají, zda byl požadovaný dotaz úspěšný nebo ne a podle toho můžeme náš požadavek upravit a odeslat znovu. Zde je základní výčet těchto statusových kódů:

- **1xx: Informational (Informační)** – Informuje klient, že požadavek stále probíhá.
 - **2xx: Success (Úspěch)** – Požadavek klienta byl úspěšně přijat.
 - **3xx: Redirection (Přesměrování)** – Pro dokončení požadavku, je potřeba aby klient provedl nějakou další akci.
 - **4xx: Client Error (Chyba klienta)** – Tato kategorie chybových stavových kódů říká, že je chyba na straně klienta.
 - **5xx: Server Error (Chyba serveru)** – Chyba je tentokrát na straně serveru.
- (47)

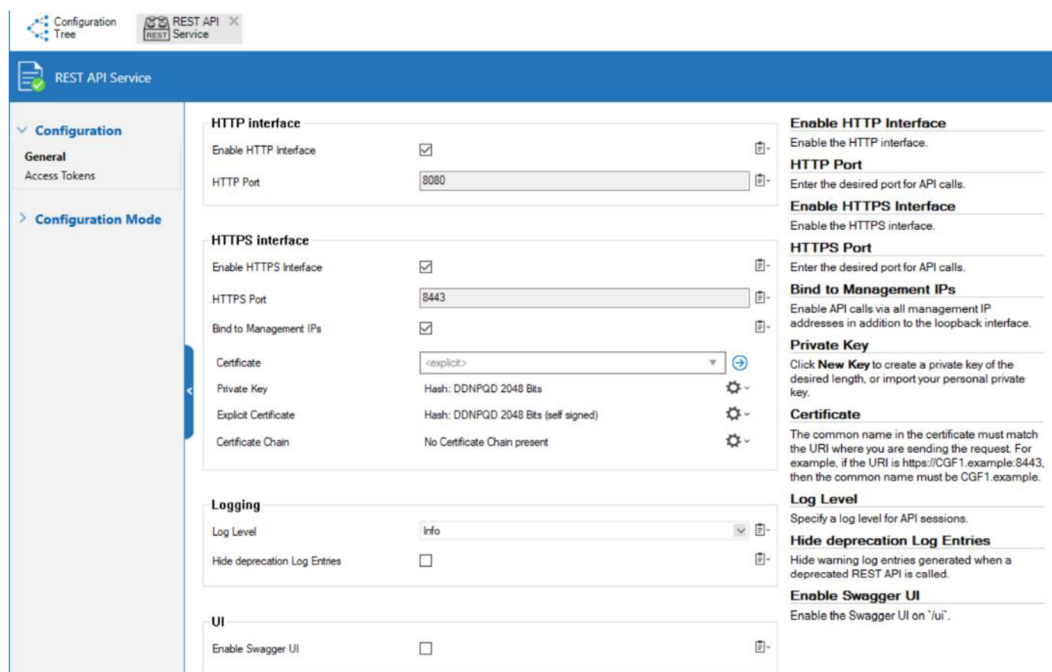
1.10.2 Barracuda Firewall REST API

Barracuda CGFW REST API poskytuje vzdálenou správu a konfiguraci Barracuda CGFW pomocí již zmíněného formátu JSON. Na stránku REST API Service, kde probíhá jeho konfigurace, je možné se dostat z konfiguračního stromu z uzlu *Infrastructure Services*, tak jak je to znázorněno na obrázku níže. (48)



Obrázek 19: Barracuda CGFW – Konfigurační strom – REST API Service (Zdroj: Vlastní zpracování)

Aby bylo možné spravovat Barracuda CGFW pomocí REST API je potřeba povolit REST API v nastavení, ať už se jedná o protokol *HTTP* nebo *HTTPS*. Zároveň musí být tato komunikace povolena *App Redirect* pravidlem, které se vytvoří v *Access Rules*. V neposlední řadě vyžaduje Barracuda CGFW ověření všech požadavků, což lze provádět například pomocí tokenů nebo jménem a heslem administrátora s dostatečným oprávněním. (48)



Obrázek 20: Barracuda CGFW – Konfigurace REST API Service (Zdroj: Vlastní zpracování)

1.11 Python

Python je vysokoúrovňový, interpretovaný programovací jazyk, který umožňuje rychlejší pochopení, než například C++ nebo Java díky své jednoduchosti syntaxe a čitelnosti kódu. Jedná se o open source nástroj, což znamená, že se vývojáři neustále podílejí na jeho rozšiřování a rozvoji. Široký výběr knihoven a jejich snadná integrace z něj navíc dělá skvělý nástroj pro různorodé úkoly jako je vývoj aplikací nebo interakce s jinými programy. (49)



Obrázek 21: Logo Python (Zdroj: (50))

1.11.1 Vestavěné prvky

Python nabízí řadu *vestavěných prvků*, jako jsou vestavěné funkce, klíčová slova a speciální syntaxe, které vývojářům pomáhají při psaní svého kódu. Jak již bylo řečeno, tento jazyk je navržen tak, aby podporoval jak začátečníky, tak pokročilé programátory, a proto je snadné tyto prvky pochopit, zapamatovat si je, použít a kombinovat. Níže je výčet některých vestavěných prvků, které byly také použity během projektu. (49)

- *print* – vestavěná funkce pro výpis informací proměnné, řetězce a podobně. Používá se obvykle pro kontroly a výpisy výsledků typicky přímo na obrazovku uživatele.
- *append* – metoda seznamu, která přidává prvek na konec seznamu.
- *open* – používá se pro otevření souboru a vrátí souborový objekt, který umožňuje číst nebo psát do souboru. Po skončení práce se souborem je potřeba použít *close*, aby nedošlo ke ztrátě dat nebo úniku paměti.
- *with open* – podobně jako *open* je *with open* používáno pro práci se soubory. Tato syntaxe ale navíc zajišťuje správné uzavření souboru po dokončení operací s ním kontextovému manažerovi *with*.
- *def* – *def* je klíčové slovo používané pro definování vlastních funkcí v Pythonu. Za *def* následuje název funkce a v závorce parametry, které funkce přijímá.
- *return* – je klíčové slovo používané uvnitř definic funkce k ukončení funkce a vrácení hodnoty volajícímu
- *for* – klíčové slovo pro vytvoření cyklu v Pythonu, který se používá k projití prvků jakéhokoliv iterovatelného objektu (např. seznam, n-tice, slovník, řetězec)
- *if* – klíčové slovo, které umožňuje vykonávat kód na základě nějaké podmínky. Společně s *if* se také často používá klíčové slovo *else*.
- *elif* – klíčové slovo, které je zkratkou pro *else if*, umožňuje rozšířit kód o další podmínky.
- *else* – klíčové slovo pro definování bloku kódu, který má být proveden, jestliže předcházející podmínka nebo podmínky nejsou splněny.

- *match-case* – je speciální syntaxe, která umožňuje provádět porovnávání podmínek podobně jako kombinace *if*, *elif* a *else*, ale může být v některých případech mnohem čitelnější.
- *f-string* – je formulovatelný řetězec, umožňuje tak vložení hodnot proměnných přímo do řetězce
- *strip* – vestavěná funkce pro, která slouží k odstranění bílých znaků (např. mezery, tabulátory, nové řádky) z obou konců řetězce
- *split* – další vestavěná funkce pro řetězce. Používá se k rozdělení řetězce na seznam podřetězců na základě oddělovače
- *import* – klíčové slovo pro načtení modulu nebo knihovny, aby bylo možné využívat její funkce, třídy nebo jiné objekty v rámci skriptu (49)

1.11.2 Moduly a knihovny

V Pythonu se běžně využívají knihovny, které jsou poskládané z modulů, což je soubor, který obsahuje funkce, třídy a proměnné. Jejich uložení do samostatných modulů umožňuje opětovné použití v další programech. Modul se obvykle skládá z Python souboru s příponou *.py* a jedním z takových použitých modulů je například *csv*. (49)

Modul *csv* umožňuje práci s CSV soubory uvnitř skriptu, jako je čtení a zápis pomocí *csv.DictReader* a *csv.DictWriter*, metody jako je *writerow* pro zápis jednoho řádku a *writerows* pro zápis více řádků najednou. (49)

Dalším využitým modulem je modul *copy*, umožňující kopírování objektů. Důležitou funkcí je hloubková kopie (*deepcopy*), protože na rozdíl od povrchové kopie (*shallowcopy*), která kopíruje pouze referenci na objekt, kopíruje všechny objekty obsažené v kopírovaném objektu, včetně vnořených objektů. Vytváří tak zcela nový objekt a zaručuje, že nedojde ke změně originálního objektu při práci s kopií. (49)

V neposlední řadě je zde knihovna requests poskytující prostředky pro odesílání HTTP/S požadavků jako je *POST*, *GET* a další ve formátu JSON. V kombinaci s REST API je možné tímto způsobem automatizovat konfiguraci například u Barracuda CloudGen firewallu. Součástí knihovny requests je také modul HTTPBasicAuth použit pro snadné implementování základního autentizace v HTTP/S požadavcích. (49)

2 ANALÝZA SOUČASNÉHO STAVU

2.1 Základní charakteristika společnosti

COMGUARD a.s. je akciovou společností podnikající v oblasti IT bezpečnosti především pro B2B trh. Společnost vznikla v roce 2006 a její základní kapitál činí 3 520 000 Kč rozvržen na 22 000 akcií o jmenovité hodnotě 160 Kč/akcii. Před rokem 2015 byla právní forma společnosti společnost s ručením omezeným. Hlavním předmětem podnikání je výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona, s obory činnosti:

- Zprostředkování obchodu a služeb
- Velkoobchod a maloobchod
- Poskytování software, poradenství v oblasti informačních technologií, zpracování dat, hostingové a související činnosti a webové portály
- Pronájem a půjčování věcí movitých
- Poradenská a konzultační činnost, zpracování odborných studií a posudků
- Výzkum a vývoj v oblasti přírodních a technických věd nebo společenských věd
- Testování, měření, analýzy a kontroly
- Mimoškolní výchova a vzdělávání, pořádání kurzů, školení, včetně lektorské činnosti (51; 52; 53)

Vizi společnosti je udržování a rozvoj uznávané pozice B2B společnosti na českém a slovenském trhu v oblasti IT bezpečnosti. Toho dosahuje poskytováním špičkových technologií a ochrany digitálního prostoru díky aktivní spolupráci s mezinárodními technologickými lídry, aby mohla neustále dodávat inovativní a efektivní řešení. COMGUARD a.s. aktivně navazuje nové vztahy s odborníky a rozšiřuje své portfolio o nové produkty a technologie, přičemž klade velký důraz na kvalitu poskytovaných služeb a spokojenost klientů. (51; 52)

V současné době nabízí firma komplexní portfolio produktů a služeb zaměřených na ochranu a bezpečnost informačních technologií pro své partnery a jejich koncové zákazníky včetně zabezpečení cloudových služeb, datových center, interních sítí,

webových aplikací, vzdáleného přístupu, uživatelských stanic a mobilních zařízení, přes ochranu dat a komunikace, řízení rizik a hrozeb, poskytování pokročilých technologií až po odborné školení. (51; 52)

Mezi hlavní partnery patří společnost SOPHOS, která je dlouholetý leader v oblasti Endpoint Security především pro malé a střední společnosti, Trellix se svou platformou ePolicy Orchestrator umožňující pokročilou centrální správu datových center, interních sítí a ochranu citlivých dat a další partneři a dodavatelé jako jsou Skyhigh Security, GYTPOL, Ekran, Whalebone, Rapid7, Wallix, LogRhythm, Barracuda a SecurEnvoy poskytující řadu produktů a služeb v oblasti kybernetické bezpečnosti. (51; 52)



Obrázek 22: Logo společnosti COMGUARD a.s. (Zdroj: (54))

Tabulka 9: Základní charakteristika společnosti (Zdroj: Vlastní zpracování dle: (52))

Základní charakteristika společnosti	
Název společnosti	COMGUARD a.s.
Právní forma	Akciová společnost
Sídlo firmy	Sochorova 3209/38, Žabovřesky, 616 00 Brno
IČO	04305426

Orgány společnosti	Předseda správní rady: - RNDr. Karel Taft Členové správní rady: - Ing. Jan Dymáček - Ing. Karel Klumpner
Předmět podnikání	Výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona, provozovaná v oborech:
	- zprostředkování obchodu a služeb
	- velkoobchod a maloobchod
	- poskytování software, poradenství v oblasti informačních technologií, zpracování dat, hostingové a související činnosti a webové portály
	- poradenská a konzultační činnost, zpracování odborných studií a posudků
	- výzkum a vývoj v oblasti přírodních a technických věd nebo společenských věd
	- testování, měření, analýzy a kontroly
	- mimoškolní výchova a vzdělávání, pořádání kurzů, školení, včetně lektorské činnosti

2.1.1 Organizační struktura

Organizační struktura společnosti uvedena v příloze č. 1 je rozdělena do několika klíčových oddělení, na vrcholu hierarchie stojí generální ředitel (CEO), který má celkovou zodpovědnost za strategické vedení společnosti. Pod něj spadají tři hlavní oddělení: finanční oddělení, které se zabývá finančním řízením a účetnictvím; obchodní oddělení, které řídí prodej produktů a vztahy se zákazníky; a technické oddělení, specializující se především na kybernetickou bezpečnost. Mimo tato oddělení je zde také

marketingový manažer, který přímo spolupracuje s výše uvedenými odděleními a CEO, aby zajistil optimální marketingovou strategii pro společnost. (55)

Technické oddělení hraje klíčovou roli ve společnosti pro řešenou problematiku kvůli svým znalostem v oblasti kybernetické bezpečnosti. Vedoucí oddělení má finální rozhodnutí u všech částí projektu a podává hlášení přímo CEO společnosti. Do oddělení spadají další pracovníci a vedoucí, ze kterých bude následně vybrán vedoucí projektu a odborný konzultant. Ostatní pracovníci se navíc budou také podílet na některých dílčích částech a budou poskytovat odbornou pomoc v rámci projektu migrace firewallových pravidel na nový firewall. (55)

2.2 Důvod migrace

Společnost řadu let používala pro ochranu své interní sítě firewall Forcepoint Sidewinder, což je robustní next-generation firewall pro velké a střední podniky. Avšak společnost Forcepoint postupně přešla k větší propagaci svého nového produktu, Forcepoint NGFW (Next Generation Firewall). Tento produkt měl postupně nahradit Forcepoint Sidewinder, což se také nakonec stalo, když jej prohlásila za *End of Life (EOL)* produkt, tedy produkt s ukončenou podporou, což znamená že již nebude dostávat žádné nové aktualizace a technická podpora také postupně přestane být dostupná. Tato situace otevřela velkou bezpečnostní zranitelnost pro potenciální útočníky, proto musela spousta organizací přijít s novým řešením pro ochranu své sítě, aby nedošlo k jejímu zneužití. (56)

Společnost COMGUARD a.s. po výběrovém řízení učinila rozhodnutí pro přejítí na Barracuda CloudGen Firewall (CGFW), což je moderní bezpečnostní řešení pro ochranu rozsáhlých podnikových sítí. Konkrétní model je Barracuda CloudGen Firewall F180. Barracuda CGFW podobně jako jiné next-generation firewally nabízí širokou škálu pokročilých funkcí jako je Advance Threat Protection (ATP), Data Loss Prevention (DLP), Application Control, SSL Encryption a mnohé další. Mezi hlavní výhody Barracuda CGFW patří možnost snadné integrace s cloudovým prostředím a rozsáhlé možnosti pro správu VPN. (21; 57)

Ze stávajícího firewallu (Forcepoint Sidewinder) je potřeba migrovat politiky a síťové objekty. Samotný proces migrace vyžaduje pečlivé plánování a přípravu, aby byla

pravidla správně přenesena na nový firewall (Barracuda CloudGen Firewall). Tato příprava zahrnuje pověření IT pracovníka a jeho následné školení s certifikací, neustálou komunikaci a konzultaci, studování struktury pravidel a jiných nastavení obou firewallů, řízení rizik a volba vhodných technologií pro automatizaci a optimalizaci migrace.

Tato situace ale na druhou stranu dává společnosti možnost revize, úpravy a pročištění firewallových politik, protože zde mohly vzniknout za roky používání duplicity, obsoletní pravidla nebo prostě nekonzistentnost ve stylu tvorby těchto pravidel, jelikož se ve společnosti vystřídal více správců firemního firewallu.



Obrázek 23: Barracuda CloudGen Firewall (Zdroj: (58))

2.2.1 Struktura uvnitř Forcepoint Sidewinder

Struktura objektů

Na obrázku níže je uveden příklad anonymizovaných síťových objektů exportovaných z Forcepoint Sidewinder. Tyto objekty jsou exportované do souboru CSV a bude s nimi pracovat program v jazyce Python, který je bude muset přetvořit na syntaxi, které rozumí Barracuda CloudGen Firewall.

Seznam všech možných hodnot síťových objektů ve Forcepoint Sidewinder:

- *table* – typ síťového objektu
- *name* – název síťového objektu
- *ipaddr* – IP adresa, jestliže síťový objekt nějakou má
- *subnet* – hodnota podsítě, jestliže se jedná o objekt typu subnet
- *domain* – specifikuje doménu u objektu typu domain (v rámci migrace takový nebyl)
- *host* – název hostitele, jestliže se jedná o objekt typu host
- *dns_mode* – pouze u objektů typu dns
- *ttl* – maximální doba existence paketu v rámci host objektů
- *members* – seznam členů (referencovaných objektů) v rámci objektu typu netgroup
- *bits* – bitová délka subnetu, pouze u typu subnet

- *begin* – začátek IP rozsahu, pouze u typu iprange
- *end* – konec IP rozsahu, pouze u typu iprange
- *description* – popis objektu
- *last_changed_by* – čas, datum a administrátor, který provedl poslední změnu objektu (59)

table	name	ipaddr	subnet	domain	host	dns_mode	ttd	members	bits	begin	end	description	last_changed_by
host	seznam.cz				www.seznam.cz	dns	0					"	'cg14 on Thu Dec 14 14:58:14 2023'
ipaddr	admin1	192.168.1.201										"	'cg8 on Mon Apr 22 16:51:27 2013'
ipaddr	admin2	192.168.1.221										"	'cg8 on Mon Apr 22 16:51:27 2013'
ipaddr	Google-DNS	8.8.8.8										"	'cg8 on Mon Apr 22 16:51:27 2013'
subnet	google-173.194.0.0		173.194.0.0						16			"	'cg14 on Fri Jan 19 11:09:22 2018'
subnet	google-172.217.32.0		172.217.32.0						20			"	'cg14 on Fri Jan 19 11:05:15 2018'
iprange	pc-net									10.10.100.10	10.10.100.20	"	'cg5 on Tue Aug 16 06:20:31 2011'
iprange	barracuda_FWs									10.10.11.41	10.10.11.46	"	'cg15 on Wed Mar 1 19:58:04 2017'
netgroup	admins							ipaddr:192.168.1.201,ipaddr:192.168.1.221'				"	'cg15 on Mon May 7 11:25:39 2018'

Obrázek 24: Struktura objektů uvnitř Forcepoint Sidewinder (Zdroj: Vlastní zpracování)

Struktura pravidel

Obdobným způsobem bude muset být řešena struktura pravidel v rámci Forcepoint Sidewinder. Tyto pravidla sice nebudou migrována pomocí skriptu, je však důležité pochopit jejich strukturu. Níže je opět uveden anonymizovaný příklad.

Seznam všech možných hodnot pravidel ve Forcepoint Sidewinder:

- *ID* – identifikační číslo pravidla
- *Name* – název pravidla

- *PORT* – protokol a port odpovídající přenosu
- *Source* – zdroj, který má být zpracován pravidlem
- *Destination* – cíl, který má být zpracován pravidlem
- *Source NAT* – překlad odchozích IP adres
- *Destination NAT* – překlad veřejných IP adres na privátní
- *Desc* – popis pravidla (59)

ID	Name	PORT	Source	Destination	Source NAT	Destination NAT	Desc
1	allow-admins-2-external	ANY	admins	Any V4	localhost		Přístup adminů do internetu
2	allow-pcnet-2-ak1	HTTPS (TCP 443)	pc-net (IP range)	IP_10.20.20.30			Přístup k firemnímu zařízení pro zaměstnance
3	allow-ntp-from-dmz-2-cesnet-external	NTP (UDP 123)	dmz	tik.cesnet.cz, tak.cesnet.cz	localhost		
4	allow-admins-2-BCGFW	TCP 8080	admins	barracuda_FWs (iprange)			Přístup adminů k Barracuda CloudGen Firewallu
5	allow-smtp-from-external-2-mailgw-dmz	SMTP (TCP 25)	any	MX (83.22.64.24)		antispam-gw 172.16.1.10	Povolení mailu z externí sítě na antispam gateway
6	allow-webservices-from-external-2-webserver-dmz	TCP 8080, TCP 8010	any	IP_83.22.64.25		dmz ip adresa 172.16.1.54	Povolení webserveru z externí sítě a přesměrování do DMZ
7	allow-webservices-from-external-2-webserver2-dmz	TCP 8090	any	IP_83.22.64.25		dmz ip adresa 172.16.1.55:8080	Povolení webserveru2 z externí sítě a provedení přesměrování na port 8080
8	allow-restapi-from-admins-2-bcgfw	TCP 8080, TCP 8443	admins	barracuda_FW1			Povolení komunikace restapi od adminů na firewall
9	allow-rdp-admin1-2-servernet	RDP (TCP 3389)	admin1	servernet (10.20.30.0/24)			Povolení přístupu admina 1 na firemní server
10	allow-dns-from-servernet-2-googleDNS-external	DNS (UDP 53)	servernet	Google-DNS	localhost		

Obrázek 25: Struktura pravidel uvnitř Forcepoint Sidewinder (Zdroj: Vlastní zpracování)

2.2.2 Zadání projektu migrace

Tento projekt bude probíhat v několika částech. V první části bude potřeba, aby technický pracovník pověřený migrací prošel certifikačním kurzem Barracuda CloudGen Firewall, aby porozuměl jeho prostředí a uměl s ním pracovat. Tento kurz je plně hrazen společností a po jeho splnění je možné přejít k práci na projektu. Pro snazší osvojení bude zapůjčen Barracuda CGFW F18, na kterém bude možné provádět testování znalostí.

Dále se z hlediska bezpečnosti a využití lidských zdrojů provede analýza rizik a časová analýza s RACI maticí. Je tak jednodušší zamezit případným bezpečnostním incidentům a korigovat harmonogram všech pracovníků s dalšími činnostmi.

Následně přijde část samotné migrace, která bude probíhat ve dvou hlavních etapách. V první etapě budou migrovány síťové objekty za pomoci dostupných technologií. Současně budou pročištěna pravidla o ty, která již nejsou používána a nachystají se tak pro následnou migraci, která bude probíhat pro větší kontrolu ručně po jednotlivých částech. Po každé etapě bude následovat krátké shrnutí její úspěšnosti a případná korekce chyb.

Po úspěšné migraci bude potřeba přepojit firewally a monitorovat výsledky. V neposlední řadě proběhne finanční zhodnocení zaměřující se na TCO a jeho jednotlivé položky během doby životnosti firewallu. Toto zhodnocení by mělo sloužit společnosti pro plánování rozpočtů do budoucích let a na podobné projekty.

3 VLASTNÍ NÁVRHY ŘEŠENÍ

3.1 Rizika spojená s migrací

3.1.1 Identifikace rizik

Před, během, ale i po samotné migraci se mohou vyskytnout rizika spojená s tímto projektem. Pro účely projektu byla primárně identifikována tato rizika v tabulce, která však mohou být rozšířena o rizika další. Jednotlivá rizika jsou navíc doplněna o scénář, který značí možný dopad na projekt nebo společnost.

Je nutné dodat, že výčet rizik nikdy nebude kompletní, avšak pro společnost je důležité identifikovat ta stěžejní, na které se může zaměřit a ovlivnit je nějakým způsobem za přijatelnou cenu, aby se jejich pravděpodobnost výskytu či dopad snížil.

Tabulka 10: Identifikovaná rizika migrace (Zdroj: Vlastní zpracování)

ID	Hrozba	Scénář
1	Dojde k zničení nebo ztrátě migrovaných dat	Ztráta dat potřebných k provozu nového FW
2	Ztracení přístupu k firewallu	Dojde ke změně nastavení a nastane ztráta možnosti konfigurovat a řídit FW
3	Únik dat z migrovaných pravidel nebo objektů	Velmi citlivá data mohou být zneužita k útoku na vnitřní síť zvenčí díky znalosti její struktury a jejich slabých míst
4	Neoprávněný přístup na FW	FW bude zneužit neoprávněnou osobou pro škodlivé činnosti a může vzniknout možná hrozba pro společnost
5	Vynechání objektů	Pravidla nebudou moct být plně vytvořena a mohlo by dojít k chybám, které vytvoří zranitelnosti
6	Vynechání pravidel	Pravidla nebudou úplná a některý provoz nebude moct projít

7	Poškození zařízení	Zařízení bude muset být odesláno na opravu/náhradu a projekt se prodlouží
8	Odcizení cvičného zařízení	Vznikne finanční ztráta a v horším případě dojde i k úniku citlivých informací
9	Nezabezpečení vstupu do serverovny	Může být narušena bezpečnost FW, ale i jiných zařízení
10	Pověření migrací nedostatečně kvalifikovanou osobou	Dojde k vytvoření chyb při konfiguraci nebo migraci objektů a pravidel, což může vést k vytvoření zranitelností

3.1.2 Kvantifikace rizik

Rizika byla následně ohodnocena v rámci pravděpodobnosti výskytu a velikosti dopadu v případě, že dojde k realizaci rizika. Z těchto dvou hodnot se následně může určit hodnota rizika pomocí matice. Tabulka pro hodnocení pravděpodobnosti výskytu, velikosti dopadu a matice jsou v tabulkách níže.

Tabulka 11: Verbální ohodnocení pravděpodobnosti výskytu rizika (Zdroj: Vlastní zpracování dle: (37))

Verbální ohodnocení pravděpodobnosti výskytu rizika	
VP – Velká pravděpodobnost	Nad 66 %
SP – Střední pravděpodobnost	33–66 %
MP – Malá pravděpodobnost	33 % a méně

Tabulka 12: Verbální vyjádření dopadu rizika (Zdroj: Vlastní zpracování dle: (37))

Verbální vyjádření dopadu rizika	
VD – Velký dopad	• Nad 20 % hodnoty projektu • Ohrožení cíle projektu • Ohrožení termínu projektu
SD – Střední dopad	• 1 % - 20 % z hodnoty projektu • Zpoždění některých částí projektu
MD – Malý dopad	• Do 1 % hodnoty projektu • Menší úpravy v plánu projektu

Tabulka 13: Matice výpočtu hodnoty rizika (Zdroj: Vlastní zpracování dle: (37))

		Pravděpodobnost		
		VP	SP	MP
Dopad	VD	VHR	VHR	SHR
	SD	VHR	SHR	NHR
	MD	SHR	NHR	NHR

Po identifikaci rizik byly použity tabulky pro ohodnocení rizik z hlediska pravděpodobnosti výskytu, velikosti dopadu rizika a následně byla vypočítána hodnota rizika. Primární rizika, na které by měla být zaměřena pozornost při výběru opatření jsou rizika s vysokou hodnotou, což v tomto případě je riziko č. 6 Vynechání pravidel a č. 10 Pověření migrací nedostatečně kvalifikovanou osobou. Tato rizika znamenají pro projekt a společnost největší hrozbu.

Tabulka 14: Ohodnocení rizik migrace (Zdroj: Vlastní zpracování)

ID	Hrozba	Pravděpo- dobnost rizika	Dopad rizika	Hodnota rizika
1	Dojde k zničení nebo ztrátě migrovaných dat	SP	ND	NHR
2	Ztracení přístupu	SP	SD	SHR
3	Únik dat	SP	ND	NHR
4	Neoprávněný přístup na FW	NP	VD	SHR
5	Vynechání objektů	SP	SD	SHR
6	Vynechání pravidel	SP	VD	VHR
7	Poškození zařízení	NP	VD	SHR
8	Odcizení cvičného zařízení	NP	VD	SHR
9	Nezabezpečení vstupu do serverovny	NP	VD	SHR
10	Pověření migrací nedostatečně kvalifikovanou osobou	SP	VD	VHR

3.1.3 Návrhy opatření na snížení rizik

Na identifikovaná rizika bylo následně vytvořeno opatření pro snížení dopadu nebo pravděpodobnosti výskytu na akceptovatelnou úroveň. Po aplikování opatření by měla být hodnota rizika snížena na novou hodnotu rizika uvedenou v tabulce. Rizika by však měla být stále sledována, protože může dojít ke změně hodnoty rizika, ať už z důvodu zvýšení pravděpodobnosti výskytu nebo zvýšení dopadu nebo pokud se vyskytnou nová rizika.

Tabulka 15: Návrhy na opatření rizik migrace a nová hodnota rizika (Zdroj: Vlastní zpracování)

ID	Návrh na opatření	Nová hodnota rizika
1	Vytvoření záloh na jiných zařízeních, možnost opětovného exportování ze starého FW a následné očištění	NHR
2	Pravidelné zálohování konfigurace FW pro případnou obnovu zařízení. Důkladná kontrola migrovaných objektů, pravidel a jiných nastavení.	NHR
3	Ukládání dat na bezpečných zařízeních, best practices práce s citlivými daty, anonymizace dat	NHR
4	Vytvoření přístupů pro oprávněné uživatele a ochrana těchto přístupů, fyzické zabezpečení přístupu k FW v uzamčené serverovně	NHR
5	Testovací migrace a kontrola úplnosti exportovaných a importovaných objektů	NHR
6	Testovací migrace a kontrola úplnosti exportovaných a importovaných pravidel	SHR
7	Poučení o bezpečné manipulaci se zařízením, minimalizace přesun, bezpečné umístění a omezení přístupu na oprávněné osoby, vytvoření zápůjční smlouvy	NHR
8	Vytvoření zápůjční smlouvy, poučení o bezpečné úschově a přenosu zařízení, minimalizace přesun, bezpečné umístění a omezení přístupu na oprávněné osoby	NHR
9	Minimalizace oprávněných osob pro vstup do serverovny na nejpotřebnější, kontrola uzamčení vstupu	NHR
10	Školení pověřených pracovníků, určení řídicího a kvalifikovaného pracovníka, dohled nad prací pověřených pracovníků	SHR

3.1.4 Celkové posouzení rizik

Na základě aplikovaných opatření se povedlo snížit většinu rizik na NHR. Výjimkou jsou dvě rizika, Vynechání pravidel a Pověření migrací nedostatečně kvalifikovanou osobou se SHR. Společnost je ochotna tato rizika podstoupit za stanovených podmínek.

Aby nedošlo k vynechání pravidel bude kladen velký důraz na kontrolu všech pravidel při jejich pročišťování a vyřazování a také při jejich následné migraci. Jakmile budou pravidla migrována a proběhne i samotné fyzické přepojení firewallů bude společnost monitorovat síť pro případné bezpečnostní incidenty.

V rámci druhého rizika poskytne společnost oficiální kurz s certifikačními testy přímo od společnosti Barracuda, aby zaručila kvalitu dodaných výukových materiálů a podpory. Avšak stěžejní podmínkou je poctivá příprava technického pracovníka, který sám zodpovídá za plnění svých povinností, a tedy i řádného samostudia a plnění kurzů.

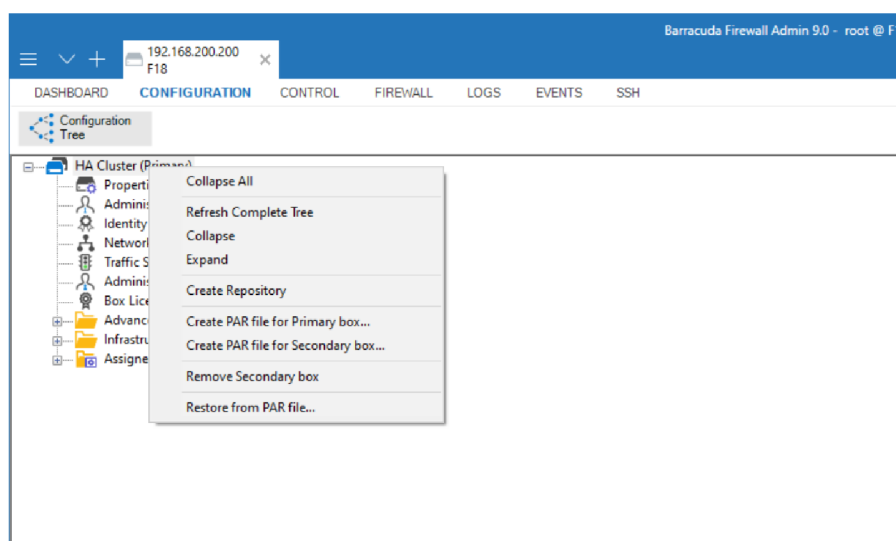
V neposlední řadě je nutno podotknout, že žádné z rizik nedosahuje VHR, což značí účinnost nasazených opatření. Společnost tak může mít větší jistotu, že nemusí podstupovat potencionálně velmi závažná rizika, která by mohla mít značně negativní dopad na celý projekt nebo společnost jako celek. Jak již ale bylo zmíněno rizika spojená s projektem migrace budou nadále sledována během vývoje celého projektu a budou řízena i nově vyskytlá rizika, aby bylo zamezeno případnému incidentu. V příloze č. 2 můžeme vidět celou analýzu rizik RIPRAN.

3.2 Popis vyskytnutého rizika a využití nápravného opatření

Během migrace objektů na nový firewall při jednom z prvních pokusů došlo k incidentu při čemž byl ztracen přístup k firewallu, tak jako je popsáno v riziku č. 2. Při tomto pokusu totiž byla vytvořena pravidla pomocí skriptu na testovacím firewallu (cvičném zařízení) a pro ušetření času bylo v plánu použít funkci Barracuda CloudGen Firewallu, která umožňuje zkopírování pravidel do schránky a jejich následné vložení na nový firewall, což se používá například při vytváření dalších firewallů v clusteru. Ušetřilo by se tak tolik času, kolik trvalo jedno projití objektů pomocí skriptu a jejich vytvoření.

Problémem bylo, že během kopírování se kopírovali i dynamické objekty, které neměly být na cílovém firewallu měněny. Mezi dynamické objekty totiž mimo jiné patří i Primary Management IP, Management IP a HA Management IP, což jsou objekty, které obsahují IP adresy, které mohou po připojení provádět změny uvnitř primárního, a i sekundárního firewallu. Tyto objekty jsou sbaleny v seznamu *DYNAMIC* a při jejich kopírování není správce nijak upozorněn, že jsou kopírované také.

Po zkopírování objektů a aktivaci po krátké chvíli došlo k odpojení od firewallu, načež došlo k troubleshootingu a bylo zjištěno, že pravděpodobně došlo k výše zmíněné situaci. Bylo proto nasazeno opatření obnovy z pravidelně zálohovaných konfiguračních souborů. Tyto soubory neboli *PAR files* je možné vytvořit v kartě *CONFIGURATION* po kliknutí pravým tlačítkem na box (viz obrázek níže) nebo v tomto případě cluster boxů, protože se jedná o primární a sekundární FW.

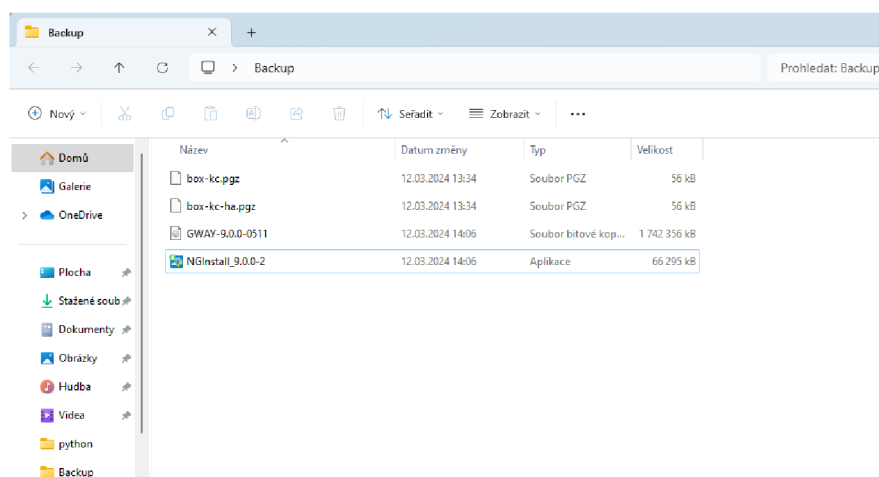


Obrázek 26: Vytvoření zálohy konfigurace firewallu (Zdroj: Vlastní zpracování)

3.2.1 Obnovení primárního firewallu

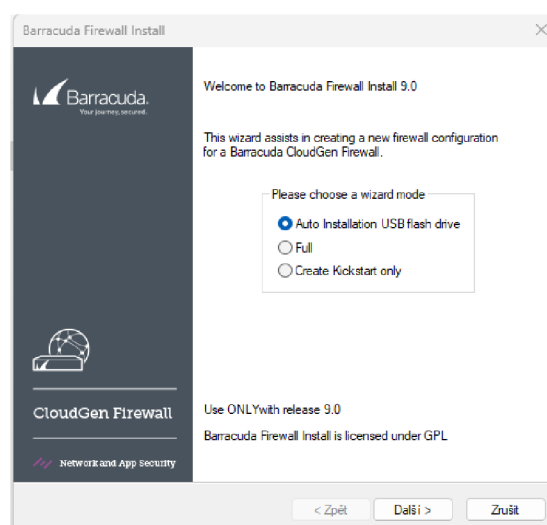
Pro samotnou obnovu zařízení je potřeba vytvořit instalační USB disk. Tento disk musí mít alespoň 2 GB, a ne víc než 32 GB, dále je potřeba stáhnout instalační soubor pro Barracuda Firewall a *CloudGen Firewall ISO image*. Níže je popsán postup vytvoření instalačního USB disku pro primární FW.

Na obrázku je možné vidět stažené soubory potřebné k vytvoření instalačního USB pro primární i sekundární FW.



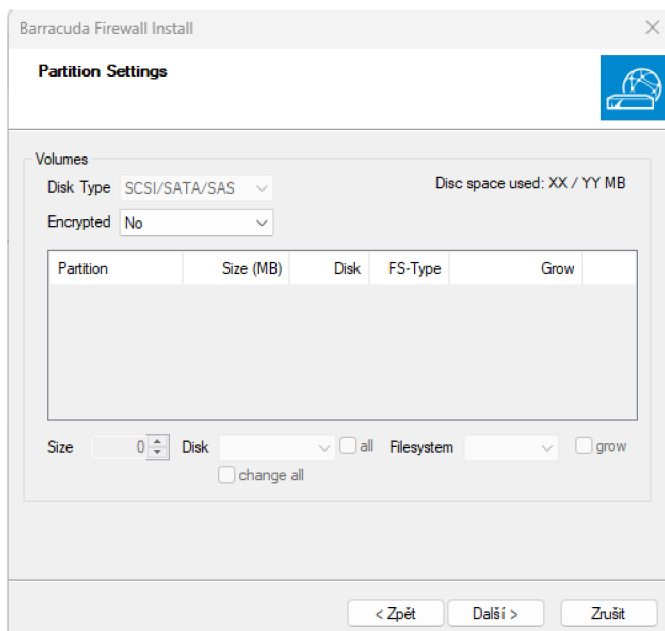
Obrázek 27: Soubory potřebné pro obnovení firewallu (Zdroj: Vlastní zpracování)

Pro vytvoření instalačního USB je potřeba prve USB vložit do zařízení, dále se musí spustit soubor *NGInstall_9.0.0-2* s administrátorskými právy. Otevře se instalační okno a zvolíme *Auto Installation USB flash drive*. Klikneme na *Další*.



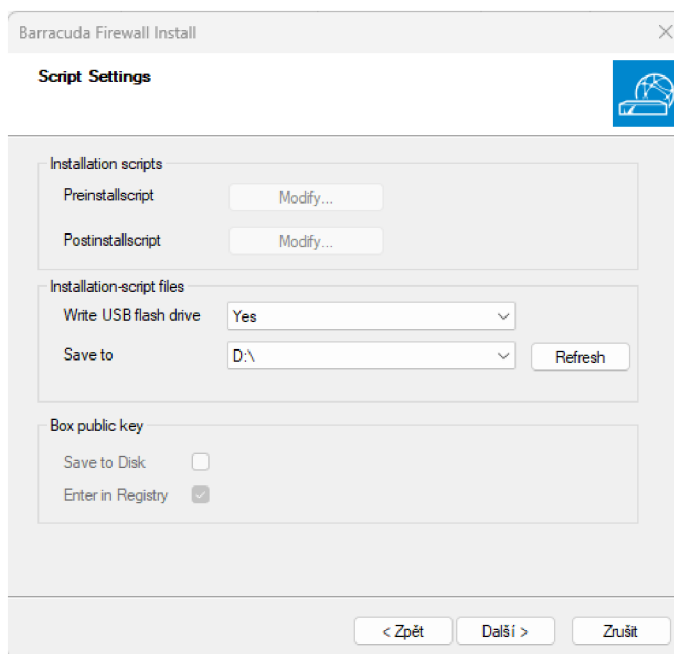
Obrázek 28: Tvorba instalačního USB – Auto Installation (Zdroj: Vlastní zpracování)

Partition settings na další stránce instalátoru je možné ignorovat, proto klikneme na *Další*.



Obrázek 29: Tvorba instalačního USB – Partition Settings (Zdroj: Vlastní zpracování)

Zde zvolíme z listu *Save to disk*, ze kterého se má stát instalační disk, což je v tomto případě *disk D*. Zároveň je potřeba zvolit *Yes* u *Write USB flash drive*, jestliže tato možnost není vybrána a klikneme na *Další*.

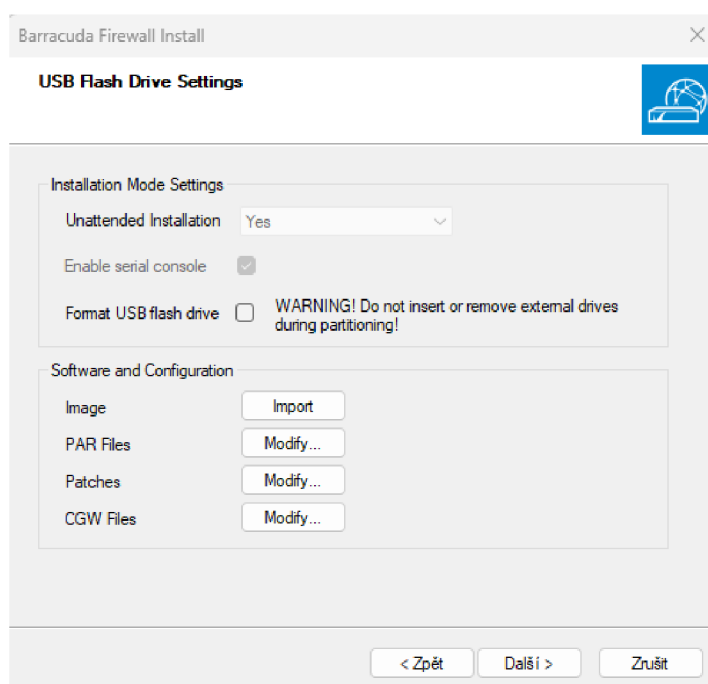


Obrázek 30: Tvorba instalačního USB - Script Settings (Zdroj: Vlastní zpracování)

Na další stránce můžeme zvolit *Format USB flash drive*, což před samotnou instalací naformátuje USB, což vymaže všechny soubory, které jsou na něm uloženy. Tento postup je doporučený, protože zamezí výskytu chyb při instalaci, je proto vhodné předtím všechny důležité soubory zálohovat mimo USB.

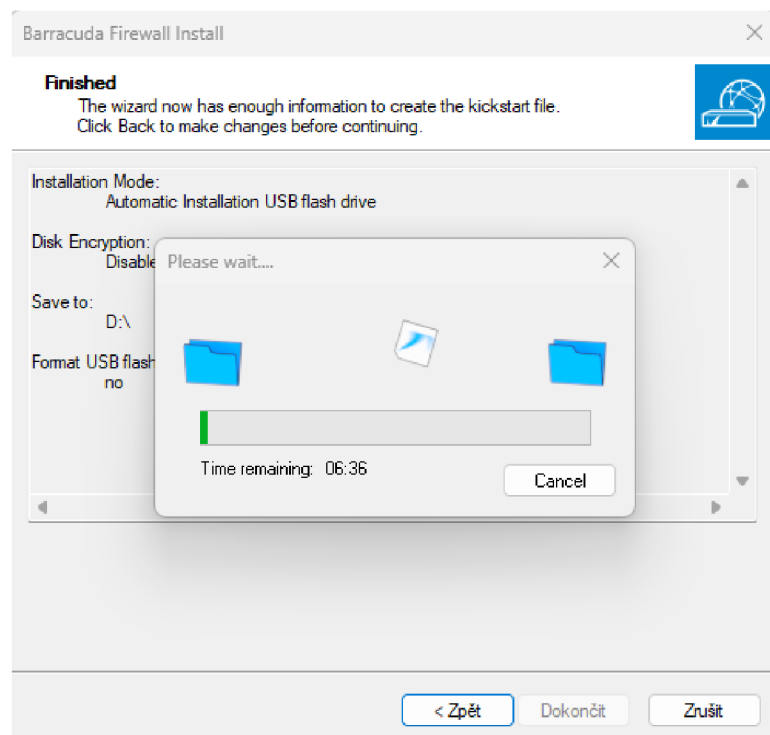
V první řadě importujeme *ISO image*, který jsme si dopředu stáhli. Jednoduše ze složky *Backup* vybereme soubor *GWAY-9.0.0-0511*. Obdobným způsobem vložíme pomocí tlačítka *Modify* PAR soubor s konfigurací primárního FW, tedy zvolíme *box-kc.pgz* ve složce *Backup*.

Nepovinně se mohou s podobným postupem nahrát soubory *Patches* a *CGW Files*, což předinstaluje některé hotfixy a patche na FW. Pro aktuální situaci to není potřeba, proto zvolíme *Další*.



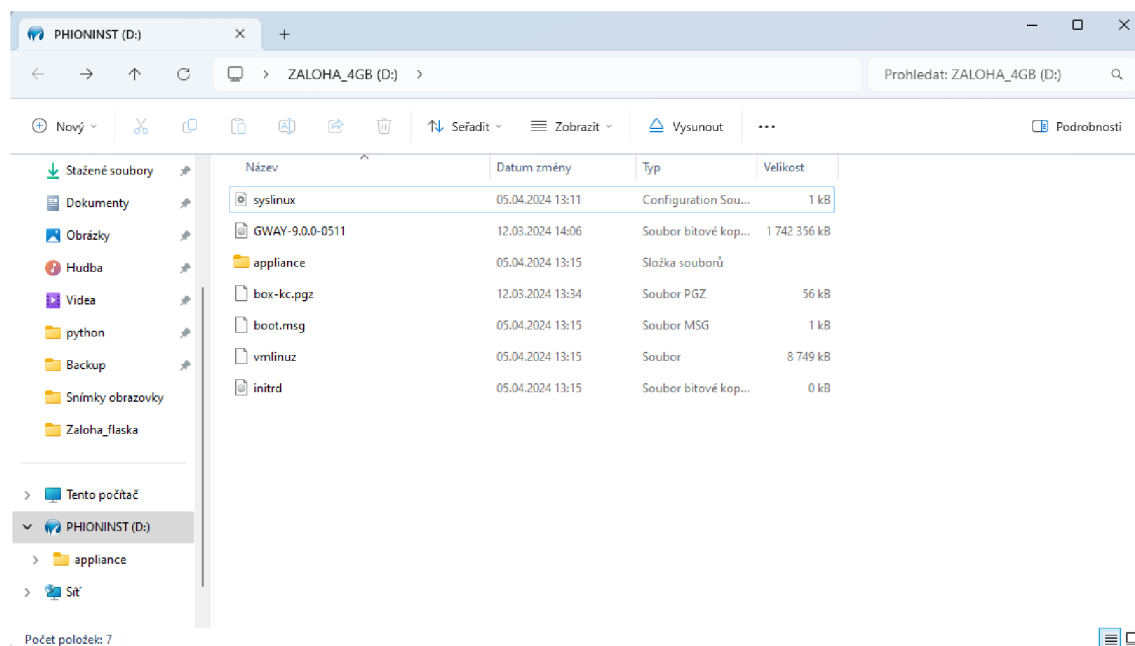
Obrázek 31: Tvorba instalačního USB - USB Flash Drive Settings (Zdroj: Vlastní zpracování)

Další stránka obsahuje shrnutí zvolených změn, provedeme finální kontrolu a zvolíme *Dokončit*. Nyní začne proces vytváření instalačního USB, který může zabrat několik minut. Je důležité během procesu neodstraňovat USB disk ze zařízení a ani zařízení nevypínat. Po úspěšném vytvoření se zobrazí okno s upozorněním, že vše proběhlo, jak má a USB je připraveno k instalaci. Můžeme jej zavřít tlačítkem *OK*.



Obrázek 32: Tvorba instalačního USB - Vytváření (Zdroj: Vlastní zpracování)

Instalační USB vypadá následně takto. Jsou v něm instalační soubory spolu s *ISO image* a konfigurací *box-kc*, což je konfigurace primárního FW. Vytvoření instalačního USB pro sekundární box bude probíhat stejným způsobem, akorát se při modifikaci PAR souborů vybere soubor *box-kc-ha.pgz*, což je zálohovaný PAR soubor sekundárního FW.

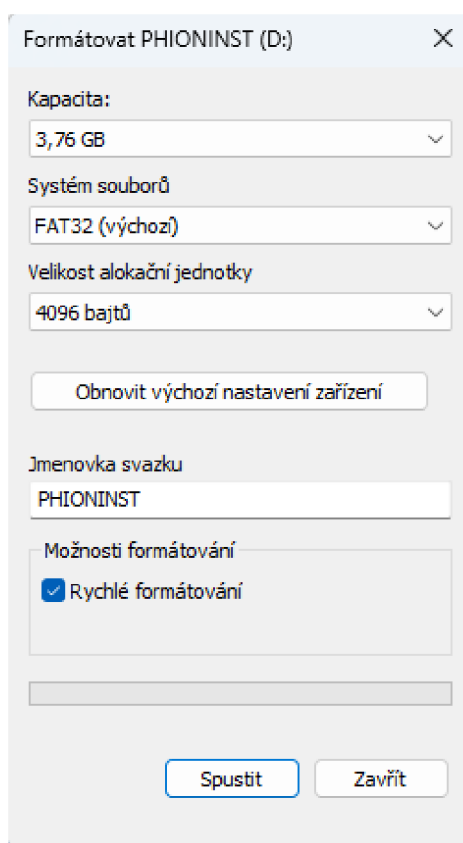


Obrázek 33: Pohled do instalačního USB (Zdroj: Vlastní zpracování)

Instalace firewallu probíhá následovně:

1. Instalační USB vložíme do zařízení (firewallu) a restartujeme ho pomocí tlačítka na zadní straně zařízení.
2. Jakmile je instalační proces hotov, zazní dlouhé zapípání. Tato fáze může zabrat několik minut.
3. Nyní je potřeba USB vyjmout a opět restartovat zařízení.
4. Jestliže všechny předchozí kroky proběhly v pořádku, mělo by být možné se opět přihlásit k FW pomocí Firewall Admin.

V rámci *best practices* je po úspěšném obnovení každého z firewallů nutné provést formátování USB disku, tak aby byla smazána data, která mohou být případně zneužita, jako například konfigurace firewallů. Naformátování provedeme pravým kliknutím na USB disk, zvolíme *Formátovat...*, Rychlé formátování a spustíme proces. Disk se po pár vteřinách formátuje, takže již nebude obsahovat citlivé soubory.

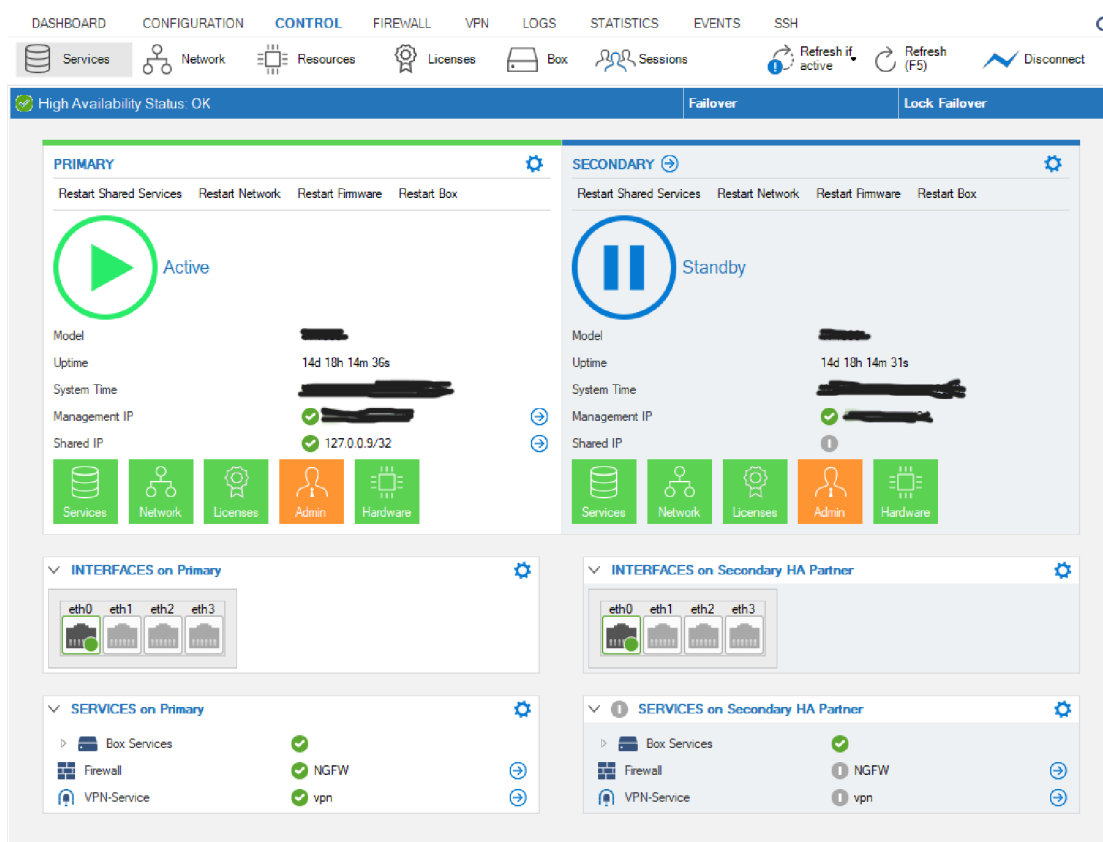


Obrázek 34: Formátování instalačního USB (Zdroj: Vlastní zpracování)

3.2.2 Obnovení sekundárního firewallu

Obnovení sekundárního FW probíhá obdobným způsobem, je ale potřeba během výběru PAR souboru vybrat *box-kc-ha.pgz* soubor, který byl vytvořen během zálohování obou firewallů. Postup následně probíhá stejně a po vytvoření instalačního disku je možné jej vložit do sekundárního firewallu a provést restart dle návodu výše.

Jestliže i druhé obnovení proběhne úspěšně, bude možné oba firewally vidět v liště *CONTROL* jako *Active* nebo *Standby* pro primární FW a *Standby* pro sekundární FW podobně jako na obrázku níže. Je důležité opět nezapomenout na naformátování disku.



Obrázek 35: Režim primárního a sekundárního FW po úspěšném obnovení (Zdroj: (60))

3.3 Časová analýza

3.3.1 Identifikace činností

Pro projekt byly identifikovány následující činnosti uvedené v tabulce níže a byla jim přidělena délka trvání. Tyto činnosti jsou převážně na sobě závislé, takže nemohou začít nebo skončit, než skončí činnost předcházející, což se později projeví u metody CPM, kde bude možné vidět kritickou cestu celého projektu. Další důležitým poznatkem je délka činností, protože činnosti Školení a Sandboxing obě trvají dva měsíce, a tudíž výrazně zvyšují délku celého projektu. Nicméně jsou to pravděpodobně nejdůležitější činnosti, protože muselo dojít k zaškolení technického pracovníka pověřeného migrací.

Tabulka 16: Identifikované činnosti projektu s délkou trvání (Zdroj: Vlastní zpracování)

Popis činnosti	Délka trvání činnosti (t)
Zadání projektu	2 dny
Školení IT pracovníka	60 dní
Sandboxing (vlastní procvičování)	60 dní
Tvorba skriptu	5 dní
Práce s REST API	2 dny
Testování	1 den
Vytvoření prostředí pro exekuci skriptu	0,5 dne
Migrace objektů	1 den
Kontrola (současná)	0,5 dne
Kontrola (po importu)	0,5 dne
Doplnění chybějících objektů	0,5 dne
Příprava FW pravidel pro migraci	5 dní
Vyčlenění první pravidel pro migraci	0,5 dne
Provedení zkušební části migrace	1 den

Kontrola zkušební migrace	0,5 dne
Úplná migrace	4 dny
Kontrola (současně)	4 dny
Předání výsledku migrace	0,5 dne
Úpravy a zhodnocení připravenosti	5 dní
Přepojení FWs	1 den
Ukončení projektu	2 dny

3.3.2 RACI matice

Na základě stanovených činností byla vytvořena RACI matice, která přiřazuje jednotlivým členům projektového týmu odpovědnosti za vypracování činností. První řádek tabulky obsahuje členy projektového týmu. Hlavní člen pověřený řešením většiny činností je IT pracovník.

Tabulka 17: RACI matice projektu migrace (Zdroj: Vlastní zpracování)

Popis činnosti	Vedoucí projektu	Odborný konzultant	IT pracovník	Vedoucí oddělení
Zadání projektu	R/A	C	I	R/C
Školení IT pracovníka	R/A		R	I
Sandboxing (vlastní procvičování)	I/C		R/A	
Tvorba skriptu	I/C		R/A	
Práce s REST API	I/C		R/A	
Testování	I/C	C	R/A	
Vytvoření prostředí pro exekuci skriptu	R/A	C	R	
Migrace objektů	I/C	C	R/A	I
Kontrola (současná)	I/C		R/A	

Kontrola (po importu)	I/C		R/A	
Doplnění chybějících objektů	I/C	C	R/A	
Příprava FW pravidel pro migraci	R/A	R	I	
Vyčlenění první pravidel pro migraci	I/C	R/A	I	
Provedení zkušební části migrace	I/C	C	R/A	
Kontrola zkušební migrace	I/C	C	R/A	
Úplná migrace	I/C	C	R/A	I
Kontrola (současně)	I/C	C	R/A	
Předání výsledku migrace	R	I	R/A	I
Úpravy a zhodnocení připravenosti	R/A	C	I	I
Přepojení FWs	R/A	C	R	I
Ukončení projektu	R/A	I	I	R

3.3.3 CPM

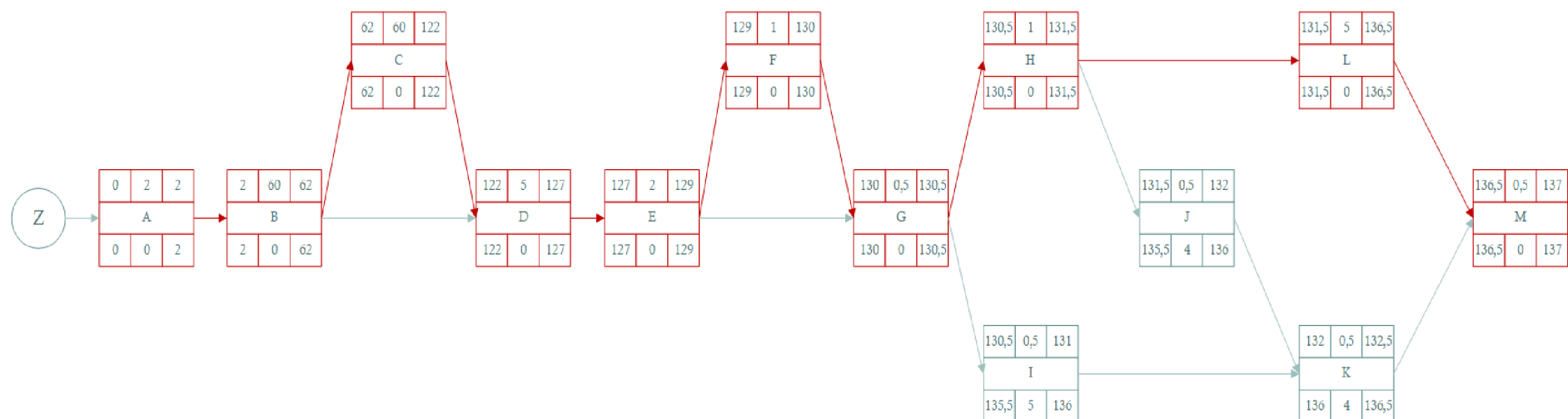
V příloze č. 3 jsou činnosti, jejich identifikační písmeno, doba trvání, předcházející činnost a časové charakteristiky. Činnosti s RC rovnou nule jsou činnosti ležící na kritické cestě, což je nejkratší délka trvání projektu. Jedná se o činnosti *A, B, C, D, E, F, G, H, L, M, N, O, P, R, S, T, U*. Doba trvání projektu, pokud nedojde ke zpoždění, je *151 dní*.

Časové charakteristiky:

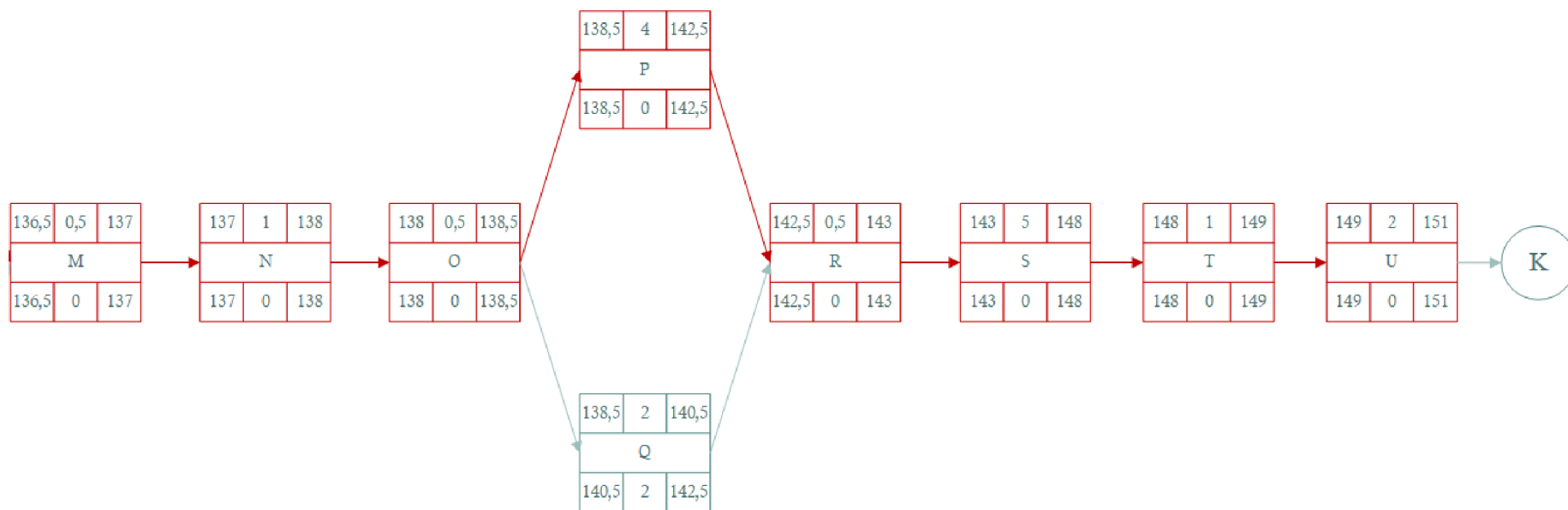
- ZM – Začátek možný
- KM – Konec možný
- KP – Konec přípustný
- ZP – Začátek přípustný
- RC – Rezerva celková

3.3.4 Síťový graf

Pro lepší znázornění návaznosti činností a zvýraznění kritické cesty byl vytvořen síťový graf, který byl pro čitelnost rozdělen do dvou obrázků, toto *rozdělení* bylo provedeno v činnosti M. Jednotlivé činnosti jsou znázorněny uzly, označené příslušným identifikačním písmenem a červené činnosti značí kritickou cestu. Uvnitř uzlu jsou navíc časové charakteristiky z předešlé tabulky. Kritická cesta odpovídá také tabulce a vede přes uzel $A > B > C > D > E > F > G > H > L > M > N > O > P > R > S > T > U$. Začátek grafu je označen kruhem Z (Začátek) a konec je označen kruhem K (Konec).



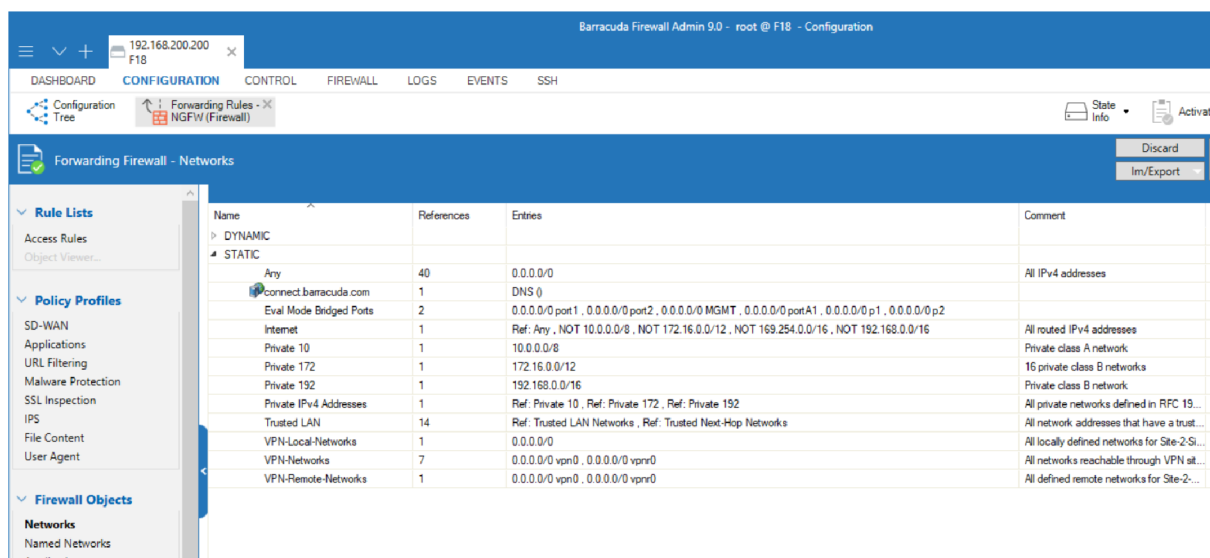
Obrázek 36: Síťový graf - část 1. (Zdroj: Vlastní zpracování)



Obrázek 37: Síťový graf - část 2. (Zdroj: Vlastní zpracování)

3.4 Migrace objektů

Před samotnou migrací pravidel, je potřeba vytvořit objekty, které v těchto pravidlech budou použity. Jedná se o *síťové objekty* jako například *IP adresy*, *IP range*, *subnety*, *netgroupy* a *hostnames*. Tyto objekty, jak již bylo zmíněno, ulehčují správu pravidel, snižují chybovost a dělají nastavení firewallu přehlednější.

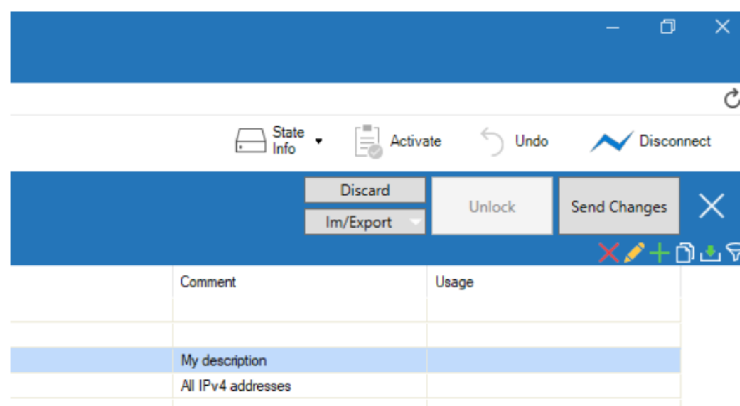


Name	References	Entries	Comment
Any	40	0.0.0.0/0	All IPv4 addresses
connect.barracuda.com	1	DNS ()	
Eval Mode Bridged Ports	2	0.0.0.0/0 port1, 0.0.0.0/0 port2, 0.0.0.0/0 MGMT, 0.0.0.0/0 portA1, 0.0.0.0/0 p1, 0.0.0.0/0 p2	
Internet	1	Ref: Any, NOT 10.0.0.0/8, NOT 172.16.0.0/12, NOT 169.254.0.0/16, NOT 192.168.0.0/16	All routed IPv4 addresses
Private 10	1	10.0.0.0/8	Private class A network
Private 172	1	172.16.0.0/12	16 private class B networks
Private 192	1	192.168.0.0/16	Private class B network
Private IPv4 Addresses	1	Ref: Private 10, Ref: Private 172, Ref: Private 192	All private networks defined in RFC 19...
Trusted LAN	14	Ref: Trusted LAN Networks, Ref: Trusted Next-Hop Networks	All network addresses that have a trust...
VPN-Local-Networks	1	0.0.0.0/0	All locally defined networks for Site-2-Si...
VPN-Networks	7	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All networks reachable through VPN sit...
VPN-Remote-Networks	1	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All defined remote networks for Site-2-...

Obrázek 38: Přehled síťových objektů v Barracuda CGFW (Zdroj Vlastní zpracování)

3.4.1 Ruční migrace objektů

Jedním ze způsobů vložení nových objektů je ruční vytvoření v *GUI*. Pro vytvoření nového objektu je potřeba v první řadě zamknout nastavení a kliknout na zelený symbol plus pro přidání nového objektu.



Obrázek 39: Uzamčení konfigurace síťových objektů (Zdroj Vlastní zpracování)

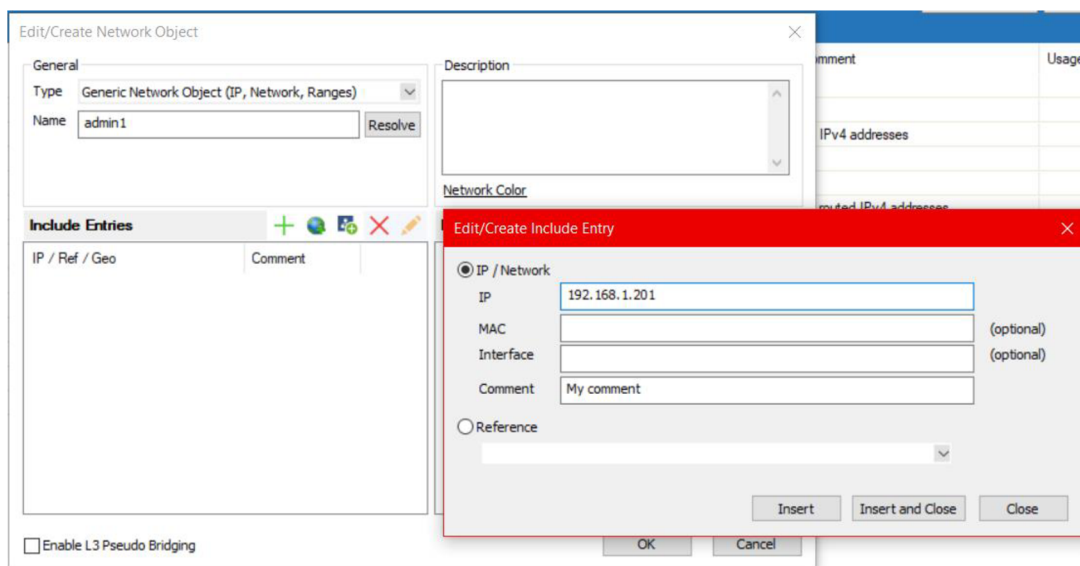
Dále se otevře rozhraní pro editaci a tvorbu nových síťových objektů. Je zde několik hodnot, které je nutné vyplnit. Patří sem *typ objektu*, *název* a *included entries*, což je minimum požadavků, které objekt musí splňovat. Další hodnoty, které nejsou povinné, jsou *popis*, *barva*, *excluded entries* a *geolocation included* a *excluded entries*.

Obrázek 40: Editor síťových objektů (Zdroj Vlastní zpracování)

V první řadě je potřeba zvolit typ objektu, který se vytváří, což se provede jednoduše z rozbalovacího seznamu. Pro tento příklad postačí *Generic Network Object*, protože vytváříme IP adresu. Objekt zároveň pojmenujeme třeba admin1.

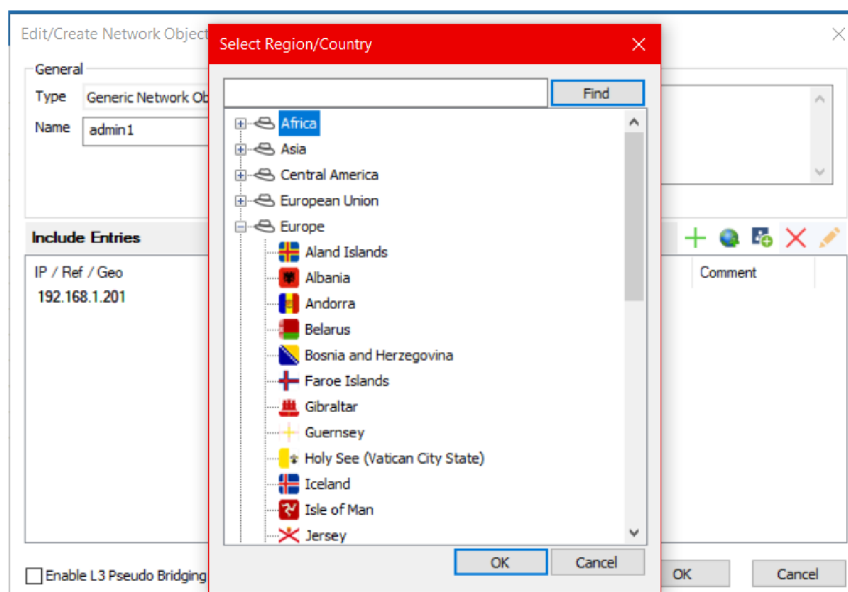
Obrázek 41: Editor síťových objektů - typ objektu (Zdroj Vlastní zpracování)

V *Include Entries* dále zvolíme zelené znaménko plus a otevře se nám další menu. V tomto menu je možné přidat *IP adresu*, *MAC adresu* nebo *Interface* objektu. Je možné také přidat popis anebo zvolit referenci, pokud se odkazujeme na již existující objekty. Přidáme IP adresu pro objekt *admin1* a také popis *My comment* a zmáčkneme tlačítko *Insert and Close*.



Obrázek 42: Editor síťových objektů - přidání IP adresy (Zdroj Vlastní zpracování)

Podobným způsobem je možné přidat do těchto listů i geologické lokace pomocí tlačítka, které vypadá jako *globus*. Jednotlivé lokace jsou rozděleny podle kontinentů nebo jejich částí a je možné vybrat jednotlivé státy, které se přidají do vybraného listu.



Obrázek 43: Editor síťových objektů - přidání regionu/země (Zdroj Vlastní zpracování)

Obrázek 44: Editor síťových objektů - vyplněné údaje (Zdroj Vlastní zpracování)

Pro přidání vytvořeného objektu jej můžeme přidat tlačítkem *OK* a zobrazí se v seznamu všech objektů. Je možné se na něj teď odkazovat a používat například při vytváření pravidel nebo dalších objektů. Pokud v budoucnu budou provedeny úpravy tohoto objektu, upraví se automaticky i v pravidlech, které jej používají.

Name	References	Entries	Comment	Usage
admin1	0	192.168.1.201	My description	
Any	40	0.0.0.0/0	All IPv4 addresses	
connect.barracuda.com	1	DNS ()		
Eval Mode Bridged Ports	2	0.0.0.0/0 port1, 0.0.0.0/0 port2, 0.0.0.0/0 MGMT, 0.0.0.0/0 portA1, 0.0.0.0/0 p1, 0.0.0.0/0 p2		
Internet	1	Ref: Any, NOT 10.0.0.0/8, NOT 172.16.0.0/12, NOT 169.254.0.0/16, NOT 192.168.0.0/16	All routed IPv4 addresses	
Private 10	1	10.0.0.0/8	Private class A network	
Private 172	1	172.16.0.0/12	16 private class B networks	
Private 192	1	192.168.0.0/16	Private class B network	
Private IPv4 Addresses	1	Ref: Private 10, Ref: Private 172, Ref: Private 192	All private networks defined in RFC 19...	
Trusted LAN	14	Ref: Trusted LAN Networks, Ref: Trusted Next-Hop Networks	All network addresses that have a trust...	
VPN-Local-Networks	1	0.0.0.0/0	All locally defined networks for Site-2-Si...	
VPN-Networks	7	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All networks reachable through VPN sit...	
VPN-Remote-Networks	1	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All defined remote networks for Site-2-...	

Obrázek 45: Nově přidáný síťový objekt admin1 (Zdroj Vlastní zpracování)

3.4.2 Migrace objektů pomocí skriptu

V rámci projektu bylo migrováno několik set objektů a ruční migrace by byla v první řadě časově náročná a zadruhé by mohlo dojít k chybám. Nabízela se zde proto možnost vytvoření skriptu, který by usnadnil konverzi formátu migrovaných objektů na nový formát a jejich následné hromadné nahrání na nový firewall.

První pokus migrace pomocí skriptu

Barracuda CloudGen Firewall nabízí možnost importu objektů z *CSV* a *TXT souborů* s oddělovači. Jednalo se tedy o prvotní plán migrace objektů. Nový, a i starý firewall má svou specifickou syntaxi, která musí být dodržena, aby migrace proběhla, a protože se tyto syntaxe liší, bylo potřeba provést úpravy u každého objektu.

Pro zjednodušení migrace byl vytvořen skript Python, který projde soubor se starými objekty a jednotlivé položky upraví na novou syntaxi, které potom nahraje do nového souboru. Tento nový soubor je možné potom nahrát na nový firewall. Níže bude popsána funkčnost jednotlivých částí skriptu. Na obrázku je možné opět vidět formát ve kterém byly objekty exportovány do CSV souboru.

table	name	ipaddr	subnet	domain	host	dns_mode	ttl	members	bits	begin	end	description	last_changed_by
host	seznam.cz				www.seznam.cz	dns	0					"cg14 on Thu Dec 14 14:58:14 2023"	"
ipaddr	admin1	192.168.1.201										"cg8 on Mon Apr 22 16:51:27 2013"	"
ipaddr	admin2	192.168.1.221										"cg8 on Mon Apr 22 16:51:27 2013"	"
ipaddr	Google-DNS	8.8.8.8										"cg8 on Mon Apr 22 16:51:27 2013"	"
subnet	google-173.194.0.0		173.194.0.0						16			"cg14 on Fri Jan 19 11:09:22 2018"	"
subnet	google-172.217.32.0		172.217.32.0						20			"cg14 on Fri Jan 19 11:05:15 2018"	"
iprange	pc-net								10.10.100.10	10.10.100.20		"cg5 on Tue Aug 16 06:20:31 2011"	"
iprange	barracuda_FWs								10.10.11.41	10.10.11.46		"cg15 on Wed Mar 1 19:58:04 2017"	"
netgroup	admins							ipaddr:192.168.1.201,ipaddr:192.168.1.221				"cg15 on Mon May 7 11:25:39 2018"	"

Obrázek 46: Struktura síťových objektů Forcepoint Sidewinder (Zdroj Vlastní zpracování)

Zde je možné vidět začátek skriptu, kde je prve nahrána knihovna *csv*, která umožňuje používání funkcí pro čtení, zápis a tvorbu CSV souborů, ale i mnohé další. Dále je zde vytvořena první funkce, která očišťuje řetězce v CSV souboru o uvozovky. Tato funkce bude dále volaná v jiných funkcích.

```
1
2 import csv
3
4 def quote_term(old_row):
5     if old_row["description"] in ['"', "'", '"']:
6         return ''
7     return f'"{old_row["description"].strip('"\'')}"'
8
```

Obrázek 47: Skript1 - funkce quote_term (Zdroj Vlastní zpracování)

Jeden z možných typů síťových objektů jsou tzv. *hostname objekty*. Byla pro ně vytvořena funkce, která přetvoří jejich formát na nový formát pro FW a uloží je do nového souboru na nový řádek. Tyto hodnoty v novém formátu jsou každé uloženy v příslušné proměnné, jako například název v proměnné *new_name* nebo popis v proměnné *new_desc*. Na očištění popisu je navíc volána vytvořená funkce *quote_term*. Stěžejní proměnnou je *new_value*, do které se ukládá hodnota host, tedy *hostname* z CSV souboru.

```
8
9 def process_host_row(old_row, csv_writer):
10     new_name = f'"{old_row["name"]}"'
11     new_value = f'"{old_row["host"]}"'
12     new_desc = quote_term(old_row)
13
14     new_row = {'name': new_name, 'value': new_value, 'description': new_desc}
15
16     csv_writer.writerow(new_row)
17
```

Obrázek 48: Skript1 - funkce pro host objekty (Zdroj Vlastní zpracování)

Podobným způsobem byla vytvořena funkce pro IP adresy, změnou je zde typ, který byl změněn na *iplist*. Hodnota nahrávána do proměnné *new_value* je navíc brána ze sloupce *ipaddr*.

```
17
18 def process_ipaddr_row(old_row, csv_writer):
19     new_name = f'"{old_row["name"]}"'
20     new_value = f'"{old_row["ipaddr"]}"'
21     new_desc = quote_term(old_row)
22
23     new_row = {"name": new_name, "value": new_value, "type": f'"iplist"', 'description': new_desc}
24
25     csv_writer.writerow(new_row)
26
```

Obrázek 49: Skript1 - funkce pro IP adresy (Zdroj Vlastní zpracování)

Další objektem a funkcí je *subnet*, kde se uvádí typ objektu *netlist*. Hodnota pro proměnnou *new_value* je tentokrát získaná ze dvou sloupců. Prvním je *subnet* a druhý *bits*, který udává bitovou délku podsítě.

```
27
28 def process_subnet_row(old_row, csv_writer):
29     new_name = f'"{old_row["name"]}"'
30     new_value = f'"{old_row["subnet"]}/{old_row["bits"]}"'
31     new_desc = quote_term(old_row)
32
33     new_row = {"name": new_name, "value": new_value, "type": f'"netlist"', 'description': new_desc}
34
35     csv_writer.writerow(new_row)
36
```

Obrázek 50: Skript1 - funkce pro subnety (Zdroj Vlastní zpracování)

Funkce pro objekty ve skupinách, tedy *netgroup*ech je trochu jiná, jelikož se jednotliví členové musí vkládat po jednom, tedy jeden řádek, jeden člen. Je proto hodnota ve starém CSV souboru rozdělena a uložena do seznamu *ips*. Dále je vytvořena proměnná *new_name*, kam se bude ukládat název.

Hlavní částí funkce je cyklus, který prochází celý CSV soubor postupně a hledá, jestli některý ze starých objektů není typem IP adresa, jestliže ano, porovná název této adresy v příslušné proměnné s hodnotami v proměnné *ips*, jestliže najde shodu, bude tento objekt vložen jako *reference* na nový řádek.

```
37
38 def process_netgroup_row(old_row, csv_writer, rows):
39
40     ips = [x.split(":")[1].strip("") for x in old_row["members"].split(",")]
41
42     refs = []
43
44     for row in rows:
45         if row["table"] == "ipaddr":
46             if row["ipaddr"] in ips:
47                 refs.append(row["name"])
48                 new_name = f'{old_row["name"]}'
49                 new_value = f'{row["name"]}'
50                 new_desc = quote_term(old_row)
51
52                 new_row = {"name": new_name, "value": new_value, "type": f'ref', 'description': new_desc}
53
54                 csv_writer.writerow(new_row)
55
```

Obrázek 51: Skript1 - funkce pro netgroupy (Zdroj Vlastní zpracování)

Pro vytvoření funkce, která by konvertovala *IP rozsahy* bylo opět nutné udělat změnu. Rozsahy jsou totiž ve starém CSV souboru zapsány do dvou sloupců, kde jeden *begin* a druhý *end*. Nový FW přijímá rozsahy ve formátu s pomlčkou na konci. Bylo proto potřeba spojit tyto dvě hodnoty dohromady a poté je nahrát.

```
56
57 def process_iprange_row(old_row, csv_writer):
58     new_name = f'{old_row["name"]}'
59
60     rng = old_row["end"].split(".")[1]
61
62     new_value = f'{old_row["begin"]}-{rng}'
63     new_desc = quote_term(old_row)
64
65     new_row = {"name": new_name, "value": new_value, "type": f'iplist', 'description': new_desc}
66
67     csv_writer.writerow(new_row)
68
```

Obrázek 52: Skript1 - funkce pro IP rozsahy (Zdroj Vlastní zpracování)

Zde je uvedena část skriptu, která umožňuje čtení a zápis v CSV souborech. První polovina je použita pro funkci `process_netgroup_row`, kde bylo potřeba procházet jednotlivé řádky, proto byly uloženy jako pole do proměnné `rows`.

Druhá polovina slouží ke klasickému čtení a zápisu, kde se čte ze souboru `old_data.csv` a nové objekty jsou ukládány jako `new_data.csv`. Veškeré hodnoty uvnitř objektů musí být odděleny středníky.

```
69
70 with open('C:/Users/misav/OneDrive/Plocha/old_data.csv', 'r', encoding="utf-8") as old_csv:
71     rows = [x for x in csv.DictReader(old_csv, delimiter=";")]
72     print(rows)
73
74
75 with open('C:/Users/misav/OneDrive/Plocha/old_data.csv', 'r', encoding="utf-8") as old_csv,
76 open('C:/Users/misav/OneDrive/Plocha/new_data.csv', 'w', newline='', encoding="utf-8") as new_csv:
77     # Vytvořte čtečku a zapisovač CSV
78     csv_reader = csv.DictReader(old_csv, delimiter=";")
79     csv_writer = csv.DictWriter(new_csv, fieldnames=("name", "value", "type", "description"), delimiter=";")
80
81     csv_writer.writeheader()
```

Obrázek 53: Skript1 - čtení a zápis do CSV souborů (Zdroj: Vlastní zpracování)

Hlavním tělem skriptu je cyklus, který hledá shodu ve sloupci `table`, kde je vypsán typ objektu, jestliže dojde ke shodě, bude provedena příslušná funkce, která byla definována výše.

```
81
82     for row in csv_reader:
83         print(row)
84
85         match row["table"]:
86             case "host":
87
88                 process_host_row(row, csv_writer)
89             case "ipaddr":
90
91                 process_ipaddr_row(row, csv_writer)
92
93             case "subnet":
94
95                 process_subnet_row(row, csv_writer)
96
97             case "netgroup":
98
99                 process_netgroup_row(row, csv_writer, rows)
100
101             case "iprange":
102
103                 process_iprange_row(row, csv_writer)
104     print("Operace dokončena. Nový CSV soubor vytvořen.")
```

Obrázek 54: Skript1 - cyklus volající funkce (Zdroj: Vlastní zpracování)

Níže je možné vidět výsledek přetvoření objektů na nový formát, který je možné nahrát na Barracuda CloudGen Firewall. Během samotného nahrání bylo však zjištěno, že jsou podporované pouze dva typy síťových objektů pro nahrání pomocí CSV souboru. Těmito typy jsou *iplist* a *netlist*, což nesplňuje požadavky na migraci, kde většinu objektů tvoří *ipadresy*. Byl proto zvolen nový postup za použití technologie REST API, který bude popsán dále.

name	hodnota	typ	popis
"seznam.cz"	"www.seznam.cz"		
"admin1"	"192.168.1.201"	"iplist"	
"admin2"	"192.168.1.221"	"iplist"	
"Google-DNS"	"8.8.8.8"	"iplist"	
"google-173.194.0.0"	"173.194.0.0/16"	"netlist"	
"google-172.217.32.0"	"172.217.32.0/20"	"netlist"	"hello"
"pc-net"	"10.10.100.10-20"	"iplist"	
"barracuda_FWs"	"10.10.11.41-46"	"iplist"	
"admins"	"admin1"	"ref"	
"admins"	"admin2"	"ref"	

Obrázek 55: Skript1 - přetvořené síťové objekty do nového formátu (Zdroj: Vlastní zpracování)

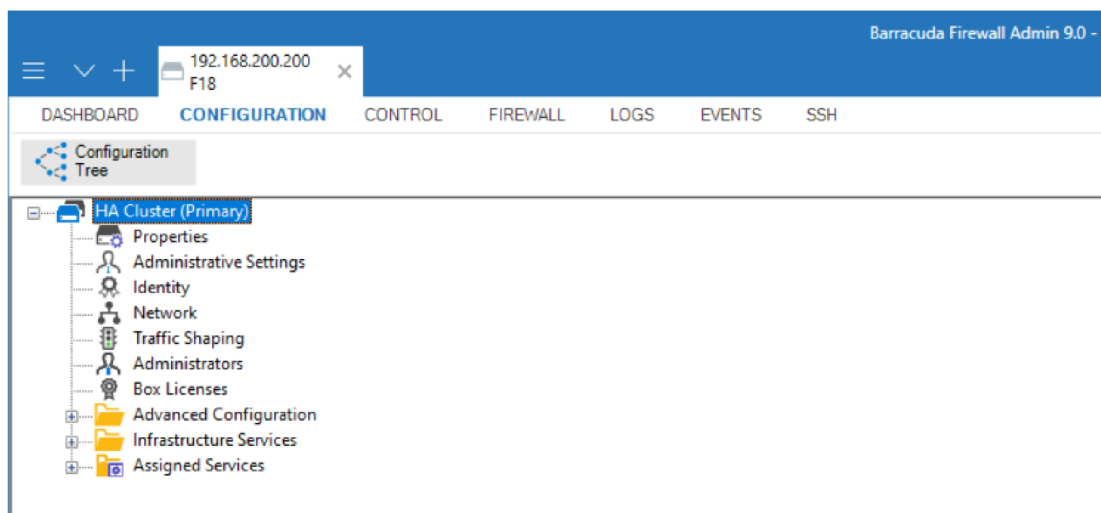
Druhý pokus migrace pomocí skriptu

Kvůli zjištění po prvním pokusu migrace, byla jako náhradní řešení zvolena migrace objektů pomocí RESTful API, které je podporováno v Barracuda CloudGen Firewallu, a které umožňuje migrování všech typů síťových objektů pomocí požadavků HTTP a HTTPS. V tomto případě není již potřeba vytvářet nový CSV soubor, protože objekty budou nahrávány na FW. Opět pro ulehčení práce a snížení chybovosti byl využit Python, kde navíc mohl být použit skript nebo alespoň jeho část z prvního pokusu o migraci.

Povolení komunikace s REST API

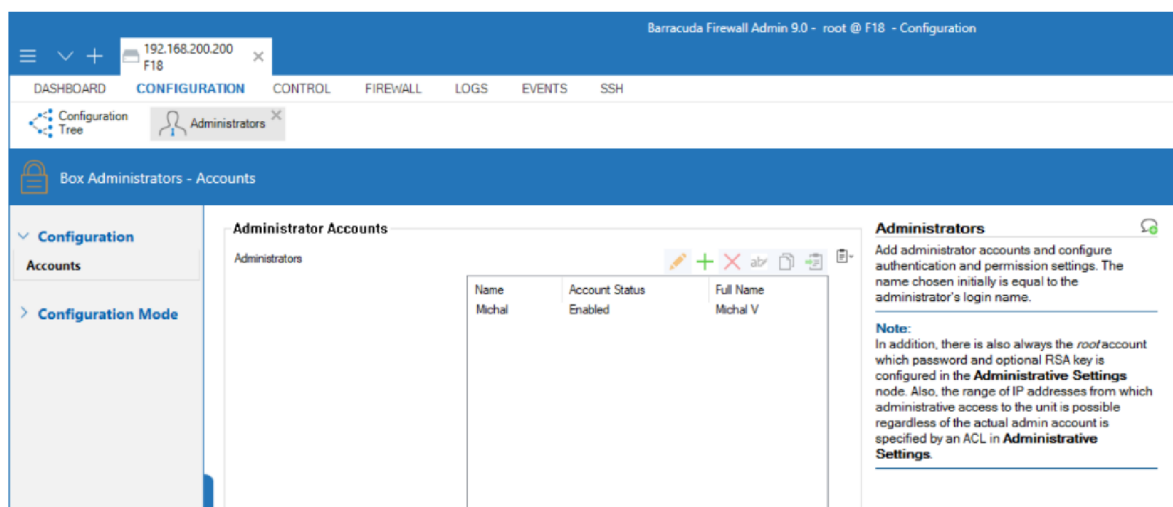
V první řadě je ale potřeba ve FW povolit komunikaci s REST API, což je provedeno následovně.

V panelu *CONFIGURATION* zvolíme *Administrators* v konfiguračním stromu, kde budeme moci přidat nového administrátora, pomocí jehož údajů se budou požadavky REST API ověřovat vůči FW.



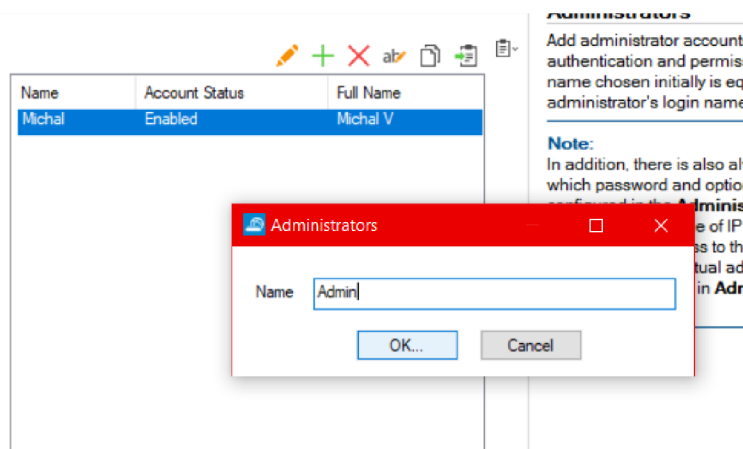
Obrázek 56: Konfigurační strom - Administrators (Zdroj: Vlastní zpracování)

Nyní se otevře nastavení *Administrators*, kde můžeme vidět všechny přidané administrátory. Pro vytvoření nového administrátora, zamkneme oddíl pomocí tlačítka *Lock* a klikneme na ikonku zeleného plus nebo pravým tlačítkem do seznamu administrátorů a zvolíme *New*.



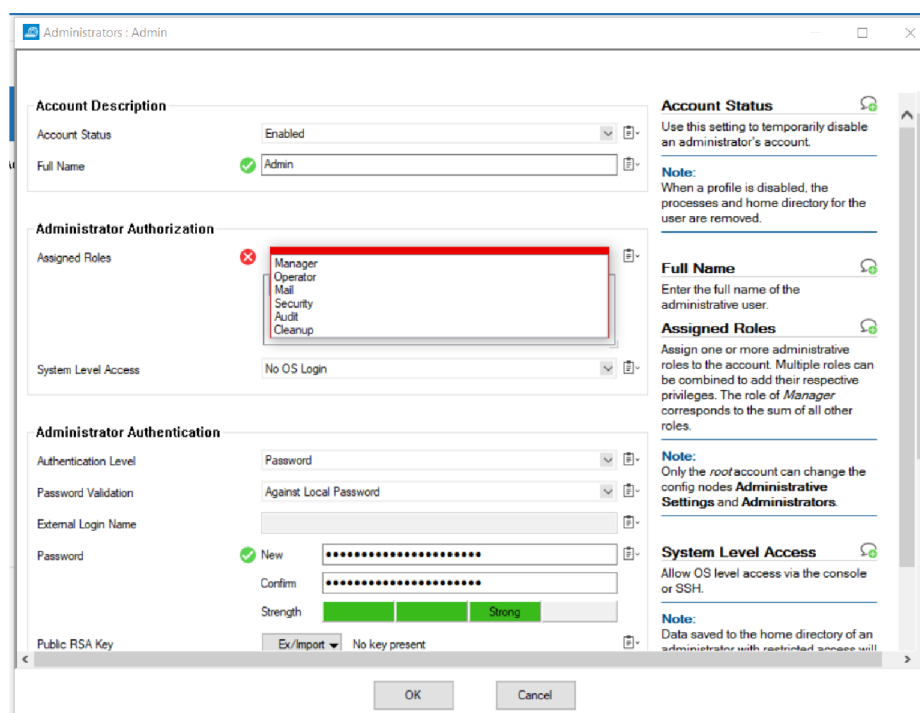
Obrázek 57: Konfigurace administrátorů (Zdroj: Vlastní zpracování)

Následně se otevře vyskakovací okno, kam napíšeme název našeho nového administrátora. Vytvoříme administrátora *Admin* a zmáčkeme *OK*, čímž se otevře okno pro přidání nového administrátora.



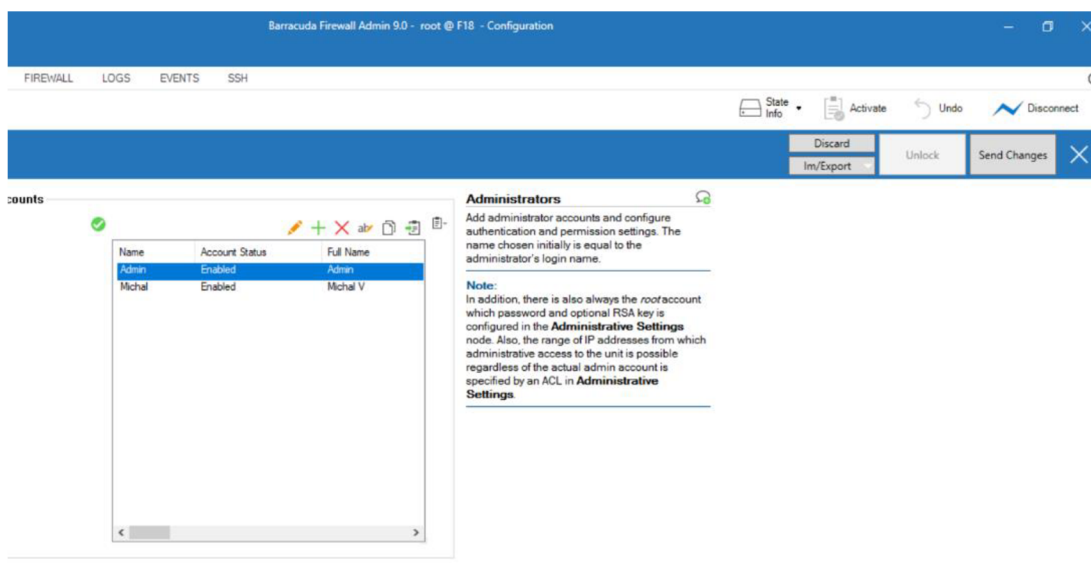
Obrázek 58: Přidání administrátora - název (Zdroj: Vlastní zpracování)

Zde je nutné přidat více podrobností ohledně vytvářeného administrátora. Můžeme aktivovat a deaktivovat administrátora, musíme přidat celé jméno, musíme zvolit role, které určují kam může administrátor vidět a co může měnit, a především musíme přidat typ autentizace a v tomto případě i heslo. Jakmile jsou všechny hodnoty zadane, stačí potvrdit tlačítkem *OK*.



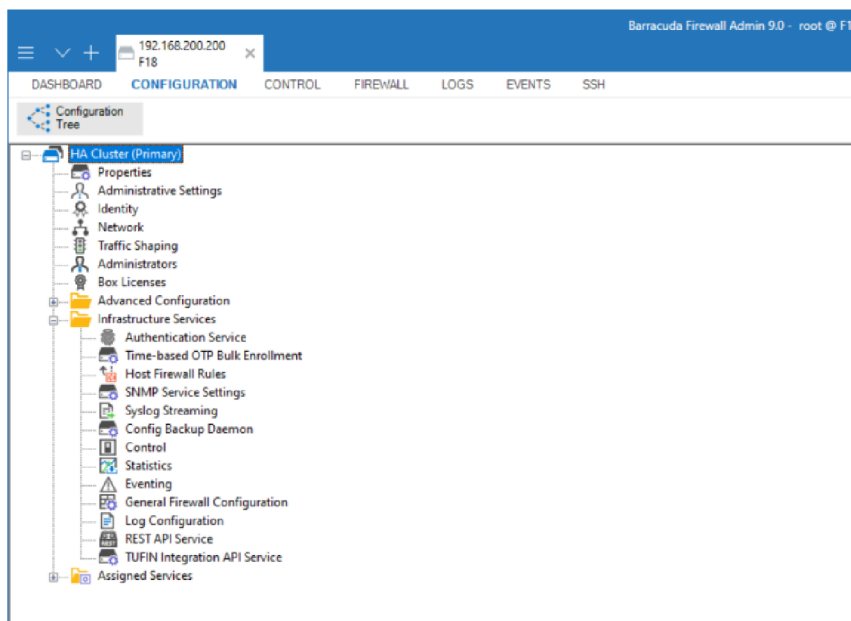
Obrázek 59: Přidání administrátora - další nastavení (Zdroj: Vlastní zpracování)

Nový administrátor se zobrazí v seznamu administrátorů. Je teď nutné odeslat změny firewallu pomocí tlačítka *Send Changes* a aktivovat tyto změny tlačítkem *Activate*. Administrátor je tak připravený k použití.



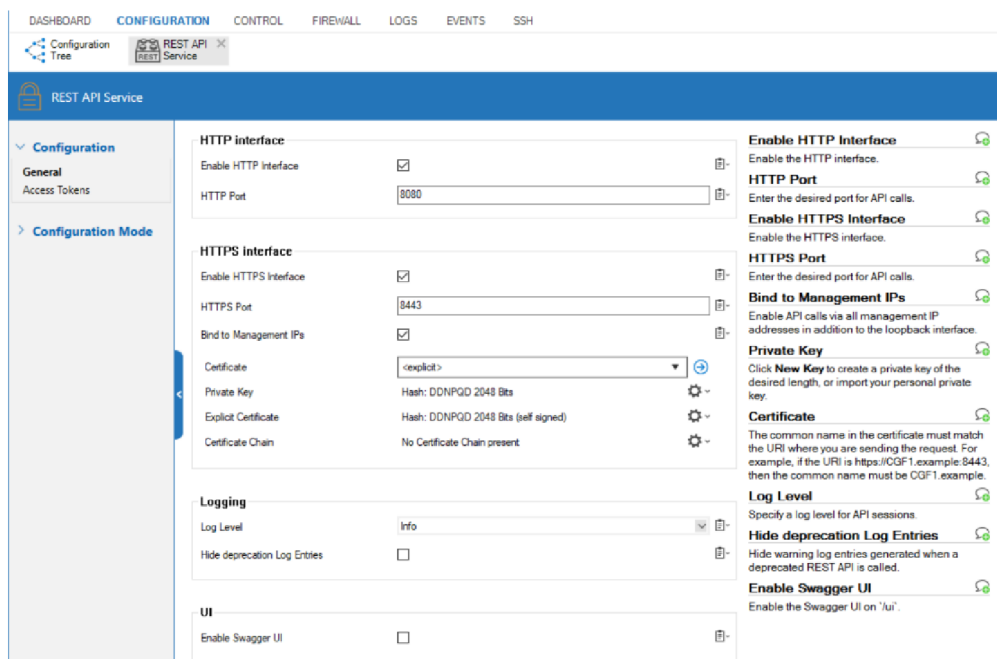
Obrázek 60: Přidání administrátora - nově vytvoření administrátor Admin (Zdroj: Vlastní zpracování)

Kromě vytvoření administrátora je potřeba také povolit komunikaci REST API, což je nutné v první řadě povolit v nastavení *REST API Service*, kam se dostaneme z konfiguračního stromu v záložce *CONFIGURATION*. Rozbalíme *Box (HA Cluster)*, *Infrastructure Services* a otevřeme *REST API Service*.



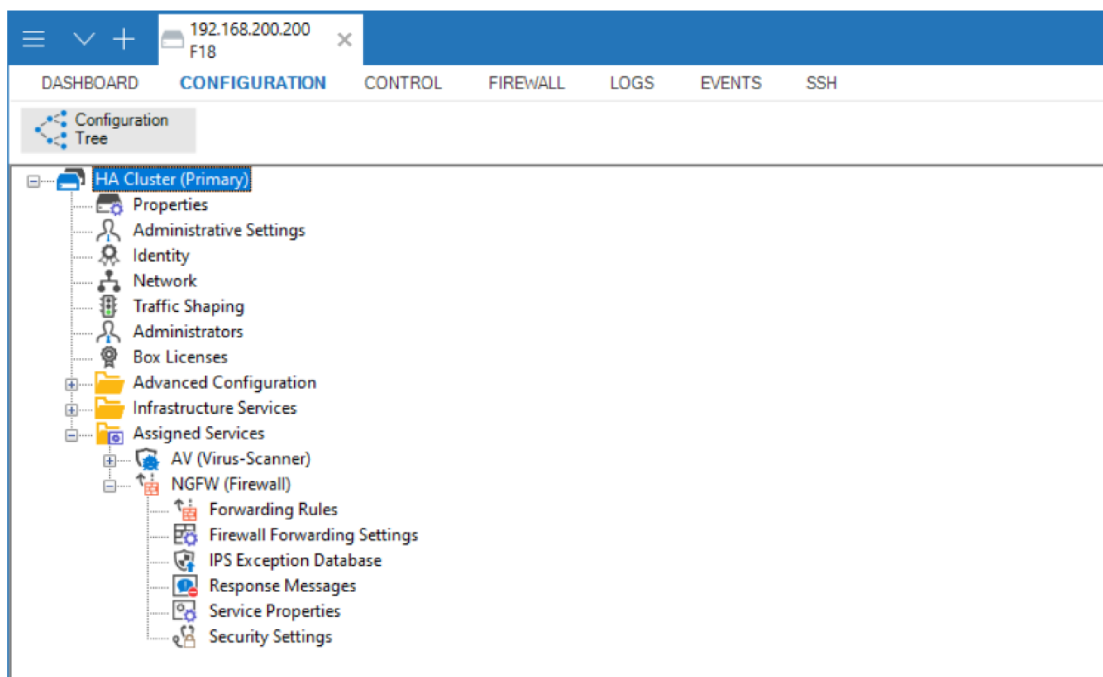
Obrázek 61: Konfigurační strom - Infrastructure Services - REST API Service (Zdroj: Vlastní zpracování)

Zde můžeme povolit HTTP anebo i HTTPS rozhraní na příslušných portech 8080 a 8443 pro REST API. Pro funkčnost komunikace přes protokol HTTPS je nutné vytvořit a poté přidělit certifikát, kterým se potom požadavky budou ověřovat. Toto řešení nabízí další úroveň zabezpečení, jestliže by se REST API používalo při běžném provozu. Pro migraci objektů bude stačit komunikace přes HTTP.



Obrázek 62: Povolení REST API (Zdroj: Vlastní zpracování)

Druhý krok k povolení komunikace s REST API je vytvoření pravidla *Access Rule*, které veškerou komunikaci povolí a přesměruje ji na správnou adresu firewallu. Abychom se k vytvoření pravidla dostali, je potřeba v konfiguračním stromu v záložce *CONFIGURATION* rozbalit záložku *Assigned Services* u odpovídajícího *Boxu*, dále rozbalit *NGFW (Firewall)* a otevřít *Forwarding Rules*



Obrázek 63: Konfigurační strom - Assigned Services - NGFW (Firewall) - Forwarding Rules (Zdroj: Vlastní zpracování)

Otevře se nám *Forwarding Ruleset*, tedy seznam všech pravidel *Access Rules*. Pro přidání nového pravidla uzamkneme nastavení pomocí tlačítka *Lock* a přidáme nové pravidlo zeleným tlačítkem plus nebo kliknutím pravým tlačítkem myši a zvolením *New*.

ON CONTROL FIREWALL LOGS EVENTS SSH										
Forwarding Rules - (Firewall)										
Main Rules										
Action	Name	Features	Policy Usage	NAT Mode	Service	Source	Destination	User	Sche...	QoS
0	Section1 (12)									
1	Pass Dynamic NAT	BOX-BARRACUDA-CUD...	None	Explicit NAT	ScreenConnect	127.0.0.9, 192.168.200.0/24	connect.barracuda.com	Any	Always	High (ID 2)
2	Block	BlockATDQuarantine	N.A.	N.A.	Any ALLIP, ECHO, TCP, TCP...	Any	Any	Any	Always	N.A.
3	Pass SD-WAN	BOX-EVAL-MODE-BRIDGE	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
4	Pass SD-WAN	BOX-SERVICEIP-MGMT...	None	Explicit NAT	Ref: HTTPS, Ref: NGF-MG...	Ref: HTTPS, Ref: NGF-MG...	Ref: HTTPS, Ref: NGF-MG...	Any	Always	No-Shaping
5	App Redirect	BOX-SETUP-MGMT-ACC...	N.A.	N.A.	Ref: NGF-MGMT-BOX, Ref...	Ref: NGF-MGMT-BOX, Ref...	Ref: NGF-MGMT-BOX, Ref...	Any	Always	High (ID 2)
6	App Redirect	BOX-LOCALDNSCACHE	N.A.	N.A.	ECHO, TCP, Z2, TCP, 443...	ECHO, TCP, Z2, TCP, 443...	ECHO, TCP, Z2, TCP, 443...	Any	Always	Realtime (ID 1)
7	Pass SD-WAN	BOX-LAN-2-INTERNET	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
8	Pass SD-WAN	BOX-LAN-2-LAN	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
9	Pass SD-WAN	BOX-LAN-2-VPN-SITE	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
10	Pass SD-WAN	BOX-VPN-SITE-2-SITE	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
11	Pass SD-WAN	BOX-VPNCLIENTS-2-LAN	All Policy Pr...	N.A.	Any	Any	Any	Any	Always	N.A.
12	Block	BLOCKALL	N.A.	N.A.	Any	Any	Any	Any	Always	N.A.

Obrázek 64: Forwarding rules (Zdroj: Vlastní zpracování)

V menu, které se otevře vyplníme vše následujícím způsobem. Typ pravidla je *App Redirect*, název je *RestAPI-Access*, jako *Source* zvolíme *Trusted LAN* a u *Destination*

zvolíme *Management IP*. V poslední řadě je potřeba zvolit *Service*, zde zvolíme *TCP 8080*, tedy port pro komunikaci REST API přes *HTTP*. Nakonec je potřeba přidat adresu, na kterou bude komunikace přeměřována. Touto adresou je *loopback adresa 127.0.0.1* na *portu 8080*, kterou napíšeme do kolonky *Redirection*. Pravidlo uložíme tlačítkem *OK*.

Obrázek 65: Vytvoření Access Rule pro REST API (Zdroj: Vlastní zpracování)

Na obrázku níže je možné vidět nově vytvořené pravidlo pro přístup REST API. Nyní je možné provádět plnou administraci a konfiguraci firewallu pomocí REST API, a tudíž i přidávat síťové objekty.

ION CONTROL FIREWALL LOGS EVENTS SSH									
Loading Rules - X (Firewall)									
Main Rules									
Action	Name	Features	Policy Usage	NAT Mode	Service	Source	Destination	User	
0 App Redirect 127.0.0.1:8080	RestAPI-Access		N.A.	N.A.	TCP 8080	Trusted LAN 127.0.0.9, 192.168.200.0/24	Management IP 192.168.200.200	Any	
1 Pass Dynamic NAT	BOX-BARRACUDA-CUD...		None	Explicit NAT	ScreenConnect TCP 443, TCP 8040, TCP ...	Ref: Trusted LAN, Ref: WI... 127.0.0.9, 192.168.200.0/24	connect.barracuda.com	Any	
2 Block	BlockATDQuarantine		N.A.	N.A.	Any ALLIP, ECHO, TCP *, TCP ...	ATD Quarantine 0.0.0.0/0 dev MGMT, 0.0.0...	Any 0.0.0.0/0	Any	
3 Pass SD-WAN	BOX-EVAL-MODE-BRIDGE		All Policy Pr...	N.A.	Any ALLIP, ECHO, TCP *, TCP ...	Eval Mode Bridged Ports 0.0.0.0/0 dev MGMT, 0.0.0...	Eval Mode Bridged Ports 0.0.0.0/0 dev MGMT, 0.0.0...	Any	
4 Dest NAT Primary Management IP, On...	BOX-SERVICEIP-MGMT...		None	Explicit NAT	Ref: HTTPS, Ref: NGF-MG... ECHO, TCP 22, TCP 443...	BoxACL 0.0.0.0/0	Service IPs 127.0.0.9	Any	
5 App Redirect 192.168.200.200	BOX-SETUP-MGMT-ACC...		N.A.	N.A.	Ref: NGF-MGMT-BOX, Ref... ECHO, TCP 22, TCP 443...	Private IPv4 Addresses 10.0.0.0/8, 172.16.0.0/12...	DHCP1 Local IP	Any	
6 App Redirect 127.0.0.1:53	BOX-LOCALDNSCACHE		N.A.	N.A.	DNS TCP 53, UDP 53	Trusted LAN 127.0.0.9, 192.168.200.0/24	Any 0.0.0.0/0	Any	
7 Pass SD-WAN	BOX-LAN-2-INTERNET		All Policy Pr...	N.A.	Any ALLIP, ECHO, TCP *, TCP ...	Ref: Trusted LAN 127.0.0.9, 192.168.200.0/24	Ref: Internet 0.0.0.0/0, NOT 10.0.0.0/8, ...	Any	
8 Pass SD-WAN	BOX-LAN-2-LAN		All Policy Pr...	N.A.	Any ALLIP, ECHO, TCP *, TCP ...	Trusted LAN 127.0.0.9, 192.168.200.0/24	Trusted LAN 127.0.0.9, 192.168.200.0/24	Any	

Obrázek 66: Nově vytvořené pravidlo pro REST API (Zdroj: Vlastní zpracování)

Druhý pokus migrace pomocí skriptu

Díky povolení komunikace pro REST API je nyní možné vytvořit skript, který by odesílal požadavky na FW, a které budou přidávat jednotlivé síťové objekty.

Do samotného skriptu byly importovány knihovny *csv*, *requests*, *copy* a navíc *HTTPBasicAuth* z knihovny *requests.auth* (viz obrázek níže, řádek 2 až 5). Tyto knihovny umožní práci s CSV soubory, odesílání REST API requests a autentizaci pomocí jména a hesla u nově vytvořeného administrátora.

Dále byl do proměnné *url* uložený odkaz pro odesílání požadavků firewallu. Zde je potřeba místo *your_fw_ip* vložit *management IP*, kterou firewall používá a za ni přidat *port* komunikace *HTTP protokolu*. Defaultně je tato IP nastavena na *192.168.200.200*.

Obdobně byla do proměnné *header* uložena hlavička, která specifikuje typ souboru, který je posílán na *json*, jméno do proměnné *name* a heslo do proměnné *passw*. Tyto vytvořené proměnné budou používány během odesílání požadavků firewallu.

```
2 import csv
3 import requests
4 import copy
5 from requests.auth import HTTPBasicAuth
6
7 #url for communicating with the FW using the HTTP protocol on port 8080, 192.168.200.200 is the default mngmnt IP
8 #rest api HTTP communication had to be allowed through an access rule in the FW
9 url = "http://your_fw_ip:8080/rest/config/v1/forwarding-firewall/objects/networks?envelope=false"
10 header = {'Content-Type': 'application/json'}
11 #authentication credential for FW administrator
12 name = 'admin'
13 passw = 'password'
```

Obrázek 67: Skript2 - import knihoven a vytvoření hlaviček (Zdroj: Vlastní zpracování)

Požadavky firewallu jsou doručovány jako *payload*, který má specifickou kostru. Tato kostra byla zjednodušena a byly ponechány pouze *parametry* potřebné pro migraci stávajících objektů. Tato kostra byla následně uložena do proměnné *payload*.

```

16
17 #the payload for REST API requests on Barracuda CloudGen FW
18 #only the parameters needed for this migration are used here
19 payload = {
20     "comment": "This is my object.",
21     "dnsLifetime": 600,
22     "excluded": [
23     ],
24     "geo": {
25         "excluded": [
26         ],
27         "included": [
28         ]
29     },
30     "included": [
31         {
32             "entry": {
33                 "ip": "127.0.0.3"
34             }
35         }
36     ],
37     "name": "MyObject3",
38     "type": "generic"
39 }

```

Obrázek 68: Skript2 - vytvoření proměnné payload (Zdroj: Vlastní zpracování)

Nyní je potřeba definovat *funkce* pro práci s CSV souborem a *payload*. Jako první byla definována funkce pro odstranění parametru *dnsLifetime* z *payload*. Tuto funkci budou volat jiné funkce, které pracují s objekty bez *dnsLifetime*, jako je například IP adresa, IP rozsah apod.

```

41
42 #function for removing parameter 'dnsLifetime' from new_payloadu for functions which don't need dnsLifetime
43 def term_dns(new_payload):
44     del new_payload['dnsLifetime']
45

```

Obrázek 69: Skript2 - funkce pro odstranění dnsLifetime (Zdroj: Vlastní zpracování)

Druhá funkce je upravená funkce z původního skriptu, kde při kopírování popisku objektu ze sloupce *description*, jsou odstraněny veškeré *uvozovky*.

```

46
47 #function for adjusting descriptions from csv file for migration
48 def quote_term(old_row):
49     if old_row["description"] in ['"', "'", '']: #if the string contains either ", ' or '
50         return '' #returns empty value
51     return f'{old_row["description"].strip("\'\"")}' #in other case strips the string off " or '
52

```

Obrázek 70: Skript2 - funkce quote_term (Zdroj: Vlastní zpracování)

První funkce pro vkládání objektů slouží pro migraci *IP adres*. Po zavolání je automaticky vytvořena kopie *payload* do proměnné *new_payload*. Tato kopie umožňuje provádění úprav pro různé typy objektů, aniž by vznikaly chyby. Následně je zavolána funkce *term_dns*, která byla popsána výše.

Poté je do proměnné *new_name*, *new_value* a *new_desc* uloženo jméno, IP adresa a popis v tomto pořadí. Hodnota *new_name* je navíc během procesu očištěna o uvozovky a během vkládání popisu je zavolána funkce *quote_term*.

Tyto nově vytvořené proměnné jsou poté nahrány do příslušných parametrů v *new_payload*. IP adresa v proměnné *new_payload* je například uložena do parametru *included>entry>ip*. Jako kontrola je následně proveden výpis celé proměnné *new_payload*.

Takto vytvořený *payload* je následně odeslán firewallu uvnitř *request POST*. Tento *request* obsahuje také proměnné *url*, *header* a *name* a *passw*, které byly definovány na začátku kódu. Po odeslání se *request* autentizuje vůči firewallu pomocí definovaných přihlašovacích údajů a bude propuštěn vytvořeným *Access Rule*. Jako kontrola je zde výpis *status kódu*, který udává, jak požadavek proběhl.

```
53
54 #function for migrating ip addresses from csv file
55 def process_ipaddr_row(old_row):
56     new_payload = copy.deepcopy(payload) #forms a copy of the payload for work and alteration
57     term_dns(new_payload) #calling the function to remove dnslifetime
58
59     new_name = f'{old_row["name"].strip("\'\"")}' #extracts the name of the object into variable new_name and strips it off of " or '
60     new_value = f'{old_row["ipaddr"]}' #extracts the ip address into variable new_value
61     new_desc = quote_term(old_row) #calling the function for stripping the description and saving it into variable new_desc
62
63     #inserting new values from the variables into corresponding parameters in new_payload
64     new_payload["included"][0]["entry"]["ip"] = new_value
65     new_payload["name"] = new_name
66     new_payload["comment"] = new_desc
67
68     print(new_payload) #check print
69
70 #sending POST request to the FW, it contains headers, authentication and the new_payload variable. Saving it into response variable
71 response = requests.post(url, json=new_payload, headers=header, auth=HTTPBasicAuth(name, passw))
72 print(response.status_code) #print of the request status code
73
```

Obrázek 71: Skript2 - funkce pro migraci objektů IP adresa (Zdroj: Vlastní zpracování)

Druhou funkcí pro vkládání síťových objektů je funkce *process_subnet_row*, která přidává *podsíť*. Výrazná změna oproti předchozí funkci je hodnota ukládaná do *new_value*, protože jsou zde spojeny dva záznamy z CSV souboru, a to *subnet* a *bits*, což je *bitová délka subnetu*.

Data jsou obdobně nahrána do *payload* v proměnné *new_payload*. Dochází zde ke změně typu objektu na *singleIPv4Network*, což je jedna IPv4 síť. *Payload* je poté odeslán firewallu a je vypsán *status kód*.

```

74
75 #function for migrating subnets from csv file
76 def process_subnet_row(old_row):
77     new_payload = copy.deepcopy(payload)
78     term_dns(new_payload)
79
80     new_name = f'{old_row["name"].strip("\\"')}'
81     new_value = f'{old_row["subnet"]}/{old_row["bits"]}' #combining the subnet value and / and the bit lenght
82     new_desc = quote_term(old_row)
83
84     new_payload["included"][0]["entry"]["ip"] = new_value
85     new_payload["name"] = new_name
86     new_payload["comment"] = new_desc
87     new_payload["type"] = 'singleIPv4Network' #changing the type of the object to singleIPv4Network
88
89     print(new_payload)
90
91     response = requests.post(url, json=new_payload, headers=header, auth=HTTPBasicAuth(name, passw))
92     print(response.status_code)
93

```

Obrázek 72: Skript2 - funkce pro migraci objektů subnet (Zdroj: Vlastní zpracování)

Další funkce migruje *IP rozsahy*. Zde je prováněna konverze hodnoty rozsahu, kde potřeba spojit začátek rozsahu ze sloupce *begin* a konec ze sloupce *end*. Konec rozsahu je očištěn o první část adresy, tak aby zůstalo jen poslední bity a je uložen do proměnné *rng*, který je k začátku připojena v proměnné *new_value*. Všechna data jsou opět nahraná do *payload*, který je následně odeslán na firewall a ten vrátí *status kód*.

```

94
95 #function for migrating IP ranges
96 def process_iprange_row(old_row):
97     new_payload = copy.deepcopy(payload)
98     term_dns(new_payload)
99
100     new_name = f'{old_row["name"].strip("\\"')}'
101     rng = old_row["end"].split(".")[1] #saving the end value of the ip range from csv file into variable rng
102     new_value = f'{old_row["begin"]}-{rng}' #combining the beginning of the range and - and the end part of the ip range (rng)
103     new_desc = quote_term(old_row)
104
105     new_payload["included"][0]["entry"]["ip"] = new_value
106     new_payload["name"] = new_name
107     new_payload["comment"] = new_desc
108
109     print(new_payload)
110
111     response = requests.post(url, json=new_payload, headers=header, auth=HTTPBasicAuth(name, passw))
112     print(response.status_code)
113

```

Obrázek 73: Skript2 - funkce pro migraci objektů IP rozsah (Zdroj: Vlastní zpracování)

Funkce *process_netgroup_row* migruje *netgroups*, které se odkazují již na existující síťové objekty, nejčastěji IP adresy. Po vytvoření kopie *payload* je vymazán obsah parametru *included* z *new_payload*, protože jsou členové *netgroup* vkládáni jako *reference* a nesmí obsahovat jiné části *parametru*, jako třeba *ip*, *mac* nebo *interface*.

Hlavní částí funkce je *cyklus*, který prochází veškeré řádky v CSV souboru a porovnává, zda se jedná o typ objektu *ipaddr*, jestliže je nalezena shoda, porovná následně cyklus hodnotu IP adresy nebo název objektu s hodnotami uloženými v proměnné *ips*, což jsou

jednotlivé *odkazované* objekty z CSV souboru, jestliže je opět nalezena shoda, přidá se do *payload* v parametru *included* název tohoto objektu jako *referenci*.

Jakmile cyklus projde celý soubor a nalezne všechny reference, je *request* odeslán firewallu, který opět vrátí *status kód*.

```
114
115 #function for migrating netgroups
116 def process_netgroup_row(old_row, rows):
117     new_payload = copy.deepcopy(payload)
118     new_payload["included"] = [] #removing parameters from the body of the payload in new_payload
119     term_dns(new_payload)
120
121     ips = [x.split(":")[1].strip("'") for x in old_row["members"].split(",")] #splitting individual referenced ip addresses from the csv file
122
123     refs = []
124
125     new_name = f'{old_row["name"].strip("\'")}'
126     new_desc = quote_term(old_row)
127
128     new_payload["name"] = new_name
129     new_payload["comment"] = new_desc
130
131     #cycle which will go through each row in the csv file
132     for row in rows:
133         if row["table"] == "ipaddr": #if the object type in column is ipaddr
134             if row["ipaddr"] or row["name"] in ips: #compares the value with saved values in variable ips
135                 new_payload["included"].append({
136                     "references": f'{row["name"]}', #adds it into new_payload as a reference if there is a match
137                 })
138
139     print(new_payload)
140     response = requests.post(url, json=new_payload, headers=header, auth=HTTPBasicAuth(name, passw))
141     print(response.status_code)
```

Obrázek 74: Skript2 - funkce pro migraci objektů netgroup (Zdroj: Vlastní zpracování)

Poslední funkcí pro migraci síťových objektů, je funkce *process_host_row*, která migruje objekty typu *hostname*. Po vytvoření kopie *payload*, je z parametru *included* opět odstraněn veškerý obsah. Zároveň zde **není** použita funkce *term_dns*, protože bude potřeba uložit *dnsLifetime* do *payload*.

V Barracuda CloudGen Firewall jsou *hostname* zapisovány přímo do názvu objektu, proto je zde do proměnné *new_name* uložena hodnota ze sloupce *host*. *DnsLifetime* je uložen ze sloupce *ttl* v CSV souboru do proměnné *new_ttl*, která je společně s ostatními proměnnými nahrána do *payload*.

V *payload* je navíc změněn typ objektu na *hostname*, aby jej firewall mohl rozpoznat. Takto vytvořený *payload* je následně odeslán firewallu jako *request* stejným způsobem jako u předcházejících objektů. Ten následně vrátí kontrolní *status kód*.

```

143
144 #function for migrating hostnames
145 def process_host_row(old_row):
146     new_payload = copy.deepcopy(payload)
147     new_payload['included'] = []
148
149     new_name = f'{old_row["host"].strip("\\"')}'
150     new_ttl = int(old_row["ttl"]) #saving dns lifetime from the csv file into a variable
151     new_desc = quote_term(old_row)
152
153     new_payload["dnsLifetime"] = new_ttl #inserting the dnsLifetime into new_payload
154     new_payload["name"] = new_name
155     new_payload["comment"] = new_desc
156     new_payload["type"] = 'hostname' #changing the object typo to hostname
157
158     print(new_payload)
159     response = requests.post(url, json=new_payload, headers=header, auth=HTTPBasicAuth(name, passw))
160     print(response.status_code)
161

```

Obrázek 75: Skript2 - funkce pro migraci objektů hostnames (Zdroj: Vlastní zpracování)

With open umožňuje otevření souboru, je potřeba do závorek zapsat *cestu k vybranému souboru*, zvolit *typ práv*, v tomto případě *r*, což znamená čtení a *způsob encoding* souboru. Takto otevřený soubor pro čtení je uložen do proměnné *old_csv*, která je následně použita v proměnné *rows*, kde je zároveň specifikován *středník* jako oddělovač. Aby bylo možné ve funkci *process_netgroups_row* procházet jednotlivé řádky CSV souboru, jsou uloženy do proměnné *rows* jako *pole*.

```

162
163 #opening the csv file for reading and creation of a list rows, separator: ;
164 with open('C:/Users/misav/OneDrive/Plocha/old_data.csv', 'r', encoding="utf-8") as old_csv:
165     rows = [x for x in csv.DictReader(old_csv, delimiter=";")]
166     print(rows)
167

```

Obrázek 76: Skript2 - čtení CSV souboru a vytvoření pole rows (Zdroj: Vlastní zpracování)

Obdobným způsobem je uloženo do proměnné *csv_reader* čtení z CSV souboru, kde jsou staré objekty. Tato proměnná je používána dál v kódu.

```

168
169 with open('C:/Users/misav/OneDrive/Plocha/old_data.csv', 'r', encoding="utf-8") as old_csv:
170     csv_reader = csv.DictReader(old_csv, delimiter=";") #reading from the csv file saved into variable csv_reader, which is later referenced
171

```

Obrázek 77: Skript2 - čtení CSV souboru uložené do proměnné csv_reader (Zdroj: Vlastní zpracování)

Hlavním tělem skriptu je *cyklus*, který prochází všechny řádky v proměnné *csv_reader*. Je zde použito klíčové slovo *match*, které sleduje hodnotu ve sloupci *table* CSV souboru a v případě, že nalezne shodu určenou u klíčových slov *case*, zavolá příslušnou *funkci* pro migraci daného objektu. Jsou zde definované *case* pro všechny typy migrovaných objektů.

```

172
173 #cycle going through each row of the csv file
174 for row in csv_reader:
175
176     match row["table"]: #case for each type of object and calling the corresponding function
177     case "host":
178         process_host_row(row)
179
180
181     case "ipaddr":
182         process_ipaddr_row(row)
183
184
185     case "subnet":
186         process_subnet_row(row)
187
188
189     case "netgroup":
190
191         process_netgroup_row(row,rows)
192
193
194     case "iprange":
195         process_iprange_row(row)

```

Obrázek 78: Skript2 - cyklus volající vytvořené funkce (Zdroj: Vlastní zpracování)

Na obrázku č. 79 níže je kontrolní výpis všech čtených objektů v CSV souboru.

```

[{'table': 'host', 'name': 'seznam.cz', 'ipaddr': '', 'subnet': '', 'domain': '', 'host': 'www.seznam.cz', 'ipaddrs': '', 'dns_mode': 'dns', 'ttl': '0', 'members': '', 'bits': '', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg14 on Thu Dec 14 14:58:14 2023'}, {'table': 'ipaddr', 'name': 'admin1', 'ipaddr': '192.168.1.201', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg8 on Mon Apr 22 16:51:27 2013'}, {'table': 'ipaddr', 'name': 'admin2', 'ipaddr': '192.168.1.221', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg8 on Mon Apr 22 16:51:27 2013'}, {'table': 'ipaddr', 'name': 'Google-DNS', 'ipaddr': '8.8.8.8', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg8 on Mon Apr 22 16:51:27 2013'}, {'table': 'subnet', 'name': 'google-173.194.0.0', 'ipaddr': '', 'subnet': '173.194.0.0', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '16', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg14 on Fri Jan 19 11:09:22 2018'}, {'table': 'subnet', 'name': 'google-172.217.32.0', 'ipaddr': '', 'subnet': '172.217.32.0', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '20', 'begin': '', 'end': '', 'time_intervals': '', 'description': 'hello', 'last_changed_by': 'cg14 on Fri Jan 19 11:05:15 2018'}, {'table': 'iprange', 'name': 'pc-net', 'ipaddr': '', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '', 'begin': '10.10.100.10', 'end': '10.10.100.20', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg5 on Tue Aug 16 06:20:31 2011'}, {'table': 'iprange', 'name': 'barracuda Fw', 'ipaddr': '', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': '', 'bits': '', 'begin': '10.10.11.41', 'end': '10.10.11.46', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg15 on Wed Mar 1 19:58:04 2017'}, {'table': 'netgroup', 'name': 'admins', 'ipaddr': '', 'subnet': '', 'domain': '', 'host': '', 'ipaddrs': '', 'dns_mode': '', 'ttl': '', 'members': 'ipaddr:192.168.1.201,ipaddr:192.168.1.221', 'bits': '', 'begin': '', 'end': '', 'time_intervals': '', 'description': '', 'last_changed_by': 'cg15 on Mon May 7 11:25:39 2018'}]]

```

Obrázek 79: Skript2 - kontrolní výpis objektů z CSV souboru (Zdroj: Vlastní zpracování)

Po odeslání *request* firewallu, vrátí firewall *status kód*, který říká, co se s objektem stalo. Kódy 2xx značí úspěšné doručení, 3xx říká, že musí dojít ještě k další úpravě požadavku, 4xx naopak hlásí chybu na straně klienta a 5xx na straně serveru a 1xx jsou pouze informativní. Společně s výpisem odesílaného *payload* tvoří výpis *status kódu* dobrý základ pro případné opravy, jestliže se objekt nepodaří vytvořit. Na obrázku níže je možné vidět výpis z terminálu a *kód 204*, což znamená *úspěšné* vytvoření objektu.


```

{ 'comment': '', 'dnslifetime': 0, 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [], 'name': 'www.seznam.cz', 'type': 'hostname' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '192.168.1.201' } }], 'name': 'admin1', 'type': 'generic' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '192.168.1.221' } }], 'name': 'admin2', 'type': 'generic' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '8.8.8.8' } }], 'name': 'Google-DNS', 'type': 'generic' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '173.194.0.0/16' } }], 'name': 'google-173.194.0.0', 'type': 'singleIPv4Network' }
204
{ 'comment': 'hello', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '172.217.32.0/20' } }], 'name': 'google-172.217.32.0', 'type': 'singleIPv4Network' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '10.10.100.10-20' } }], 'name': 'pc-net', 'type': 'generic' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'entry': { 'ip': '10.10.11.41-46' } }], 'name': 'barracuda_FWs', 'type': 'generic' }
204
{ 'comment': '', 'excluded': [], 'geo': { 'excluded': [], 'included': [] }, 'included': [{ 'references': 'admin1', 'references': 'admin2', 'references': 'Google-DNS' }], 'name': 'admins', 'type': 'generic' }
204

```

Obrázek 80: Skript2 - návrat status kódu po každém požadavku (Zdroj: Vlastní zpracování)

Výsledek migrace je možné také zkontrolovat přímo na firewallu, protože by se zde již měly nacházet migrované objekty. Kontrolu provedeme v panelu *CONFIGURATION*, *Configuration Tree*, *Box (HA Cluster)*, *Assigned Services*, *NGFW (Firewall)*, *Forwarding Rules* a na levé straně vybereme *Networks* pod seznamem *Firewall Objects*. Na obrázku níže jsou všechny síťové objekty, označeny jsou právě *migrované*. Migrace tedy proběhla úspěšně.

Name	References	Entries	Comment	Usage
DYNAMIC				
STATIC				
admin1	1	192.168.1.201		
admin2	1	192.168.1.221		
admins	0	Ref: admin1, Ref: admin2, Ref: Google-DNS		
Any	40	0.0.0.0/0	All IPv4 addresses	
barracuda_FWs	0	10.10.11.41-46		
barracuda_FWs.com	1	DNS 0		
Evil Mode Bridged Ports	2	0.0.0.0/0 port1, 0.0.0.0/0 port2, 0.0.0.0/0 MGMT, 0.0.0.0/0 portA1, 0.0.0.0/0 p1, 0.0.0.0/0 p2		
google-172.217.32.0	0	172.217.32.0/20	hello	
google-173.194.0.0	0	173.194.0.0/16		
Google DNS	1	8.8.8.8		
Internet	1	Ref: Any, NOT 10.0.0.0/8, NOT 172.16.0.0/12, NOT 169.254.0.0/16, NOT 192.168.0.0/16	All routed IPv4 addresses	
pcnet	0	10.10.100.10-20		
Private 10	1	10.0.0.0/8	Private class A network	
Private 172	1	172.16.0.0/12	16 private class B networks	
Private 192	1	192.168.0.0/16	Private class B network	
Private IPv4 Addresses	1	Ref: Private 10, Ref: Private 172, Ref: Private 192	All private networks defined in RFC 19.	
Trusted LAN	14	Ref: Trusted LAN Networks, Ref: Trusted Next-Hop Networks	All network addresses that have a trust.	
VPN-Local-Networks	1	0.0.0.0/0	All locally defined networks for Site-2-S.	
VPN-Remote-Networks	7	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All networks reachable through VPN at...	
VPN-Remote-Networks	1	0.0.0.0/0 vpn0, 0.0.0.0/0 vpn0	All defined remote networks for Site-2...	
www.seznam.cz	0	DNS 0		

Obrázek 81: Výsledek migrace pomocí REST API uvnitř Barracuda CGFW (Zdroj: Vlastní zpracování)

3.4.3 Zhodnocení migrace objektů

Po skončení *ostré migrace* bylo ve výpisu z příkazové řádky Python zkontrolováno, jestli byly importované všechny objekty. Nebylo v plánu migrovat *IPv6 adresy* a jiné *IPv6 objekty*, proto u nich bylo možné vidět *chybu 400*, což značí špatnou *syntaxi*, jelikož nebyl změněn typ objektu na *singleIPv6Address* během jejich vkládání.

Dále zde byl objeven také výskyt chyby při vkládání *referencí*, u některých *netgroups*. Tyto nezapsané objekty byly přidány ručně, což nebylo obtížné, jelikož se jednalo o pár objektů. Krom těchto pár chyb proběhla migrace úspěšně a všech téměř *500 síťových objektů* bylo migrováno na nový firewall v rámci hodiny, kde nyní mohly být použity při tvorbě *Access Rules*.

3.5 Migrace Access Rules

Pravidla oproti síťovým objektům byla migrována ručně, a to z několika důvodů. První z důvodů bylo vyřazení nepoužívaných pravidel. Na starém FW se vlivem dlouhého používání, obměnou firemních softwarů a střídáním různých správců FW a jiných zaměstnanců vytvořilo spoustu pravidel, která již nejsou používána a pouze sťažují administrativu, proto se během postupného zpracování, což většinou probíhalo po oddílech, odstranila ta pravidla, které se již nepoužívají a zbylá pravidla byla následně migrována na nový FW.

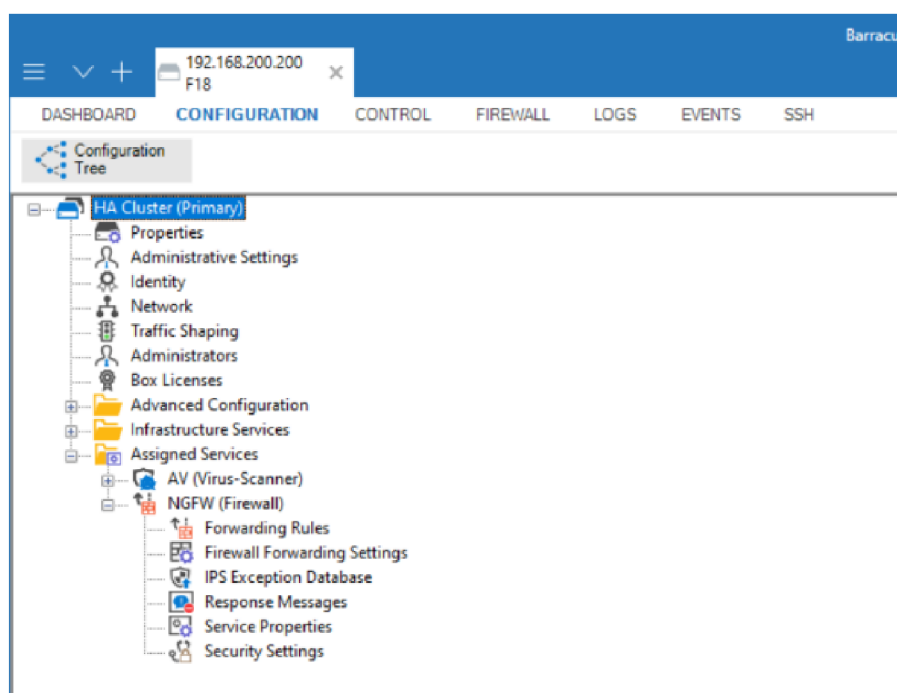
Dalším z důvodů bylo sjednocení názvosloví. Jak již bylo zmíněno stávající pravidla byla vytvořena více administrátory za dobu používání firewallu a ve společnosti neexistoval žádný formát, kterým by se měli správci řídit, během jejich pojmenovávání, proto postupné procházení umožnilo tuto situaci napravit a názvosloví sjednotit.

Mimo jiné se v *Access Rules* používají kromě *síťových objektů* také *objekty služby*, které umožňují specifikovat *source* i *destination porty a protokol*, který může komunikace nebo služba využívat. Bylo rozhodnuto, že tyto objekty se přidají ručně, jelikož je také potřeba pročistit nepotřebné a redundantní *objekty služby*.

V neposlední řadě jednotlivé přidávání pravidel umožňuje jejich současnou kontrolu správnosti. Toto by nebylo možné, kdyby byla přidávána pomocí skriptu a muselo by se do nich poté nahlížet stejně individuálně po migraci.

3.5.1 Postup přidání nového pravidla

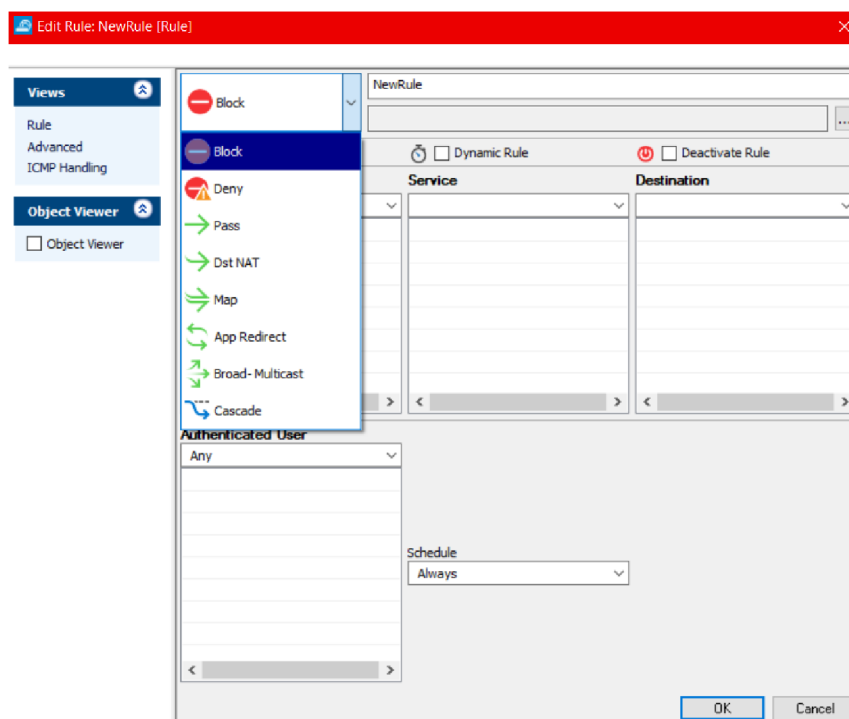
Dále bude popsán postup, jak vytvořit a přidat nová pravidla a také *objekty služby*. K *Access Rules* se dostaneme z panelu *CONFIGURATION > Configuration Tree > HA Cluster (Box) > Assigned Services > NGFW (Firewall) > Forwarding Rules*.



Obrázek 82: Konfigurační strom - Forwarding Rules (Zdroj: Vlastní zpracování)

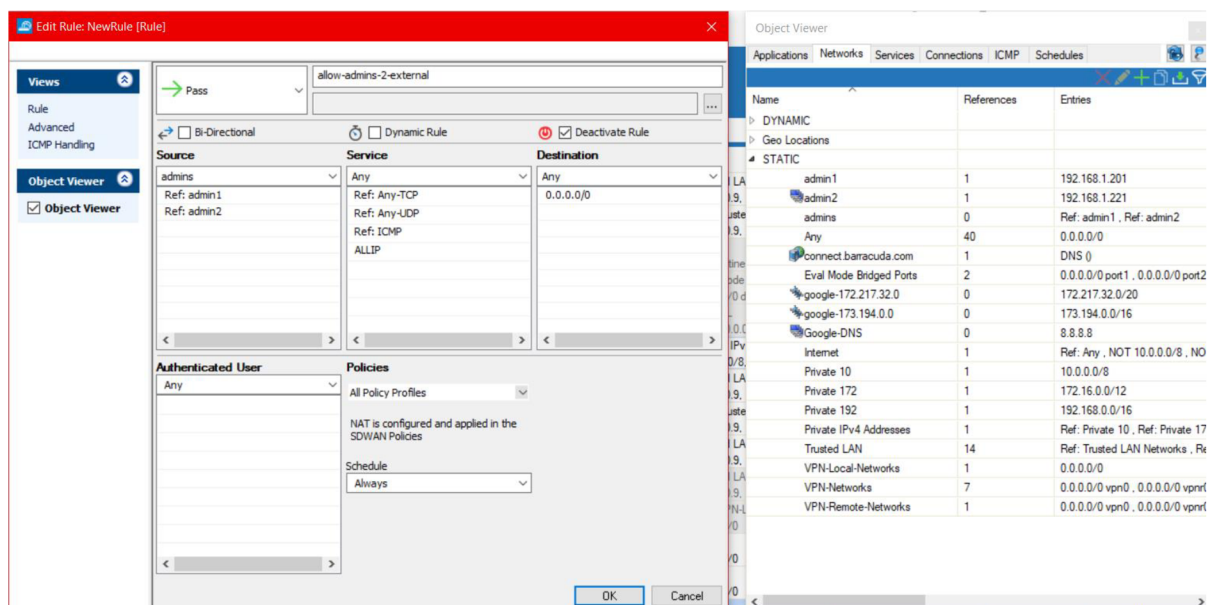
Na obrázku je možné vidět seznam všech *Access Rules*, která jsou nyní na FW přidána. Pro přidání nového pravidla je potřeba pravidla uzamknout tlačítkem *Lock* a kliknout na znaménko zeleného plus nebo kliknutím pravého tlačítka myši kdekoli v pravidlech a zvolením *New > Rule*.

Začneme volbou typu pravidla, což bude pro ukázkou *Pass*. V 90 % procentech případů se bude jednat o pravidla typu *Pass*, protože je jednodušší veškerou komunikaci zakázat pravidlem *BlockAll* a až zpětně povolovat komunikaci dle potřeby pravidly *Pass*. Pravidlo *BlockAll* je automaticky předdefinované na Barracuda CloudGen Firewallu a nachází se úplně na konci listu pravidel kde, jestliže komunikace neprojde žádným předchozím pravidlem, ukončí komunikaci a nikam nedorazí.



Obrázek 85: Editor pravidel - typy pravidel (Zdroj: Vlastní zpracování)

Na obrázku níže je možné vidět vyplněné pravidlo s použitím síťového objektu *admins*. Toto pravidlo umožňuje adminům připojení k externí síti. Pro přidání síťového objektu je možné jej najít v rozbalovacím seznamu nebo na levé straně editoru zaškrtnout *Object Viewer*. Otevře se tak seznam všech síťových objektů, přidanych i defaultních. Objekt *admins* můžeme jednoduše přetáhnout do pole *Source*. Jako *best practice* se také doporučuje pravidlo *deaktivovat*, aby nedošlo k odepření připojení ke správě firewallu před jeho otestováním, což je možné zaškrtnutím *Deactivate Rule*. Přidání nového pravidla následně potvrdíme tlačítkem *OK*.



Obrázek 86: Editor pravidel - využití síťových objektů (Zdroj: Vlastní zpracování)

Takto vytvořené pravidlo se nyní zobrazí v seznamu pravidel jako na obrázku č. 87 níže. Je možné jej ještě znovu otevřít a editovat nebo posunout v hierarchii pravidel. Veškerá zašedlá pravidla jsou deaktivovaná pravidla. Pravidlo odešleme firewallu kliknutím na tlačítko *Send Changes*.

Section 1 (14)													
1	 Pass SD-WAN	allow-admins-2-external	All Policy Pr...	N.A.	Any ALLIP, ECHO, TCP *, TCP...	admins 192.168.1.201, 192.168.1.2	Any 0.0.0.0/0	Any	Always	N			
2	 App Redirect 127.0.0.1:8080	RestAPI-Access	N.A.	N.A.	TCP 8080	Trusted LAN 127.0.0.9, 192.168.200.0/24	Management IP 192.168.200.200	Any	Always	No-S			
3	 Dynamic NAT	BOX-BARRACUDA-CUD...	None	Explicit NAT	ScreenConnect TCP 443, TCP 8040, TCP ...	Ref: Trusted LAN, Ref: WI... 127.0.0.9, 192.168.200.0/24	connect.barracuda.com	Any	Always	High			
4	 Block	BlockATDQuarantine		N.A.	N.A.	Any ALLIP, ECHO, TCP *, TCP...	Quarantine 127.0.0.0/0	Any	Always	N.A.			
5	 SD-WAN	BOX-EVAL-MODE-BRIDGE	All Policy Pr...	N.A.	Any ALLIP, ECHO, TCP *, TCP...	Eval Mode Bridged Ports 0.0.0.0/0 dev MGMT, 0.0.0...	Eval Mode Bridged Ports 0.0.0.0/0 dev MGMT, 0.0.0...	Any	Always	N.A.			
6	 Dat NAT Primary Management IP, O...	BOX-SERVICEIP-MGMT...	None	Explicit NAT	Ref: HTTPS, Ref: NGF-MG... ECHO TCP 22, TCP 443...	BoxACL 0.0.0.0/0	Service IPs 127.0.0.9	Any	Always	No-S			
7	 App Redirect 192.168.200.200	BOX-SETUP-MGMT-ACC...		N.A.	N.A.	Ref: NGF-MGMT-BOX, Ref... ECHO TCP 22, TCP 443...	Private IPv4 Addresses 10.0.0.0/8, 172.16.0.0/12...	DHCP1 Local IP	Any	Always	High		

Obrázek 87: Nově vytvořené pravidlo (Zdroj: Vlastní zpracování)

Poté klikneme na tlačítko *Activate* a zobrazí se vyskakovací okno. Můžeme aktivaci zrušit, potvrdit, nebo potvrdit a ponechat zámek u pravidel pro další úpravy. Ponecháme zámek a ukážeme si, jak pracovat s *objekty služby*.

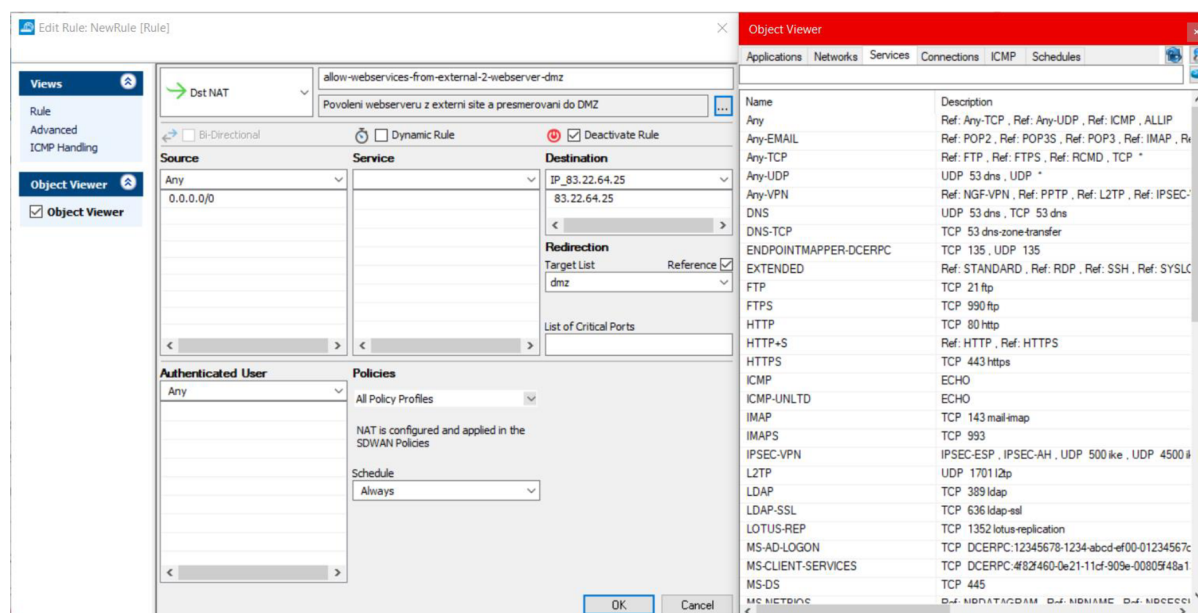


Obrázek 88: Aktivace po vytvoření pravidla (Zdroj: Vlastní zpracování)

Vytvoření Dst NAT pravidla a využití objektů služby

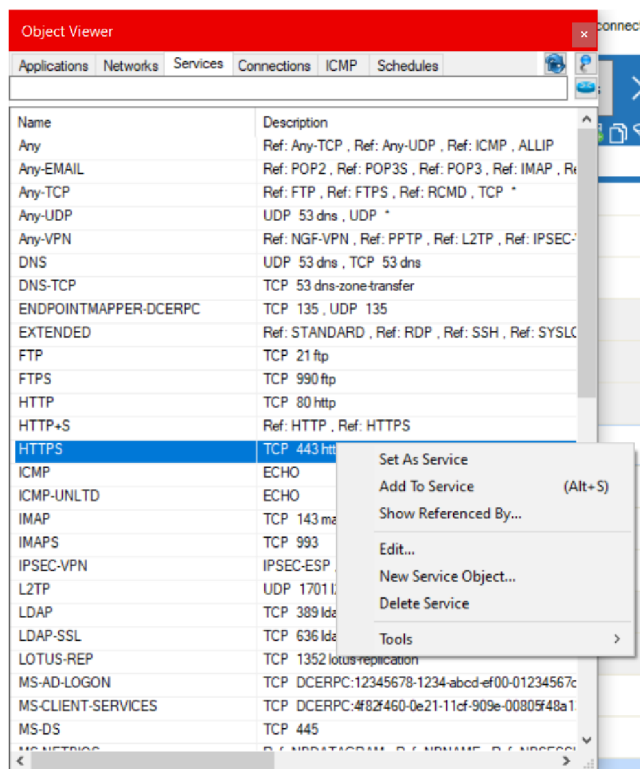
V rámci editoru pravidel se pro kritérium *Service* mohou použít také *objekty služby*, což si nyní předvedeme na příkladu při vytvoření nového pravidla *allow-webservices-from-external-2-webserver-dmz*.

Nově vytvořené pravidlo je *Dst NAT* pravidlo, v kritérii *Source* jsme zvolili *any*, což znamená, že komunikace z jakéhokoliv externího zdroje přicházející na IP adresu 83.22.64.25 bude přesměrována do *demilitarizované zóny*, což je síťový objekt *dmz*. Je ještě potřeba definovat *Service*. Na pravé straně obrázku je možné vidět *Object Viewer*, kde nahoře je zvolena záložka *Services*. Zobrazí se nám tak všechny dostupné služby jejich protokoly. Pravidlo chceme definovat pro protokol *TCP* na *portu 8080 a 8010*, a protože v defaultním seznamu nejsou na výběr, budeme tyto objekty muset *vytvořit*.



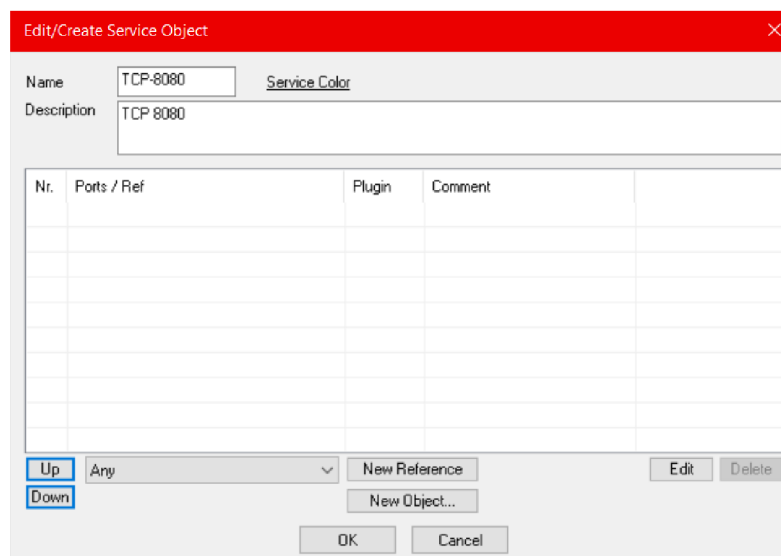
Obrázek 89: Editor pravidel - tvorba Dst NAT pravidla s použitím Service objektů (Zdroj: Vlastní zpracování)

Začneme kliknutím pravého tlačítka kamkoliv do seznamu objektů služby a zvolíme možnost *New Service Object*, což nám otevře *editor* objektů služby.



Obrázek 90: Vytvoření nového service objektu – New Service Object... (Zdroj: Vlastní zpracování)

V nově otevřeném editoru vyplníme v první řadě *název* a *popis* objektu. Zadané hodnoty jsou *TCP-8080* a *TCP 8080*. Nyní musíme přidat protokol a port, což uděláme kliknutím na tlačítko *New Object*.



Obrázek 91: Editor service objektů – název (Zdroj: Vlastní zpracování)

Po kliknutí na tlačítko se otevře menu pro přidání parametrů. Nás bude zajímat pouze *IP Protocol* a *Port Range*. Z rozbalovacího seznamu *IP Protocol* vybereme možnost *006 TCP* a do pole *Port Range* vepíšeme *8080*, což je číslo našeho portu. Je možné zde také vepsat *rozsah portů*, jako třeba *8080-8090*, což bude zahrnovat všechny porty mezi 8080 a 8090 včetně. Po vyplnění stiskneme tlačítko *OK* pro přidání.

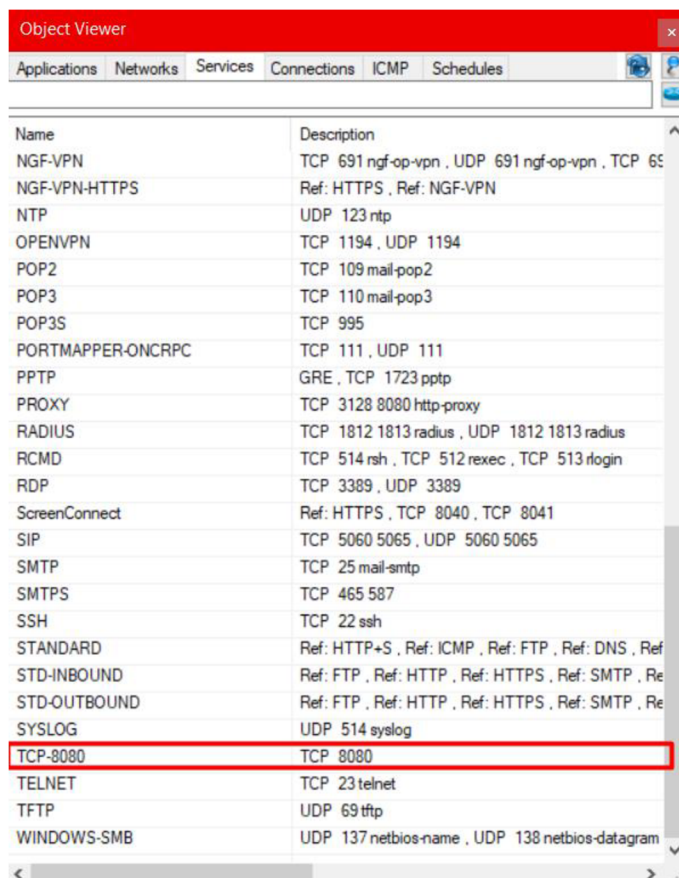
Obrázek 92: Editor service objektů - volba protokolu a portu (Zdroj: Vlastní zpracování)

Nově vytvořený objekt by teď měl být přidán v seznamu v editoru. Pro vytvoření *objektu služby* potvrdíme stisknutím *OK*.

Nr.	Ports / Ref	Plugin	Comment
01	TCP 8080		

Obrázek 93: Editor service objektů - přidání protokolu a portu (Zdroj: Vlastní zpracování)

Vytvořený *objekt služby* je nyní k dispozici v seznamu *Services* a můžeme jej použít při tvorbě pravidel. Nově vytvořený *objekt služby* je možné vidět na obrázku č. 94 níže.



Obrázek 94: Object Viewer - nově vytvořený service objekt (Zdroj: Vlastní zpracování)

Druhý způsob, jak přidat port a protokol je pomocí *reference* již existujících *objektů služby*. Můžeme si to vyzkoušet na následujícím příkladu, kde chceme vytvořit *objekt služby* s protokolem *TCP* a *porty* 8010 a 8080, pojmenujeme ho proto *TCP-8010-8080*. Nyní přidáme způsobem jako předtím *protokol TCP* na *portu* 8010. Objekt *TCP-8080* máme ale již vytvořený, proto můžeme ve spodním seznamu najít tento objekt a zmáčknout tlačítko *New Reference*, což objekt vloží jako *referenci*. Nyní změny potvrdíme tlačítkem *OK* a přidáme nový *objekt služby*.

Dialog box titled "Edit/Create Service Object".

Name: TCP-8010-8080 Service Color

Description: TCP 8010 and 8080

Nr.	Ports / Ref	Plugin	Comment
01	TCP 8010		
02	Ref: TCP 8080		

Buttons: Up, Down, TCP 8080 (dropdown), New Reference, Edit, Delete, New Object..., OK, Cancel.

Obrázek 95: Editor service objektů - reference (Zdroj: Vlastní zpracování)

Takto vytvořený objekt můžeme nyní vybrat z rozbalovacího seznamu nebo přetáhnout ze seznamu *Services* do nově tvořeného pravidla. Pravidlo potvrdíme tlačítkem *OK*, odešleme změny pomocí *Send Changes* a aktivujeme tlačítkem *Activate*.

Dialog box titled "Edit Rule: NewRule [Rule]".

Views: Rule, Advanced, ICMP Handling

Object Viewer: ☒ Object Viewer

Rule: Dst NAT allow-webservices-from-external-2-webserver-dmz

Povoleni webserveru z externi site a presmerovani do DMZ

BI-Directional: ☐ Dynamic Rule: ☐ Deactivate Rule: ☒

Source: Any Service: TCP-8010-8080 (selected: TCP 8010, Ref: TCP 8080) Destination: IP_83.22.64.25, 83.22.64.25

Redirection: Target List: dmz, Reference: ☒

Authenticated User: Any

Policies: All Policy Profiles, Schedule: Always

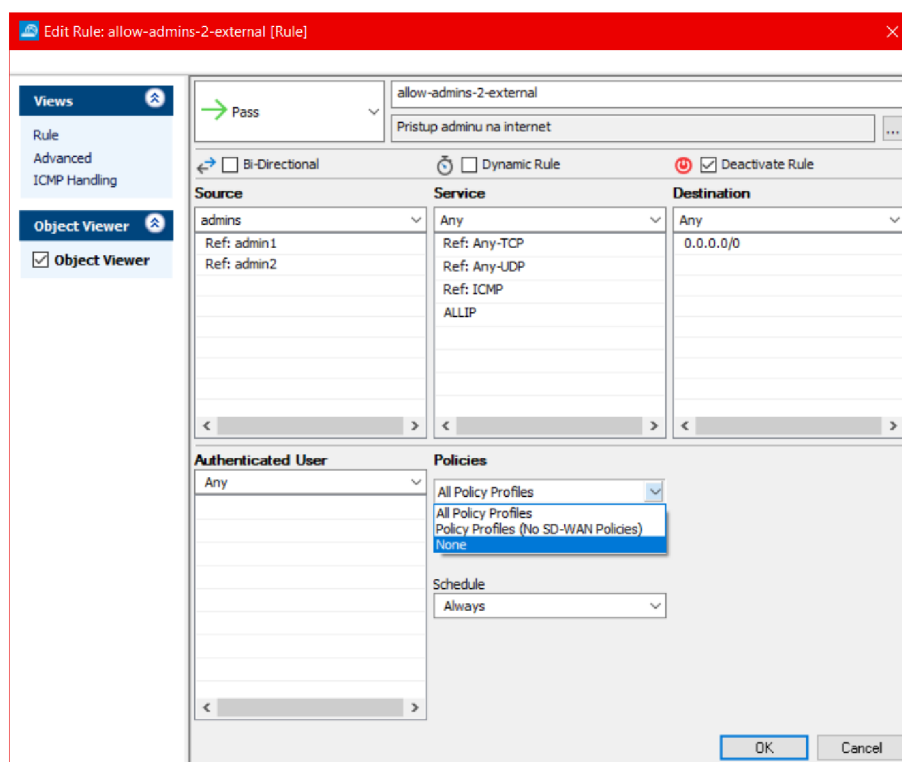
Buttons: OK, Cancel

Obrázek 96: Využití service objektu v novém pravidlu (Zdroj: Vlastní zpracování)

Přidání Source NAT

Další možné nastavení u pravidla je *Source NAT*, tedy překlad privátní IP adresy na veřejnou, aby zařízení v síti mohlo komunikovat s vnější sítí. Máme zde tři možnosti, buď nebude *Source NAT* aplikováno vůbec, aplikuje ho automaticky router nebo firewall, anebo zvolíme explicitní adresu.

Abychom si to předvedli, můžeme použít již vytvořené pravidlo *allow-admins-2-external*. V první řadě je potřeba opět uzamknout všechny pravidla, poté stačí kliknout pravým tlačítkem na zmíněné pravidlo a zvolit *Edit Rule*. V nastavení začneme změnou *Policies* na *None*, tak jako na obrázku č. 97 níže.

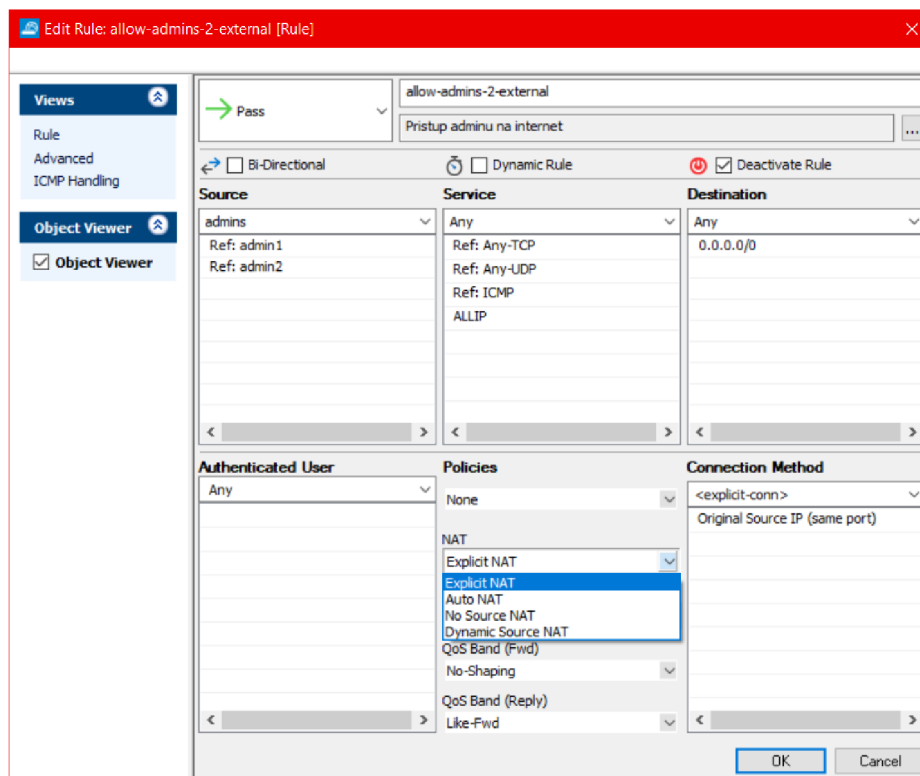


Obrázek 97: Nastavení Source NAT - volba Policies (Zdroj: Vlastní zpracování)

Nyní se objeví nová možnost nastavení, *NAT*. Zde ze seznamu můžeme vybrat potřebný typ NAT. Pro praktický příklad zvolíme *Explicit NAT*, tedy bude vkládat konkrétní adresu, kterou chceme, aby pravidlo použilo při překladu, což můžeme vidět na obrázku níže. Další možnosti zahrnují:

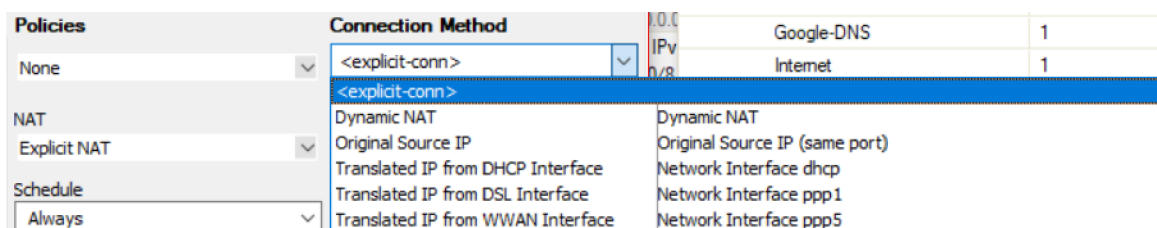
- **Auto NAT** – přiřadí odchozí komunikaci automatickou veřejnou IP adresu

- **No Source NAT** – komunikaci nebude přidělena žádná veřejná IP adresa, většinou pro interní komunikaci
- **Dynamic Source NAT** – komunikaci je přidělena veřejná IP adresa ze seznamu dostupných pro přidělení



Obrázek 98: Vytvoření Source NAT - volba typu Source NAT (Zdroj: Vlastní zpracování)

V pravé části nastavení je nyní potřeba zvolit z rozbalovacího seznamu *<explicit-conn>* a poté kliknout pravým tlačítkem do volného pole pod rozbalovacím seznamem a zvolíme možnost *Edit...*



Obrázek 99: Vytvoření Source NAT - explicitní Source NAT (Zdroj: Vlastní zpracování)

Dostaneme se tak do nastavení, kde v druhé části můžeme vidět *NAT Settings*. Zde z rozbalovacího seznamu u *Translated Source IP* zvolíme *Explicit IP*. V rámci migrace síťových objektů byly všechny *IP adresy* vloženy jako *Generic typ* síťového objektu, ale

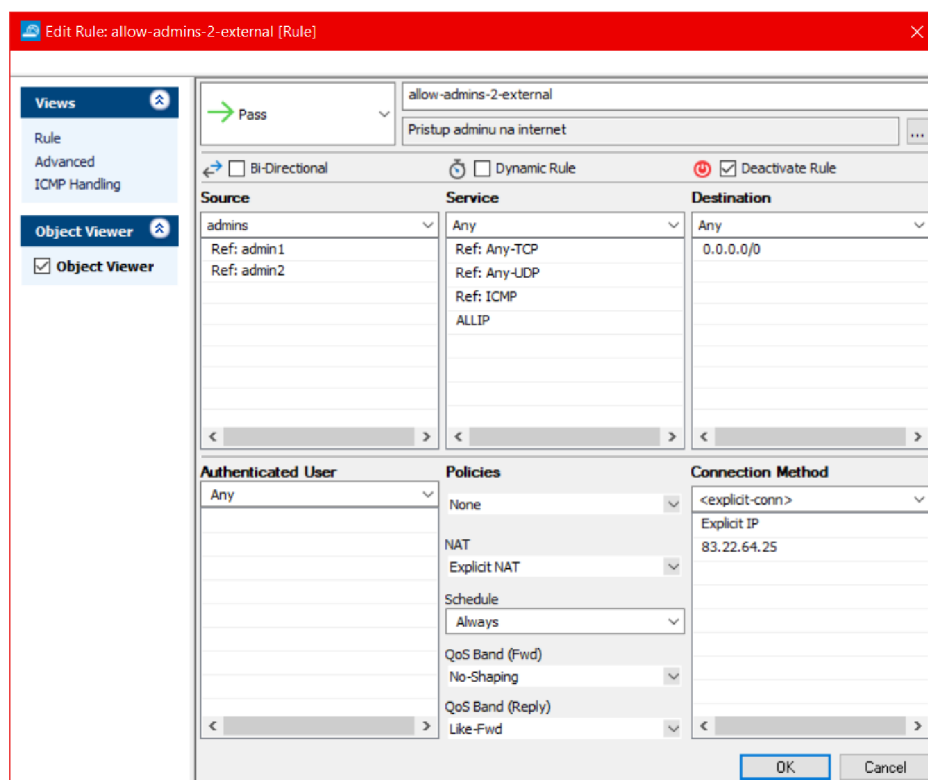
pokud by byly vloženy jako *Single IPv4 Address*, mohli bychom zvolit z rozbalovacího seznamu možnost *Single IP Network Object*, což by usnadnilo práci.

Obrázek 100: Vytvoření Source NAT – editor (Zdroj: Vlastní zpracování)

Po zvolení *Explicit IP*, můžeme do příslušné kolonky vepsat konkrétní veřejnou IP adresu, na kterou chceme, aby probíhal překlad, například 83.22.64.25 a potvrdíme tlačítkem *OK*.

Obrázek 101: Vytvoření Source NAT - vyplněná explicitní IP (Zdroj: Vlastní zpracování)

Tato explicitní IP adresa se nyní zobrazí v editoru pravidla (viz obrázek č. 102 níže) a můžeme potvrdit úpravu pravidla stisknutím tlačítka *OK*.



Obrázek 102: Nové pravidlo s explicitním Source NAT (Zdroj: Vlastní zpracování)

Přidání sekce

S postupným přibýváním pravidel začne být složitější se v nich orientovat, proto je možné pravidla rozdělit do *sekcí*. Toto rozdělení nemá efekt na funkčnost pravidel a je zde pouze z organizačního důvodu. Nyní si ukážeme, jak vytvořit *sekcí* podobnou té, která je na obrázku níže, *Basic LAN and VPN Rules*. Pro vytvoření nové sekce je opět potřeba uzamknout pravidla tlačítkem *Lock* a následně stisknout tlačítko *Insert Section* nebo po kliknutí pravého tlačítka myši vybrat *New > Section*.

Action	Name	Features	Policy Usage	NAT Mode	Service	Source	Destination	User	Sche...	Go S
Pass	allow-admins-2-external		All Policy Pr...	N.A.	Any	192.168.1.201, 192.168.1.2...	Any	Any	Always	N.A.
Pass	allow-webservices-from...		All Policy Pr...	N.A.	TCP 8010-8080	Any	IP_83.22.64.25	Any	Always	N.A.
App Redirect	RestAPI-Access		N.A.	N.A.	TCP 8080	Trusted LAN	Management IP	Any	Always	No-Shaping
Pass	BOX-BARRACUDA-CUD...		None	Explicit NAT	ScreenConnect	Ref: Trusted LAN, Ref: WI...	connect.barracuda.com	Any	Always	High (ID 2)
Block	BlockATDQuarantine		N.A.	N.A.	Any	ATD	Any	Any	Always	N.A.
Pass	BOX-EVAL-MODE-BRIDGE		All Policy Pr...	N.A.	Any	Eval Mode Bridged Ports	Eval Mode Bridged Ports	Any	Always	N.A.
Pass	BOX-SERVICEIP-MGMT...		None	Explicit NAT	Ref: HTTPS, Ref: NGF-MG...	BoxACL	Service IPs	Any	Always	No-Shaping
App Redirect	BOX-SETUP-MGMT-ACC...		N.A.	N.A.	Ref: NGF-MGMT-BOX, Ref...	Private IPv4 Addresses	DHCP1	Any	Always	High (ID 2)
App Redirect	BOX-LOCALDNSCACHE		N.A.	N.A.	DNS	Trusted LAN	Local IP	Any	Always	Realtime (ID 1)
Basic LAN and VPN Rules (6)										
Pass	BOX-LAN-2-INTERNET		All Policy Pr...	N.A.	Any	Ref: Trusted LAN	Ref: Internet	Any	Always	N.A.
Pass	BOX-LAN-2-LAN		All Policy Pr...	N.A.	Any	Trusted LAN	Trusted LAN	Any	Always	N.A.
Pass	BOX-LAN-2-VPN-SITE		All Policy Pr...	N.A.	Any	Trusted LAN	VPN-Networks	Any	Always	N.A.
Pass	BOX-VPN-SITE-2-SITE		All Policy Pr...	N.A.	Any	Ref: VPN-Local-Networks	Ref: VPN-Remote-Networks	Any	Always	N.A.
Pass	BOX-VPNCLIENTS-2-LAN		All Policy Pr...	N.A.	Any	Any	Trusted LAN	Any	Always	N.A.
Block	BLOCKALL		N.A.	N.A.	Any	Any	Any	Any	Always	N.A.

Obrázek 103: Přidání sekce - přehled všech pravidel (Zdroj: Vlastní zpracování)

Po vytvoření nové sekce můžeme změnit její název pomocí symbolu žluté tužky. Název změníme prozatím na *Migrated FW Rules* a stiskneme *Enter*, následně ji můžeme posunout v seznamu nahoru nebo dolů pomocí *šipek* vpravo nahore nebo podržením a *přetáhnutím* sekce. Po odeslání změn firewallu tlačítkem *Send Changes* a aktivací pomocí *Activate* máme nově vytvořenou *sekcí*.

Počet sekcí, které je možné vytvořit, je *neomezený* a je pouze na administrátorovi, kolik si jich bude chtít přidat pro svou přehlednost. Dalším detailem je *číslo v závorce* za názvem sekce, které udává *počet pravidel* uvnitř sekce.

Action	Name	Features	Policy Usage	NAT Mode	Service	Source	Destination	User
Pass	allow-admins-2-external		All Policy Pr...	N.A.	Any	192.168.1.201, 192.168.1.2...	Any	Any
Pass	allow-webservices-from...		All Policy Pr...	N.A.	TCP 8010-8080	Any	IP_83.22.64.25	Any
App Redirect	RestAPI-Access		N.A.	N.A.	TCP 8080	Trusted LAN	Management IP	Any
Pass	BOX-BARRACUDA-CUD...		None	Explicit NAT	ScreenConnect	Ref: Trusted LAN, Ref: WI...	connect.barracuda.com	Any
Block	BlockATDQuarantine		N.A.	N.A.	Any	ATD	Any	Any
Pass	BOX-EVAL-MODE-BRIDGE		All Policy Pr...	N.A.	Any	Eval Mode Bridged Ports	Eval Mode Bridged Ports	Any
Pass	BOX-SERVICEIP-MGMT...		None	Explicit NAT	Ref: HTTPS, Ref: NGF-MG...	BoxACL	Service IPs	Any
App Redirect	BOX-SETUP-MGMT-ACC...		N.A.	N.A.	Ref: NGF-MGMT-BOX, Ref...	Private IPv4 Addresses	DHCP1	Any
App Redirect	BOX-LOCALDNSCACHE		N.A.	N.A.	DNS	Trusted LAN	Local IP	Any

Obrázek 104: Přidání sekce - nová sekce Migrated FW Rules (Zdroj: Vlastní zpracování)

3.5.2 Závěr k vkládání pravidel

Tímto způsobem bylo přidáno zhruba *200 různých pravidel*, která byla během vkládání kontrolována pro správnost. Během migrace pravidel nedošlo k žádným problémům a byly předána pro následnou kontrolu a úpravu, aby byly případně vyřazeny některé další pravidla, která se už nepoužívají a nebyla vyřazena před samotnou migrací.

Do budoucna je pro zlepšení možné vytvořit automaticky i *objekty služby* pomocí skriptu, ale v rámci čištění byly vkládány také ručně. Jediný nedostatek byl zvolený typ objektu pro IP adresy. Jak již bylo zmíněno IP adresy byly vkládány jako *Generic typ* síťového objektu, a tudíž nemohly být použity u *Source NAT*. V tomto případě by musely být typu *Single IPv4 Address*, avšak nízké množství těchto pravidel znamenalo, že tento nedostatek neměl *velký dopad*.

Migrace tedy může být zhodnocena jako *úspěšná* a ve společnosti mohou začít přípravy k bezpečnému přepojení firewallů.

3.6 Finanční zhodnocení

3.6.1 Pořizovací náklady

Abychom určili *TCO*, je v první řadě nutné spočítat *pořizovací náklady*. Níže je uvedena tabulka se všemi identifikovanými pořizovacími náklady. Spadají sem položky jako je *samotný HW, licence*, ale také *náklady spojené s implementační činností a náklady na školení technického pracovníka*. Celková částka pořizovacích nákladů činí **530 995 Kč**.

Tabulka 18: Tabulka pořizovacích nákladů (Zdroj: Vlastní zpracování)

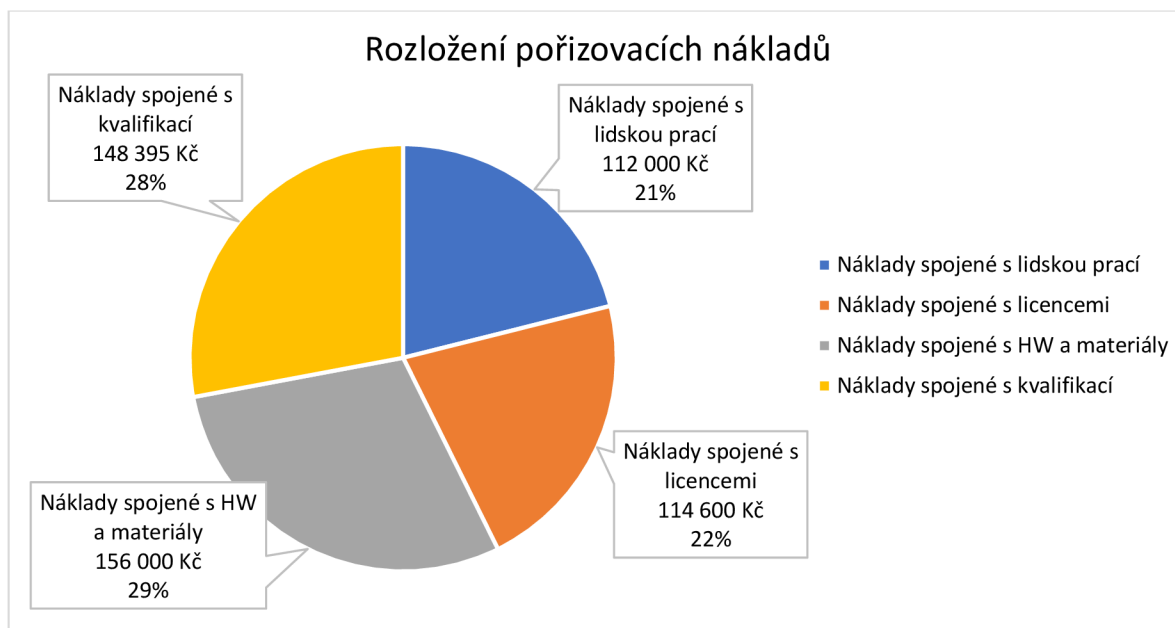
Pořizovací náklady			
Položka	Počet jednotek (ks)	Cena za jednotku (Kč)	Cena celkem (Kč)
Barracuda CloudGen Firewall Appliance F180	2	77 500	155 000
Barracuda CloudGen Firewall Appliance F18B	1	26 250	26 250
Barracuda CloudGen Firewall Training – CGF01	1	56 250	56 250
Barracuda CloudGen Firewall Training – CGF03/04	1	56 250	56 250
Implementační náklady (migrace pravidel + přepojení) (v md)	8	14 000	112 000
Barracuda CloudGen Firewall Appliance F180 – Advanced Threat Protection Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Energize Updates Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Instant Replacement Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Malware Protection Subscription (roční)	2	11 400	22 800

Barracuda CloudGen Firewall Appliance F18 – Advanced Threat Protection Subscription (měsíční)	6	472,5	2 835
Barracuda CloudGen Firewall Appliance F18 – Energize Updates Subscription (měsíční)	6	410	2 460
Barracuda CloudGen Firewall Appliance F18 – Instant Replacement Subscription (měsíční)	6	410	2 460
Barracuda CloudGen Firewall Appliance F18 – Malware Protection Subscription (měsíční)	6	315	1 890
Drobný instalační materiál	10	100	1 000
Celkem			530 995

Pořizovací náklady si můžeme pro zajímavost rozložit do čtyř kategorií a následně porovnat v grafu níže:

- **Náklady spojené s lidskou prací** – patří sem implementační činnost
- **Náklady spojené s licencemi** – platí pouze pro licence na Barracuda F180
- **Náklady spojené s HW a materiály** – samotné zařízení Barracuda F180 a drobné instalační materiály
- **Náklady spojené s kvalifikací** – spadá sem cena certifikačních kurzů, ale i cvičné zařízení Barracuda F18 spojené s platnými licencemi pro toto zařízení

Můžeme zde podotknout, že náklady, které se týkají kvalifikace technického pracovníka tvoří téměř čtvrtinu pořizovacích nákladů, můžeme tedy říct, že tuto kategorii pořizovacích nákladů je třeba *nepodceňovat*.



Graf 1: Rozložení pořizovacích nákladů (Zdroj: Vlastní zpracování)

3.6.2 Provozní náklady

Tabulka níže udává *provozní náklady na životní cyklus produktu* Barracuda CloudGen Firewall Appliance F180. Příkon jednoho firewallu je roven 45 W , při *nepřetržitém* chodu obou firewallů po celý rok nám vyjde spotřeba $788,4\text{ kWh}$, což můžeme poté vynásobit cenou $2,4\text{ Kč/kWh}$ a dostaneme finální hodnotu $1\,892\text{ Kč}$. Můžeme zde započítat menší *rezervu* pro chod klimatizačních jednotek a osvětlení v serverovně. (57; 61)

Dále jsou v tabulce zaznamenány částky potřebné pro obnovu licencí na obou zařízeních, což za každý rok činí $34\,200\text{ Kč}$ u každého. Firewally budou muset být navíc servisovány a společnost identifikovala, že tato údržba zabere 4 h měsíčně . Při sazbě $14\,000\text{ Kč/md}$, získáme částku $84\,000\text{ Kč}$ každý rok.

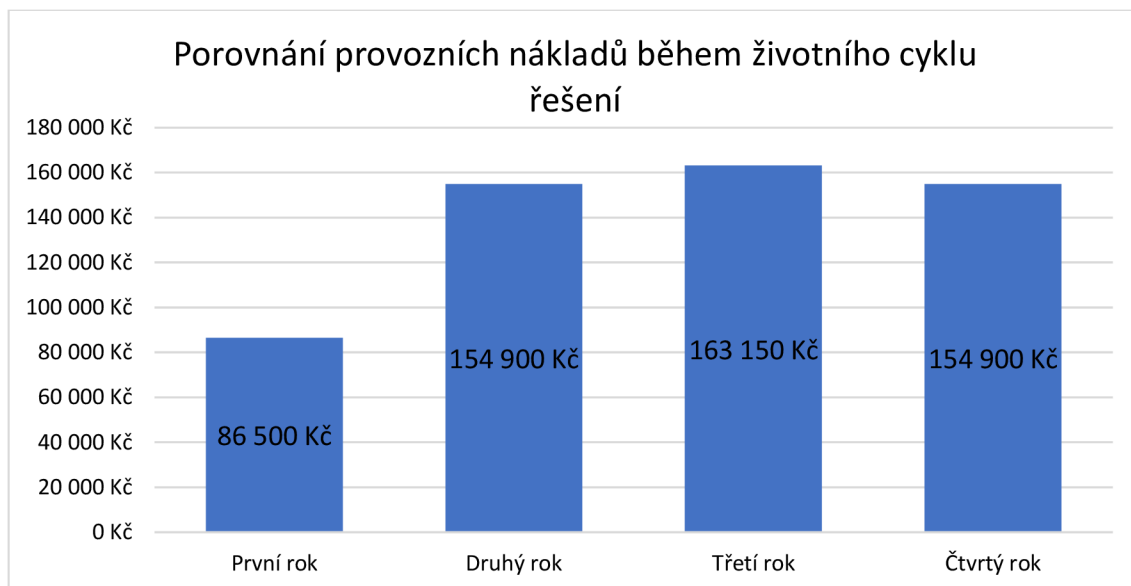
V neposlední řadě spadají do provozních nákladů také náklady na *recertifikační testování pracovníka* pověřeného správou firewallu. Certifikát má platnost *dva roky*, proto si jej musí pracovník obnovit na začátku třetího roku. Náklad na jeden recertifikační test činí $8\,250\text{ Kč}$.

Suma provozních nákladů za celou dobu životnosti řešení je odhadována na **559 450 Kč**.

Tabulka 19: Tabulka provozních nákladů za celou dobu životnosti produktu (Zdroj: Vlastní zpracování)

Provozní náklady				
Položka	První rok (Kč)	Druhý rok (Kč)	Třetí rok (Kč)	Čtvrtý rok (Kč)
Energetické náklady	2 500	2 500	2 500	2 500
Obnova licencí pro primární F180 (ATP, Energize Updates, Instant Replacement, Malware Protection)	-	34 200	34 200	34 200
Obnova licencí pro sekundární F180 (ATP, Energize Updates, Instant Replacement, Malware Protection)	-	34 200	34 200	34 200
Údržba (Servisní smlouva – 4 h měsíčně)	84 000	84 000	84 000	84 000
Recertifikační test	-	-	8 250	-
Celkem	86 500	154 900	163 150	154 900
Celkové provozní náklady za 4 roky	559 450			

Pro lepší vizualizaci můžeme provozní náklady v jednotlivých letech převést do grafu níže.



Graf 2: Porovnání provozních nákladů během životního cyklu produktu (Zdroj: Vlastní zpracování)

3.6.3 Náklady na obnovu

Dále je potřeba určit *náklady na obnovu*, které můžeme vidět v tabulce níže. Barracuda CloudGen Firewall Appliance F180 má *životnost 4 let*, protože společnost disponuje licencí Instant Replacement, což zaručuje automatickou výměnu zařízení na náklady výrobce po uplynutí 4 let. Můžeme zde uvést tedy položku zařízení s cenou 0 Kč/j, pro udržení konzistence. Instant Replacement tedy umožňuje společnosti, mimo jiné výhody, rozdělit si náklady za nové zařízení během let. (62)

Do činnosti spojené s výměnou spadá, příprava, zálohování, instalace, nahrání zálohy a následné odladění chyb. Vzhledem k tomu, že se jedná o stejný produkt, neměly by zde vzniknout žádné velké komplikace a celá činnost by měla zabrat *1 md*, při sazbě *14 000 Kč/md*.

V neposlední řadě je potřeba opět nakoupit *licence* pro nová zařízení, které budou stejné jako předchozí, proto je zde očekávaná stejná částka.

Celkové náklady na obnovu tak činní **128 600 Kč**.

Tabulka 20: Tabulka nákladů na obnovu (Zdroj: Vlastní zpracování)

Náklady na obnovu			
Položka	Počet jednotek (ks)	Cena za jednotku (Kč)	Cena celkem (Kč)
Obměna zařízení Barracuda CloudGen Firewall Appliance F180	2	0	0
Obměna zařízení (činnost) (v md)	1	14 000	14 000
Nákup nových licencí pro Barracuda CloudGen Firewall Appliance F180			-
Barracuda CloudGen Firewall Appliance F180 – Advanced Threat Protection Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Energize Updates Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Instant Replacement Subscription (roční)	2	15 300	30 600
Barracuda CloudGen Firewall Appliance F180 – Malware Protection Subscription (roční)	2	11 400	22 800
Celkem			128 600

3.6.4 Celkové náklady na vlastnictví

Na základě vypočítaných pořizovacích nákladů, provozních nákladů a nákladů na obnovu můžeme vypočítat *TCO*. V tabulce níže můžeme vidět *jednotlivé náklady*, *TCO* a *Průměrné TCO*. *TCO* je rovno **1 219 045 Kč**, což je celková částka, kterou bude

společnost stát vlastnictví tohoto produktu po dobu jeho životní cyklu. Jestliže si TCO rozložíme na dobu 4 let, získáme hodnotu **304 761 Kč/rok**.

Tabulka 21: TCO po dobu životního cyklu (Zdroj: Vlastní zpracování)

TCO po dobu životního cyklu (4 roky)	
Typ nákladu	Částka (Kč)
Pořizovací náklady	530 995
Provozní náklady	559 450
Náklady na obnovu	128 600
TCO	1 219 045
Průměrné TCO	304 761

3.6.5 Porovnání nákladů

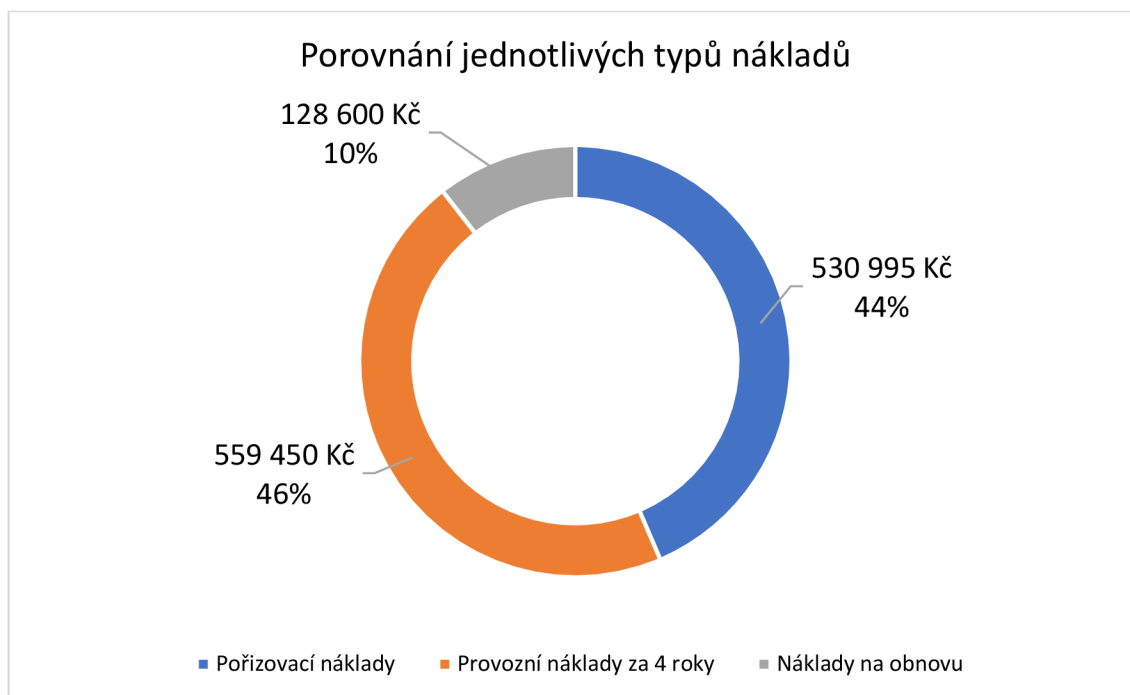
V tabulce níže můžeme vidět porovnání jednotlivých typů nákladů. Největší položkou zde jsou *provozní náklady*, které tvoří zhruba **50 %** celkových nákladů na vlastnictví produktu.

Je tedy důležité provést posouzení TCO, aby nedošlo k situaci, kdy si společnost pořídí nový produkt a neprovede jeho nasazení, protože si nespočítala správně všechny náklady spojené s jeho vlastnictvím.

Tabulka 22: Porovnání jednotlivých typů nákladů (Zdroj: Vlastní zpracování)

Porovnání jednotlivých typů nákladů			
Typ nákladů	Pořizovací náklady (Kč)	Provozní náklady za 4 roky (Kč)	Náklady na obnovu (Kč)
Částka	530 995	559 450	128 600

Pro lepší vizualizaci byl vytvořen graf porovnání jednotlivých typů nákladů, který je možné vidět níže. Jak již bylo zmíněno provozní náklady tvoří téměř polovinu celého TCO.



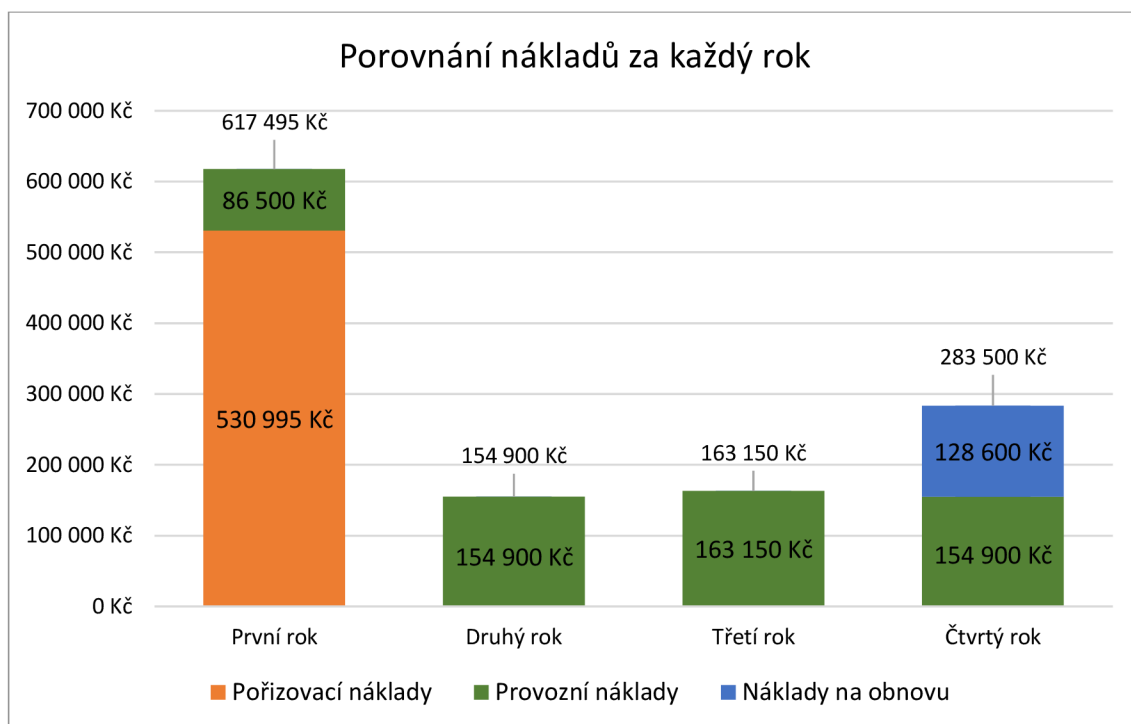
Graf 3: Porovnání jednotlivých typů nákladů (Zdroj: Vlastní zpracování)

Dále je možné porovnat náklady za *každý rok životnosti produktu*. V tabulce níže jsou uvedeny všechny typy nákladů a jejich suma za každý rok. Můžeme zde vidět, že nejvyšší náklady jsou v *prvním roce*, kdy většinu nákladů tvoří pořizovací náklady. Náklady ve čtvrtém roce jsou na druhém místě.

Tabulka 23: Porovnání nákladů za každý rok (Zdroj: Vlastní zpracování)

Porovnání nákladů za každý rok				
Typ nákladů	První rok (Kč)	Druhý rok (Kč)	Třetí rok (Kč)	Čtvrtý rok (Kč)
Pořizovací náklady	530 995	-	-	-
Provozní náklady	86 500	154 900	163 150	154 900
Náklady na obnovu	-	-	-	128 600
Celkem	617 495	154 900	163 150	283 500

Graf č. 4 níže dává do perspektivy rozložení těchto nákladů v každém roce. Opět můžeme vidět, že v prvním roce jsou náklady mnohem vyšší než jakýkoliv jiný rok.



Graf 4: Porovnání nákladů za každý rok (Zdroj: Vlastní zpracování)

ZÁVĚR

Diplomová práce se zabývala bezpečným a účinným procesem migrace firewallových politik ze stávajícího systému Forcepoint Sidewinder, který je již EoL produkt na řešení Barracuda CloudGen Firewall ve společnosti COMGUARD a.s. Cílem práce bylo identifikovat a analyzovat klíčové faktory, využít dostupné technologie a aplikovat prvky projektového managementu pro úspěšnou a bezproblémovou migraci.

V první části práce byly popsány teoretické základy firewallů, jeho typů, a především firewallů nové generace. Zároveň zde byl představen jejich zástupce Barracuda CloudGen Firewall a jeho podstatné funkce a nastavení potřebné pro projekt. Součástí teoretické části práce bylo také představení použitých technologií pro migraci, jako je Python a RESTful API a vysvětlení důležitých prvků projektového řízení, mezi které patří analýza rizik, časová analýza a analýza nákladů.

Druhá část práce se zabývala představením společnosti COMGUARD a.s., analýzou současného stavu ve společnosti a příčin, které vedly k potřebě vyměnit stávající firewall za nový a provést tak migraci jeho politik. Na základě této analýzy bylo vypracováno zadání pro samotný projekt migrace.

Stěžejní částí práce byl konkrétní návrh řešení migrace firewallových politik včetně vývoje skriptu v Pythonu a využití RESTful API krok po kroku. Byla zde provedena pečlivá analýza rizik a časového plánu projektu, a byly stanoveny odpovědnosti za jednotlivé fáze implementace. Takto navrhnuté a provedené řešení bylo následně finančně zhodnoceno v rámci celkových nákladů na vlastnictví produktu během jeho celého životního cyklu.

Výsledkem navrhovaného řešení je jasný plán, tvořený činnostmi, které mají danou délku trvání a jasně definovanou osobu zodpovědnou za jejich provedení. Analýza rizik se navíc projevila jako velice užitečná, protože připravené opatření pomohlo napravit bezpečnostní událost bez většího dopadu.

Samotná migrace proběhla ve dvou fázích, kdy v první fázi bylo migrováno několik set síťových objektů. Tato část proběhla bez větších potíží a automatizace pomocí Python skriptu a RESTful API snížila celkovou dobu trvání na zlomek. Na základě těchto objektů byla následně v druhé části přidána pravidla, které se povedlo v první řadě protřídit

a pročistit a zajistit tak větší přehled a jasně definovanou podobu. Tato část opět proběhla bez větších problémů a všechna potřebná pravidla byla migrována na nový firewall, což umožňuje společnosti přejít k další části, a to je fyzické přepojení firewallů. Celá práce byla v neposlední řadě zakončena důležitým finančním zhodnocením, které se zaměřuje na celkové náklady na vlastnictví produktu během jeho doby životnosti. Tyto náklady vyšly na 1 219 045 Kč. Tímto je možné považovat hlavní cíl práce za splněný.

Dílčím cílem práce bylo také vytvoření vhodného podkladu nebo vodítka pro společnost nebo jedince, který by se ocitl v podobné situaci. Využité metody se projeví jako velmi užitečné, především analýza rizik, jak již bylo zmíněno. Konkrétní návrh řešení by také mohl posloužit jako podrobný návod, který si může čtenář doplnit nebo interpretovat podle svých potřeb. V situaci, kdy není řešení aplikovatelné může pomoci jako inspirace při tvorbě vlastní myšlenky. Z tohoto pohledu můžeme považovat i tento dílčí cíl za splněný.

SEZNAM POUŽITÝCH ZDROJŮ

1. BYRES, Eric; KARSCH, John a CARTER, Joel. FIREWALL DEPLOYMENT FOR SCADA AND PROCESS CONTROL NETWORKS GOOD PRACTICE GUIDE. Online. Centre for the Protection of National Infrastructure, 2005. Dostupné z: https://icscsi.org/library/Documents/Best_Practices/NISCC%20-%20Good%20Practice%20Guide%20on%20Firewall%20Deployment%20for%20Control%20Systems.pdf. [cit. 2024-05-08].
2. MILLER, Lawrence C. Next-Generation Firewalls For Dummies®. Online. 2nd Edition. John Wiley & Sons, 2016. ISBN ISBN 978-1-119-24977-1. Dostupné z: https://www.paloaltonetworks.fr/content/dam/pan/en_US/assets/pdf/education/ngfw-for-dummies.pdf. [cit. 2024-05-08].
3. MORAES, Alexandre Matos da Silva Pires de. Cisco firewalls. Indianapolis: Cisco Press, 2011. ISBN 978-1-58714-109-6.
4. SANTOS, Omar; KAMPANAKIS, Panos a WOLAND, Aaron. Cisco Next-Generation Security Solutions: All-in-one Cisco ASA Firepower Services, NGIPS, and AMP. Online. Indianapolis: Cisco Press, 2016. ISBN 978-1-58714-446-2. Dostupné z: <https://www.ciscopress.com/store/cisco-next-generation-security-solutions-all-in-one-9781587144462>. [cit. 2024-05-08].
5. SEDLÁK, Petr a KONEČNÝ, Martin. Přeměna ISMS v manažerské informatice. Vydání: první. Brno: CERM, akademické nakladatelství, 2023. ISBN 978-80-7623-110-8.
6. What is the CIA Triad? Definition & Examples in Cybersecurity. Online. In: CORETELLIGENT. Coretelligent. 2022. Dostupné z: <https://coretelligent.com/insights/what-is-the-cia-triad-and-why-does-your-cybersecurity-position-depend-on-it/>. [cit. 2024-05-08].
7. CIGÁNEK, Josef. Bezpečnostní analýza firewallu. Online, Bakalářská práce. Brno: VUT – Fakulta elektrotechniky a komunikačních technologií, 2017. Dostupné z: https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=152629. [cit. 2024-05-08].

8. HONTAÑÓN, Ramón J. Linux: praktická bezpečnost. Profesional. Praha: Grada, 2003. ISBN 80-247-0652-0.
9. ARINS, Andis. Firewall as a service in SDN OpenFlow network. Online. IEEE Xplore. 2018. Dostupné z: <https://ieeexplore.ieee.org/document/7367309>. [cit. 2024-05-08].
10. HOKKANEN, Tomi. Evaluation of Virtual Firewall in Private Cloud. Online. Semantic Scholar. Dostupné z: <https://www.semanticscholar.org/paper/Evaluation-of-Virtual-Firewall-in-Private-Cloud-Hokkanen/04924acbe8f8884e5d27a9e7dfbd6505b81d2f81>. [cit. 2024-05-08].
11. SEDLÁK, Petr a KONEČNÝ, Martin. Kybernetická (ne)bezpečnost: problematika bezpečnosti v kyberprostoru. Vydání: první. Brno: CERM, akademické nakladatelství, 2021. ISBN 978-80-7623-068-2.
12. SEDLÁK, Petr. Cyber Security [prezentace]. Online. [cit. 2024-05-08].
13. Intrusion Prevention System (IPS). Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210219/intrusion-prevention-system-ips/>. [cit. 2024-05-08].
14. Advanced Threat Protection (ATP). Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210343/advanced-threat-protection-atp/>. [cit. 2024-05-08].
15. URL Filtering Policies. Online. Barracuda Campus. 2024. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210332/url-filtering-policies/>. [cit. 2024-05-08].
16. Virus Scanner. Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210342/virus-scanner/>. [cit. 2024-05-08].
17. Traffic Shaping. Online. Barracuda Campus. 2024. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210279/traffic-shaping/>. [cit. 2024-05-08].

18. CGF03204 – Firewall User Awareness – Student Guide. Online. Barracuda Campus. 2022. Dostupné z: <https://campus.barracuda.com/downloadcenter/detail/15BF/cgf03204-firewall-user-awareness-student-guide/>. [cit. 2024-05-08].
19. TLS Inspection in the Firewall. Online. Barracuda Campus. 2024. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210321/tls-inspection-in-the-firewall/>. [cit. 2024-05-08].
20. Mail Security in the Firewall. Online. Barracuda Campus. 2020. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210311/mail-security-in-the-firewall/>. [cit. 2024-05-08].
21. AUTOR, Nenalezený. Barracuda CloudGen Firewall. Online. Barracuda Networks. 2024. Dostupné z: <https://www.barracuda.com/products/network-protection/cloudgen-firewall>. [cit. 2024-05-08].
22. Overview. Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98209931/overview/>. [cit. 2024-05-08].
23. Barracuda: Your journey, secured. Online. In: Barracuda. 2020. Dostupné z: <https://www.barracuda.com/>. [cit. 2024-05-08].
24. Barracuda Firewall Admin. Online. Barracuda Campus. 2024. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210490/barracuda-firewall-admin/>. [cit. 2024-05-08].
25. CONFIGURATION Tab. Online. Barracuda Campus. 2024. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210509/configuration-tab/>. [cit. 2024-05-08].
26. Access Rules. Online. Barracuda Campus. 2021. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210198/access-rules/>. [cit. 2024-05-08].
27. Firewall Objects. Online. Barracuda Campus. 2021. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210250/firewall-objects/>. [cit. 2024-05-08].

28. Network Objects. Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210251/network-objects/>. [cit. 2024-05-08].
29. Service Objects. Online. Barracuda Campus. 2018. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210266/service-objects/>. [cit. 2024-05-08].
30. How to Configure Port Protocol Protection. Online. Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98210267/how-to-configure-port-protocol-protection/>. [cit. 2024-05-09].
31. DOLEŽAL, Jan. Projektový management: komplexně, prakticky a podle světových standardů. Expert (Grada). Praha: Grada Publishing, 2016. ISBN 978-80-247-5620-2.
32. ČESKÁ AGENTURA PRO STANDARDIZACI. ČSN EN ISO/IEC 27001 Informační bezpečnost, kybernetická bezpečnost a ochrana soukromí – Systémy managementu informační bezpečnosti – Požadavky. Praha, 2023.
33. ŠTEFÁNEK, Radoslav. Projektové řízení pro začátečníky. Brno: Computer Press, 2011. ISBN 978-8-251-2835-0.
34. Stručný popis metody RIPRAN. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/popis.html>. [cit. 2024-05-09].
35. Identifikace rizika. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/popis2.html>. [cit. 2024-05-09].
36. Kvantifikace rizika. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/popis3.html>. [cit. 2024-05-09].
37. Tabulky pro verbální hodnocení rizik. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/tab3.pdf>. [cit. 2024-05-09].
38. Snižování rizika. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/popis4.html>. [cit. 2024-05-09].
39. Celkové zhodnocení rizika. Online. RIPRAN. 2005. Dostupné z: <https://ripran.cz/popis5.html>. [cit. 2024-05-09].
40. JEŽKOVÁ, Zuzana. Projektové řízení: jak zvládnout projekty. Kuřim: Akademické centrum studentských aktivit, 2013. ISBN 978-80-905297-1-7.

41. BLADH, Anna a STRÖM, Anna. Total Cost of Ownership. Online, Tézé. Lund University: Lund University, 2008. Dostupné z: <https://lup.lub.lu.se/luur/download?func=downloadFile&recordId=1351067&fileId=2435166>. [cit. 2024-05-09].
42. CTO's Guide to the Total Cost of Ownership (TCO) of a Digital Product. Online. Simform. 2023. Dostupné z: <https://www.simform.com/blog/ctos-guide-total-cost-of-ownership/>. [cit. 2024-05-09].
43. Types of costs that comprise TCO. Online. In: Simform. 2023. Dostupné z: <https://www.simform.com/blog/ctos-guide-total-cost-of-ownership/>. [cit. 2024-05-09].
44. RICHARDSON, Leonard; AMUNDSEN, Michael a RUBY, Sam. RESTful Web APIs. Sebastopol: O'Reilly, 2013. ISBN 978-1-4493-5806-8.
45. GUPTA, Lokesh. What is REST? Online. Restfulapi.net. 2023. Dostupné z: <https://restfulapi.net/>. [cit. 2024-05-09].
46. Add a new network object. Online. Barracuda Campus. 2022. Dostupné z: <https://campus.barracuda.com/product/nextgenfirewallf/api/>. [cit. 2024-05-09].
47. GUPTA, Lokesh. HTTP Status Codes. Online. Restfulapi.net. 2023. Dostupné z: <https://restfulapi.net/http-status-codes/#4xx>. [cit. 2024-05-09].
48. REST API. Online. Barracuda Campus. 2022. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/98209932/rest-api/>. [cit. 2024-05-09].
49. SUMMERFIELD, Mark. Python 3: výukový kurz. Vyd. 1. Brno: Computer Press, 2010. ISBN 978-80-251-2737-7.
50. Python Logo. Online. In: Cleanpng. 2021. Dostupné z: <https://www.cleanpng.com/png-python-logo-programmer-fierce-python-cliparts-663849/download-png.html>. [cit. 2024-05-09].
51. COMGUARD A.S. Profil společnosti. Online. COMGUARD A.S. COMGUARD. 2022. Dostupné z: <https://www.comguard.cz/profil-spolecnosti>. [cit. 2024-05-09].
52. Výroční zpráva 2022 COMGUARD a.s. Online. In: Justice.cz. 2023. Dostupné z: <https://or.justice.cz/ias/ui/vypis-sl->

- detail?dokument=78514829&subjektId=907360&spis=993929. [cit. 2024-05-09].
53. Úplné znění stanov akciové společnosti COMGUARD a.s. ke dni 19. května 2022. Online. In: Justice.cz. 2023. Dostupné z: <https://or.justice.cz/ias/ui/vypis-sl-detail?dokument=72793436&subjektId=907360&spis=993929>. [cit. 2024-05-09].
54. COMGUARD. Online. In: COMGUARD. 2024. Dostupné z: <https://www.comguard.cz/>. [cit. 2024-05-09].
55. Organizační struktura společnosti COMGUARD a.s. Brno: COMGUARD a.s., 2024. Interní dokument společnosti. [cit. 2024-05-09].
56. End of Sale and End of Life notice for Firewall Enterprise models S2008, S3008 and S4016 Sidewinder Control Center model C1015. Online. Forcepoint. 2023. Dostupné z: <https://support.forcepoint.com/customerhub/s/article/End-of-Sale-and-End-of-Life-notice-for-Firewall-Enterprise-models-S2008--S3008--and-S4016---Sidewinder-Control-Center-model-C1015>. [cit. 2024-05-09].
57. F180 and F183 Revision A. Online. Barracuda Campus. 2021. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/71862277/f180-and-f183-revision-a/>. [cit. 2024-05-09].
58. Barracuda CloudGen Firewall F180. Online. In: BarraGuard. 2022. Dostupné z: <https://www.barraguard.com/NG-Firewall-F180.asp>. [cit. 2024-05-09].
59. Forcepoint Sidewinder 8.3.2. Online. Websense. 2013. Dostupné z: https://www.websense.com/content/support/library/si/v83/mgmt/fe_832_pg_a_en-us.pdf. [cit. 2024-05-09].
60. How to Migrate the Server Node to the Assigned Services Node for Stand-Alone HA Pairs. Online. In: Barracuda Campus. 2023. Dostupné z: <https://campus.barracuda.com/product/cloudgenfirewall/doc/91986262/migration-of-the-server-node-to-the-assigned-services-node-optional/>. [cit. 2024-05-09].
61. KURZY.CZ. Elektřina – ceny a grafy elektřiny, vývoj ceny elektřiny 1 kWh - 1 rok – měna CZK. Online. Kurzy.cz. 2024. Dostupné z: <https://www.kurzy.cz/komodity/cena-elektriny-graf-vyvoje-ceny/1kWh-czk-1-rok>. [cit. 2024-05-09].

62. Instant Replacement. Online. Barracuda Networks. 2024. Dostupné z:
<https://www.barracuda.com/support/instant-replacement>. [cit. 2024-05-13].

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

a.s.	
Akciová Společnost	57
API	
Application Programming Interface	48
App	
Application	28
ARP	
Address Resolution Protocol	29
ATP	
Advanced Threat Protection	21
AWS	
Amazon Web Services	22
B2B	
Business to Business	57
C&C	
Command and Control	20
CC	
Control Center	23
CEO	
Chief Executive Officer	59
CGFW	
CloudGen Firewall	22
CIA	
Confidentiality, Integrity, Availability	15
CPM	
Critical Path Method	41
CSV	
Comma-Separated Values	55
<i>DDoS</i>	
Distributed Denial of Service	15
DHCP	
Dynamic Host Configuration Protocol	24
DLP	
Data Loss Prevention	60
DMZ	
Demilitarizovaná zóna	15
DNS	
Domain Name System	31
DPI	
Deep Packet Inspection	15
Dst NAT	
Destination Network Address Translation	28
<i>EOL</i>	

<i>End of Life</i>	60
FW	
Firewall	14
FWaaS	
Firewall as a Service	19
GB	
Gigabyte	73
GUI	
Graphical User Interface	84
HA	
High Availability	72
HTTP	
Hypertext Transfer Protocol	52
HTTP/S	
Hypertext Transfer Protocol/Secure	48
HTTPS	
Hypertext Transfer Protocol Secure	52
HW	
<i>Hardware</i>	123
ICMP	
Internet Control Message Protocol	17
IČO	
Identifikační Číslo Osoby	58
ID	
Identification	34
IDS	
Intrusion Detection System	18
Ing.	
Inženýr	59
IP adresy	
<i>Internet Protocol address</i>	84
IP range	
<i>Rozsah použitelných IP adres</i>	84
IPS	
Intrusion Prevention System	18
IPsec	
Internet Protocol Security	23
IPv4	
Internet Protocol version 4	29
IPv6	
Internet Protocol version 6	29
ISO	
<i>Optical disc image</i>	73
ISO/OSI	
Referenční model	16
IT	
Informační technologie	57
JSON	

JavaScript Object Notation	49
Kč	
Koruna česká.....	57
KM	
Konec možný	42
KP	
Konec přípustný	42
ks	
kus.....	123
kWh	
kilowatthodina	125
LAN	
Local Area Network	97
MAC	
Media Access Control.....	29
md	
Manday	123
MIME	
Multipurpose Internet Mail Extensions	30
NAT	29
ND	
Nízký dopad.....	36
NGFW	
Next-generation firewall	18
NHR	
Nízká hodnota rizika	36
NP	
Nízká pravděpodobnost	36
PAR	
Parchive soubor	72
PERT	
Program Evaluation Review Technique	41
RACI	
Matice odpovědnosti.....	45
RC	
Rezerva celková	43
REST API	
REpresentational State Transfer Application Programming Interface	48
RIPRAN	
RIsk PROject ANalys	34
RNDr.	
Doktor přírodních věd.....	59
Sandboxing	
Izolované testování	79
SD	
Střední dopad	36
SHR	
Střední hodnota rizika.....	36

Source NAT	
Source Network Address Translation	117
SP	
Střední pravděpodobnost	36
SSH	
Secure Shell Protocol	24
SSL	
Secure Socket Layer	18
TCO	
Total Cost of Ownership	46
TCP	
Transmission Control Protocol	17
TCP-RST	
TCP Reset	28
TINA	
TINA VPN Protokol	23
<i>ttl</i>	
<i>Time to Live</i>	103
<i>TXT</i>	
<i>Textový soubor</i>	88
UDP	
User Datagram Protocol	17
URL	
Uniform Resource Locator	21
USB	
Universal Serial Bus	73
VD	
Velký dopad	36
VHR	
Vysoká hodnota rizika	36
VP	
Vysoká pravděpodobnost	36
VPN	
Virtual Private Network	23
VUT	
Vysoké učení technické	34
<i>ZM</i>	
<i>Začátek možný</i>	42
<i>ZP</i>	
<i>Začátek přípustný</i>	42

SEZNAM POUŽITÝCH OBRÁZKŮ

Obrázek 1: Zjednodušený diagram firewallu chránící vnitřní síť (Zdroj: Vlastní zpracování dle: (1)).....	14
Obrázek 2: Zjednodušený diagram vytvoření DMZ pomocí firewallu (Zdroj: Vlastní zpracování dle: (1)).....	15
Obrázek 3: CIA triáda (Zdroj: Vlastní zpracování dle: (6))	16
Obrázek 4: Logo společnosti Barracuda (Zdroj: (23)).....	23
Obrázek 5: Servisní lišta v Barracuda CGFW (Zdroj: (24)).....	24
Obrázek 6: Konfigurační strom Barracuda CGFW (Zdroj: Vlastní zpracování).....	25
Obrázek 7: Ovládací prvky konfiguračních stránek (Zdroj: (25)).....	26
Obrázek 8: Barracuda CGFW – Forwarding Rules – Access Rules (Zdroj: Vlastní zpracování).....	27
Obrázek 9: Barracuda CGFW - Network Objects (Zdroj: Vlastní zpracování)	30
Obrázek 10: Barracuda CGFW – Service Objects (Zdroj: Vlastní zpracování).....	32
Obrázek 11: Síťový graf – hrana (Zdroj: Vlastní zpracování dle: (40)).....	44
Obrázek 12: Příklad hranově definovaného síťového grafu bez hodnot (Zdroj: Vlastní zpracování dle: (40)).....	44
Obrázek 13: Síťový graf – uzel (Zdroj: Vlastní zpracování dle: (40))	45
Obrázek 14: Příklad uzlově definovaného síťového grafu bez hodnot (Zdroj: Vlastní zpracování dle: (40)).....	45
Obrázek 15: Typy nákladů, ze kterých se skládá TCO (Zdroj: (43))	47
Obrázek 16: REST API - Požadavek (Zdroj: (46)).....	50
Obrázek 17: REST API – Objekt (Zdroj: (46))	50
Obrázek 18: REST API – Pole (Zdroj: (46))	50
Obrázek 19: Barracuda CGFW – Konfigurační strom – REST API Service (Zdroj: Vlastní zpracování).....	52
Obrázek 20: Barracuda CGFW – Konfigurace REST API Service (Zdroj: Vlastní zpracování).....	53
Obrázek 21: Logo Python (Zdroj: (50)).....	53
Obrázek 22: Logo společnosti COMGUARD a.s. (Zdroj: (54))	58
Obrázek 23: Barracuda CloudGen Firewall (Zdroj: (58))	61

Obrázek 24: Struktura objektů uvnitř Forcepoint Sidewinder (Zdroj: Vlastní zpracování)	63
Obrázek 25: Struktura pravidel uvnitř Forcepoint Sidewinder (Zdroj: Vlastní zpracování)	64
Obrázek 26: Vytvoření zálohy konfigurace firewallu (Zdroj: Vlastní zpracování).....	72
Obrázek 27: Soubory potřebné pro obnovení firewallu (Zdroj: Vlastní zpracování).....	73
Obrázek 28: Tvorba instalačního USB – Auto Installation (Zdroj: Vlastní zpracování)	73
Obrázek 29: Tvorba instalačního USB – Partition Settings (Zdroj: Vlastní zpracování)	74
Obrázek 30: Tvorba instalačního USB - Script Settings (Zdroj: Vlastní zpracování)....	74
Obrázek 31: Tvorba instalačního USB - USB Flash Drive Settings (Zdroj: Vlastní zpracování).....	75
Obrázek 32: Tvorba instalačního USB - Vytváření (Zdroj: Vlastní zpracování)	76
Obrázek 33: Pohled do instalačního USB (Zdroj: Vlastní zpracování).....	76
Obrázek 34: Formátování instalačního USB (Zdroj: Vlastní zpracování)	77
Obrázek 35: Režim primárního a sekundárního FW po úspěšném obnovení (Zdroj: (60))	78
Obrázek 36: Síťový graf - část 1. (Zdroj: Vlastní zpracování).....	82
Obrázek 37: Síťový graf - část 2. (Zdroj: Vlastní zpracování).....	83
Obrázek 38: Přehled síťových objektů v Barracuda CGFW (Zdroj Vlastní zpracování)	84
Obrázek 39: Uzamčení konfigurace síťových objektů (Zdroj Vlastní zpracování).....	84
Obrázek 40: Editor síťových objektů (Zdroj Vlastní zpracování).....	85
Obrázek 41: Editor síťových objektů - typ objektu (Zdroj Vlastní zpracování).....	85
Obrázek 42: Editor síťových objektů - přidání IP adresy (Zdroj Vlastní zpracování) ...	86
Obrázek 43: Editor síťových objektů - přidání regionu/země (Zdroj Vlastní zpracování)	86
Obrázek 44: Editor síťových objektů - vyplněné údaje (Zdroj Vlastní zpracování)	87
Obrázek 45: Nově přidáný síťový objekt admin1 (Zdroj Vlastní zpracování).....	87
Obrázek 46: Struktura síťových objektů Forcepoint Sidewinder (Zdroj Vlastní zpracování).....	88
Obrázek 47: Skript1 - funkce quote_term (Zdroj Vlastní zpracování).....	88
Obrázek 48: Skript1 - funkce pro host objekty (Zdroj Vlastní zpracování)	89

Obrázek 49: Skript1 - funkce pro IP adresy (Zdroj: Vlastní zpracování).....	89
Obrázek 50: Skript1 - funkce pro subnety (Zdroj: Vlastní zpracování)	89
Obrázek 51: Skript1 - funkce pro netgroupy (Zdroj: Vlastní zpracování).....	90
Obrázek 52: Skript1 - funkce pro IP rozsahy (Zdroj: Vlastní zpracování).....	90
Obrázek 53: Skript1 - čtení a zápis do CSV souborů (Zdroj: Vlastní zpracování)	91
Obrázek 54: Skript1 - cyklus volající funkce (Zdroj: Vlastní zpracování).....	91
Obrázek 55: Skript1 - přetvořené síťové objekty do nového formátu (Zdroj: Vlastní zpracování).....	92
Obrázek 56: Konfigurační strom - Administrators (Zdroj: Vlastní zpracování)	93
Obrázek 57: Konfigurace administrátorů (Zdroj: Vlastní zpracování).....	93
Obrázek 58: Přidání administrátora - název (Zdroj: Vlastní zpracování).....	94
Obrázek 59: Přidání administrátora - další nastavení (Zdroj: Vlastní zpracování)	94
Obrázek 60: Přidání administrátora - nově vytvoření administrátor Admin (Zdroj: Vlastní zpracování).....	95
Obrázek 61: Konfigurační strom - Infrastructure Services - REST API Service (Zdroj: Vlastní zpracování).....	95
Obrázek 62: Povolení REST API (Zdroj: Vlastní zpracování).....	96
Obrázek 63: Konfigurační strom - Assigned Services - NGFW (Firewall) - Forwarding Rules (Zdroj: Vlastní zpracování).....	97
Obrázek 64: Forwarding rules (Zdroj: Vlastní zpracování).....	97
Obrázek 65: Vytvoření Access Rule pro REST API (Zdroj: Vlastní zpracování)	98
Obrázek 66: Nově vytvořené pravidlo pro REST API (Zdroj: Vlastní zpracování)	98
Obrázek 67: Skript2 - import knihoven a vytvoření hlaviček (Zdroj: Vlastní zpracování)	99
Obrázek 68: Skript2 - vytvoření proměnné payload (Zdroj: Vlastní zpracování).....	100
Obrázek 69: Skript2 - funkce pro odstranění dnsLifetime (Zdroj: Vlastní zpracování)	100
Obrázek 70: Skript2 - funkce quote_term (Zdroj: Vlastní zpracování).....	100
Obrázek 71: Skript2 - funkce pro migraci objektů IP adresa (Zdroj: Vlastní zpracování)	101
Obrázek 72: Skript2 - funkce pro migraci objektů subnet (Zdroj: Vlastní zpracování).....	102

Obrázek 73: Skript2 - funkce pro migraci objektů IP rozsah (Zdroj: Vlastní zpracování)	102
Obrázek 74: Skript2 - funkce pro migraci objektů netgroup (Zdroj: Vlastní zpracování)	103
Obrázek 75: Skript2 - funkce pro migraci objektů hostnames (Zdroj: Vlastní zpracování)	104
Obrázek 76: Skript2 - čtení CSV souboru a vytvoření pole rows (Zdroj: Vlastní zpracování)	104
Obrázek 77: Skript2 - čtení CSV souboru uložené do proměnné csv_reader (Zdroj: Vlastní zpracování)	104
Obrázek 78: Skript2 - cyklus volající vytvořené funkce (Zdroj: Vlastní zpracování)	105
Obrázek 79: Skript2 - kontrolní výpis objektů z CSV souboru (Zdroj: Vlastní zpracování)	105
Obrázek 80: Skript2 - návrat status kódu po každém požadavku (Zdroj: Vlastní zpracování)	106
Obrázek 81: Výsledek migrace pomocí REST API uvnitř Barracuda CGFW (Zdroj: Vlastní zpracování)	106
Obrázek 82: Konfigurační strom - Forwarding Rules (Zdroj: Vlastní zpracování)	108
Obrázek 83: Access Rules před migrací (Zdroj: Vlastní zpracování)	109
Obrázek 84: Editor pravidel - přidání nového pravidla (Zdroj: Vlastní zpracování)	109
Obrázek 85: Editor pravidel - typy pravidel (Zdroj: Vlastní zpracování)	110
Obrázek 86: Editor pravidel - využití síťových objektů (Zdroj: Vlastní zpracování)	111
Obrázek 87: Nově vytvořené pravidlo (Zdroj: Vlastní zpracování)	111
Obrázek 88: Aktivace po vytvoření pravidla (Zdroj: Vlastní zpracování)	111
Obrázek 89: Editor pravidel - tvorba Dst NAT pravidla s použitím Service objektů (Zdroj: Vlastní zpracování)	112
Obrázek 90: Vytvoření nového service objektu – New Service Object... (Zdroj: Vlastní zpracování)	113
Obrázek 91: Editor service objektů – název (Zdroj: Vlastní zpracování)	113
Obrázek 92: Editor service objektů - volba protokolu a portu (Zdroj: Vlastní zpracování)	114

Obrázek 93: Editor service objektů - přidání protokol a port (Zdroj: Vlastní zpracování)	114
Obrázek 94: Object Viewer - nově vytvořený service objekt (Zdroj: Vlastní zpracování)	115
Obrázek 95: Editor service objektů - reference (Zdroj: Vlastní zpracování)	116
Obrázek 96: Využití service objektu v novém pravidlu (Zdroj: Vlastní zpracování)...	116
Obrázek 97: Nastavení Source NAT - volba Policies (Zdroj: Vlastní zpracování).....	117
Obrázek 98: Vytvoření Source NAT - volba typu Source NAT (Zdroj: Vlastní zpracování)	118
Obrázek 99: Vytvoření Source NAT - explicitní Source NAT (Zdroj: Vlastní zpracování)	118
Obrázek 100: Vytvoření Source NAT – editor (Zdroj: Vlastní zpracování)	119
Obrázek 101: Vytvoření Source NAT - vyplněná explicitní IP (Zdroj: Vlastní zpracování)	119
Obrázek 102: Nové pravidlo s explicitním Source NAT (Zdroj: Vlastní zpracování).	120
Obrázek 103: Přidání sekce - přehled všech pravidel (Zdroj: Vlastní zpracování).....	121
Obrázek 104: Přidání sekce - nová sekce Migrated FW Rules (Zdroj: Vlastní zpracování)	121

SEZNAM POUŽITÝCH TABULEK

Tabulka 1: Tabulka RIPRAN (Zdroj: Vlastní zpracování dle: (35)).....	35
Tabulka 2: Verbální ohodnocení pravděpodobnosti výskytu rizika (Zdroj: Vlastní zpracování dle: (37)).....	36
Tabulka 3: Verbální ohodnocení dopadu rizika (Zdroj: Vlastní zpracování dle: (37)) ..	36
Tabulka 4: Matice pro přiřazení třídy hodnoty rizika (Zdroj: Vlastní zpracování dle: (37))	37
Tabulka 5: Tabulka RIPRAN – Ohodnocená (Zdroj: Vlastní zpracování dle: (35)).....	37
Tabulka 6: Tabulka RIPRAN – Návrh na opatření (Zdroj: Vlastní zpracování dle: (35))	39
Tabulka 7: Kompletní tabulka RIPRAN (Zdroj: Vlastní zpracování dle: (35))	40
Tabulka 8: RACI matice – příklad (Zdroj: Vlastní zpracování dle: (31))	46
Tabulka 9: Základní charakteristika společnosti (Zdroj: Vlastní zpracování dle: (52)) .	58
Tabulka 10: Identifikovaná rizika migrace (Zdroj: Vlastní zpracování)	66
Tabulka 11: Verbální ohodnocení pravděpodobnosti výskytu rizika (Zdroj: Vlastní zpracování dle: (37)).....	67
Tabulka 12: Verbální vyjádření dopadu rizika (Zdroj: Vlastní zpracování dle: (37))....	68
Tabulka 13: Matice výpočtu hodnoty rizika (Zdroj: Vlastní zpracování dle: (37)).....	68
Tabulka 14: Ohodnocení rizik migrace (Zdroj: Vlastní zpracování).....	69
Tabulka 15: Návrhy na opatření rizik migrace a nová hodnota rizika (Zdroj: Vlastní zpracování).....	70
Tabulka 16: Identifikované činnosti projektu s délkou trvání (Zdroj: Vlastní zpracování)	79
Tabulka 17: RACI matice projektu migrace (Zdroj: Vlastní zpracování)	80
Tabulka 18: Tabulka pořizovacích nákladů (Zdroj: Vlastní zpracování)	123
Tabulka 19: Tabulka provozních nákladů za celou dobu životnosti produktu (Zdroj: Vlastní zpracování).....	126
Tabulka 20: Tabulka nákladů na obnovu (Zdroj: Vlastní zpracování).....	128
Tabulka 21: TCO po dobu životního cyklu (Zdroj: Vlastní zpracování)	129
Tabulka 22: Porovnání jednotlivých typů nákladů (Zdroj: Vlastní zpracování)	129
Tabulka 23: Porovnání nákladů za každý rok (Zdroj: Vlastní zpracování).....	130

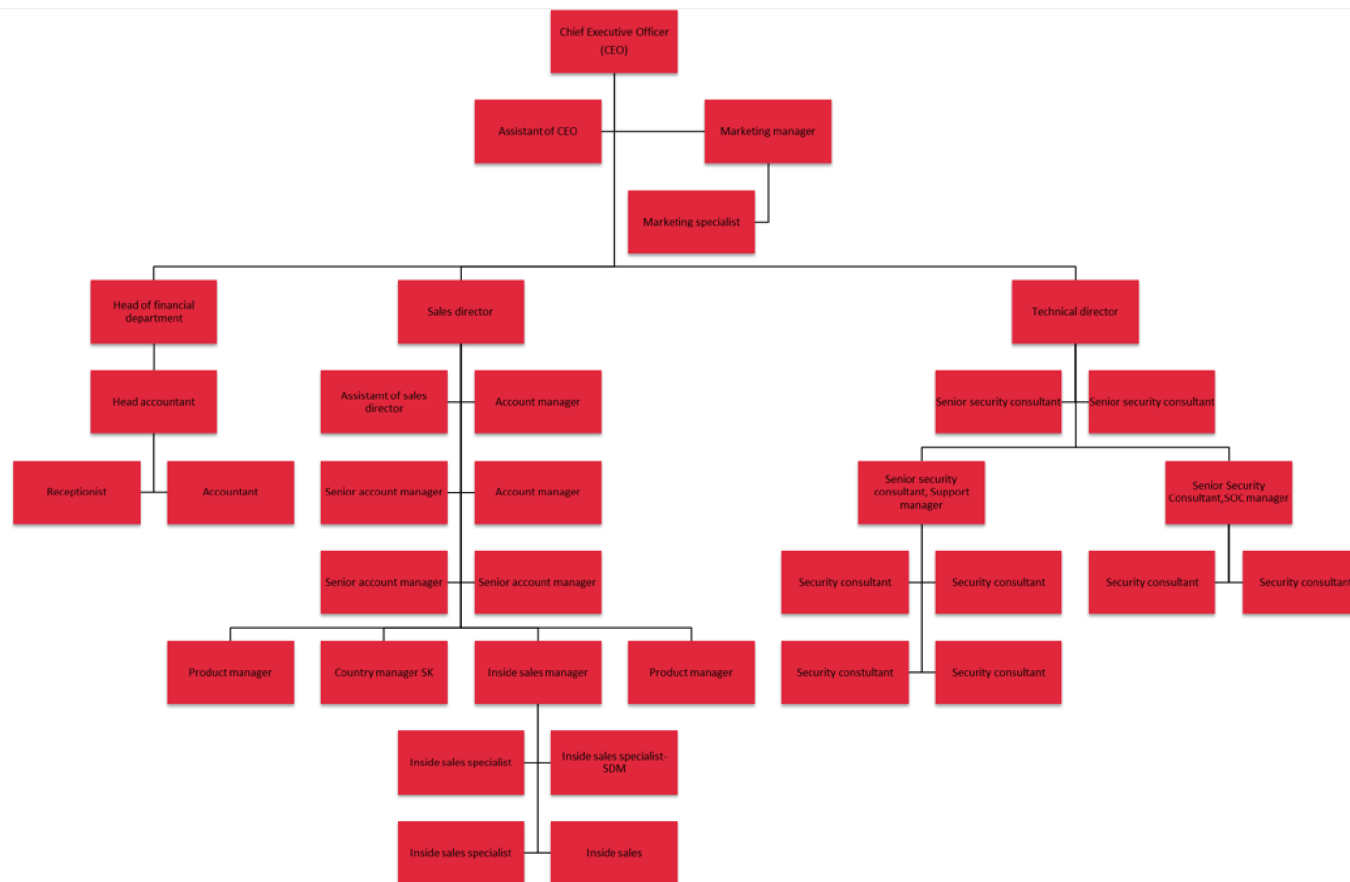
SEZNAM POUŽITÝCH GRAFŮ

Graf 1: Rozložení pořizovacích nákladů (Zdroj: Vlastní zpracování).....	125
Graf 2: Porovnání provozních nákladů během životního cyklu produktu (Zdroj: Vlastní zpracování).....	127
Graf 3: Porovnání jednotlivých typů nákladů (Zdroj: Vlastní zpracování)	130
Graf 4: Porovnání nákladů za každý rok (Zdroj: Vlastní zpracování)	131

SEZNAM PŘÍLOH

Příloha 1: Organizační struktura společnosti COMGUARD a.s. (Zdroj: (52))	153
Příloha 2: Tabulka RIPRAN - Celkové posouzení rizik (Zdroj: Vlastní zpracování) ..	154
Příloha 3: CPM – tabulka s časovými charakteristikami a kritickou cestou (Zdroj: Vlastní zpracování).....	156

PŘÍLOHY



Příloha 1: Organizační struktura společnosti COMGUARD a.s. (Zdroj: (52))

Příloha 2: Tabulka RIPRAN - Celkové posouzení rizik (Zdroj: Vlastní zpracování)

ID	Hrozba	Scénář	Prav. rizika	Dopad rizika	Hodnota rizika	Návrh na opatření	Nová hodnota rizika
1	Dojde k zničení nebo ztrátě migrovaných dat	Ztráta dat potřebných k provozu nového FW	SP	ND	NHR	Vytvoření záloh na jiných zařízeních, možnost opětovného exportování ze starého FW a následné očištění	NHR
2	Ztracení přístupu k firewallu	Dojde ke změně nastavení a nastane ztráta možnosti konfigurovat a řídit FW	SP	SD	SHR	Pravidelné zálohování konfigurace FW pro případnou obnovu zařízení. Důkladná kontrola migrovaných objektů, pravidel a jiných nastavení.	NHR
3	Únik dat z migrovaných pravidel nebo objektů	Velmi citlivá data mohou být zneužita k útoku na vnitřní síť zvenčí díky znalosti její struktury a jejich slabých míst	SP	ND	NHR	Ukládání dat na bezpečných zařízeních, best practices práce s citlivými daty, anonymizace dat	NHR
4	Neoprávněný přístup na FW	FW bude zneužit neoprávněnou osobou pro škodlivé činnosti a může vzniknout možná hrozba pro společnost	NP	VD	SHR	Vytvoření přístupů pro oprávněné uživatele a ochrana těchto přístupů, fyzické zabezpečení přístupu k FW v uzamčené serverovně	NHR
5	Vynechání objektů	Pravidla nebudou moci být plně vytvořena a mohlo by dojít k chybám, které vytvoří zranitelnosti	SP	SD	SHR	Testovací migrace a kontrola úplnosti exportovaných a importovaných objektů	NHR

6	Vynechání pravidel	Pravidla nebudou úplná a některý provoz nebude moci projít	SP	VD	VHR	Testovací migrace a kontrola úplnosti exportovaných a importovaných pravidel	SHR
7	Poškození zařízení	Zařízení bude muset být odesláno na opravu/náhradu a projekt se prodlouží	NP	VD	SHR	Poučení o bezpečné manipulaci se zařízením, minimalizace přesun, bezpečné umístění a omezení přístupu na oprávněné osoby, vytvoření zápůjční smlouvy	NHR
8	Odcizení cvičného zařízení	Vznikne finanční ztráta a v horším případě dojde i k úniku citlivých informací	NP	VD	SHR	Vytvoření zápůjční smlouvy, poučení o bezpečné úschově a přenosu zařízení, minimalizace přesun, bezpečné umístění a omezení přístupu na oprávněné osoby	NHR
9	Nezabezpečení vstupu do serverovny	Může být narušena bezpečnost FW, ale i jiných zařízení	NP	VD	SHR	Minimalizace oprávněných osob pro vstup do serverovny na nejpotřebnější, kontrola uzamčení vstupu	NHR
10	Pověření migrací nedostatečně kvalifikovanou osobou	Dojde k vytvoření chyb při konfiguraci nebo migraci objektů a pravidel, což může vést k vytvoření zranitelností	SP	VD	VHR	Školení pověřených pracovníků, určení řídicího a kvalifikovaného pracovníka, dohled nad prací pověřených pracovníků	SHR

Příloha 3: CPM – tabulka s časovými charakteristikami a kritickou cestou (Zdroj: Vlastní zpracování)

ID	Popis činnosti	Doba trvání činnosti (t)	Předcházející činnost	ZM	KM	KP	ZP	RC
A	Zadání projektu	2 dny	-	0	2	2	0	0
B	Školení IT pracovníka	60 dní	A	2	62	62	2	0
C	Sandboxing (vlastní procvičování)	60 dní	B	62	122	122	62	0
D	Tvorba skriptu	5 dní	B, C	122	127	127	122	0
E	Práce s REST API	2 dny	D	127	129	129	127	0
F	Testování	1 den	E	129	130	130	129	0
G	Vytvoření prostředí pro exekuci skriptu	0,5 dne	E, F	130	130,5	130,5	130	0
H	Migrace objektů	1 den	G	130,5	131,5	131,5	130,5	0
I	Kontrola (současná)	0,5 dne	G	130,5	131	136	135,5	5
J	Kontrola (po importu)	0,5 dne	H	131,5	132	136	135,5	4
K	Doplnění chybějících objektů	0,5 dne	I, J	132	132,5	136,5	136	4

L	Příprava FW pravidel pro migraci	5 dní	H	131,5	136,5	136,5	131,5	0
M	Vyčlenění první pravidel pro migraci	0,5 dne	K, L	136,5	137	137	136,5	0
N	Provedení zkušební části migrace	1 den	M	137	138	138	137	0
O	Kontrola zkušební migrace	0,5 dne	N	138	138,5	138,5	138	0
P	Úplná migrace	4 dny	O	138,5	142,5	142,5	138,5	0
Q	Kontrola (současně)	2 dny	O	138,5	140,5	142,5	140,5	2
R	Předání výsledku migrace	0,5 dne	P, Q	142,5	143	143	142,5	0
S	Úpravy a zhodnocení připravenosti	5 dní	R	143	148	148	143	0
T	Přepojení FWs	1 den	S	148	149	149	148	0
U	Ukončení projektu	2 dny	T	149	151	151	149	0