Univerzita Jana Evangelisty Purkyně
v Ústí nad Labem
Přírodovědecká fakulta

Analýza síťových dat v kontextu kybernetické bezpečnosti

DIPLOMOVÁ PRÁCE

Vypracoval: Bc. Jan Keller

Vedoucí práce: RNDr. Jan Krejčí, Ph.D.

Studijní program: Aplikovaná informatika

Ústí nad Labem 2023

UNIVERZITA JANA EVANGELISTY PURKYNĚ V ÚSTÍ NAD LABEM

Přírodovědecká fakulta

Akademický rok: 2022/2023

ZADÁNÍ DIPLOMOVÉ PRÁCE

Jméno a příjmení:	**Bc. Jan KELLER**
Osobní číslo:	**F21372**
Studijní program:	**N0613P140003 Aplikovaná informatika**
Téma práce:	**Analýza síťových dat v kontextu kybernetické bezpečnosti**
Zadávající katedra:	**Katedra informatiky**

Zásady pro vypracování

Cílem diplomové práce je návrh a implementace funkční infrastruktury sloužící k analýze síťových dat na vrstvách L3 až L7 ISO-OSI pomocí Next Generation Firewall (NGWF). Dalším cílem je provedení analýzy celého řešení infrastruktury a porovnání tohoto řešení s dostupnými komerčními řešeními. Součástí práce budou také příklady možných řešení reportingů nad daty.

Osnova:

Přehled současného stavu problematiky
Teoretická část

- Úvod do problematiky analýzy síťového provozu
- Představení vybraných technologií (NGFW aj.)
- Přehled aktuálně dostupných řešení

Praktická část

- Analýza požadavků
- Návrh vlastního řešení
- Bezpečnost vlastního řešení
- Porovnání vlastního návrhu s existujícími řešeními
- Možnosti reportingu

Zhodnocení výsledků
Závěr

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. CHAPPELL, Laura. *Wireshark Network Analysis*. Second edition. San Jose: Chappell University, 2012. ISBN 1893939944.
2. HAUGDAHL, J. Scott. *Network Analysis and Troubleshooting*. Boston: Addison-Wesley, 2000. ISBN 0201433192.
3. SOBELL, Mark G. *Mistrovství v Linuxu: příkazový řádek, shell, programování*. Brno: Computer Press, 2007. ISBN 978-80-251-1726-2.
4. ZWICKY, Elizabeth D., Simon COOPER a D. Brent CHAPMAN. *Building Internet Firewalls: Internet and Web Security*. 2nd Edition. Cambridge: O´Reilly, 2000. ISBN 1565928717.

Vedoucí diplomové práce: **RNDr. Jan Krejčí, Ph.D.**
Katedra informatiky

Oponent diplomové práce: **Ing. Pavel Simr**
Policie České republiky

Datum zadání diplomové práce: **5. listopadu 2022**
Termín odevzdání diplomové práce: **8. prosince 2023**

L.S. RNDr. Jiří Škvor, Ph.D.
Digitálně podepsal RNDr. Jiří Škvor, Ph.D.
Datum: 2023.11.28 16:47:54 +01'00'

doc. RNDr. Michal Varady, Ph.D.
děkan

RNDr. Jiří Škvor, Ph.D.
vedoucí katedry

V Ústí nad Labem dne 28. listopadu 2023

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a použil jen pramenů, které cituji a uvádím v přiloženém seznamu literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona c. 121/2000 Sb., ve znění zákona c. 81/2005 Sb., autorský zákon, zejména se skutečností, že Univerzita Jana Evangelisty Purkyně v Ústí nad Labem má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Jana Evangelisty Purkyně v Ústí nad Labem oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladu, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

V Ústí nad Labem dne 29. listopadu 2023 Podpis: . . . . . .

Děkuji vedoucímu práce RNDr. Janu Krejčímu, Ph.D.
za neocenitelné rady a pomoc při tvorbě diplomové práce.

Analýza síťových dat v kontextu kybernetické bezpečnosti

Abstrakt:

Hlavním cílem diplomové práce je návrh a implementace funkční infrastruktury sloužící k analýze síťových dat na vrstvách L3 až L7 ISO/OSI pomocí Next Generation Firewall (NGWF). Dalším cílem je provedení analýzy celého řešení infrastruktury a porovnání tohoto řešení s dostupnými komerčními řešeními. Součástí práce budou také příklady možných řešení reportingů nad daty.

Klíčová slova: Analýza síťového provozu, Next Generation Firewall, Infrastruktura

Network data analysis in the context of cyber security

Abstract:

The main goal of the thesis is the design and implementation of a functional infrastructure used to analyze network data on layers L3 to L7 ISO/OSI using the Next Generation Firewall (NGWF). Another goal is to perform an analysis of the entire infrastructure solution and compare this solution with available commercial solutions. The work will also include examples of possible data reporting solutions.

Keywords: Network analysis, Next Generation Firewall, Infrasctructure

Obsah

Úvod

V našem stále více propojeném světě slouží počítačové sítě jako páteř komunikace, výměny informací a globální konektivity. Ať už posíláme e-mail, streamujeme film nebo provádíme finanční transakci, data putují složitou sítí zařízení a připojení. Tento proces přenosu dat je označován jako provoz počítačové sítě. Je realizován přenosem tzv. datových paketů, které jsou přeposílány mezi zařízeními a systémy v rámci sítě. Datové pakety jsou sloužené z metadat pro vlastní přenos a uživatelská data, kterými mohou být dokumunety různých typů jako je text, obrázky, videa nebo příkazy. Tyto balíčky uživatelksých dat putují napříč sítí přes jednotlivé aktivními prvky, kterými mohou být počítače, směrovače, přepínače a servery. Síťový provoz většinou prochází přes místí síť LAN, po nadřazenou síť WAN až doputuje ke své destinace v rámci světového internetu.

Tato diplomová práce se zaměřuje na analýzu síťového provozu v místní síti LAN, která je chráněná zařízením označovaným jako firewall. Firewall je hardwarové zařízení pro zabezpečení sítě nebo softwarová aplikace určená k monitorování, filtrování a řízení příchozího a odchozího síťového provozu na základě sady předdefinovaných bezpečnostních pravidel. V podstatě funguje jako bariéra mezi důvěryhodnou interní sítí a nedůvěryhodnou externí sítí, typicky internetem. Tradiční firewally primárně filtrují provoz na základě IP adres a portů. Firewally nové generace (NGFW) jsou zařízení pro zabezpečení sítě, která rozšiřují schopnosti tradičních firewallů s pokročilými bezpečnostními funkcemi. Díky tomu také poskytují hloubkovou kontrolu paketů, prevenci narušení, povědomí o aplikacích a další ochranu sítí před širokou škálou hrozeb.

Samotná implementace technologií firewallu, i přes to, že bude obsahovat optimální bezpečnostní pravidla (nastavení), nestačí pro zajištění bezpečného provozu skrz perimetr sítě. K zajištění vyšší bezpečnosti je nezbytné provádět analýzu síťového provozu. Analýza síťového provozu je významnou fází využívající proaktivní přístup pro vývoj schémat preventivního řízení zahlcení a pro zjištění normálních a škodlivých paketů. Cílem těchto schémat je zabránit zahlcení sítě distribucí síťových zdrojů s ohledem na předpokládaný provoz. Jsou navrženy a experimentovány různé techniky pro analýzu síťového provozu, které si v této práci představíme. V rámci datové komunikace totiž dochází k přenosu vysoce důvěrných a cenných informací, a proto je zde analýzá síťového provozu kritická pro zvýšení informační bezpečnosti.

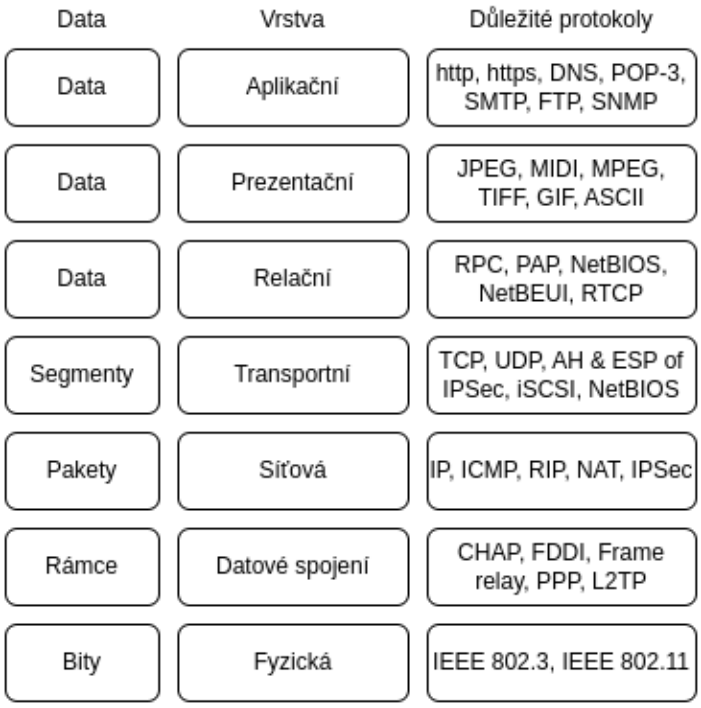
V praktické části práce se napřed zaměříme na vybudování funkční infrastruktury sloužící k analýze síťových dat pomocí NGFW. Následně jsou získaná data strojově analyzována pro potřeby vizualizace chování sítě a alertingy, které upozorní správce sítě na nezvyklé či neočekávané anomálie.

1 Úvod do problematiky

Abychom si blíže popsali analýzu síťového provozu, v krátkosti si připomeňme důležité klíčové pojmy. Začněme ISO/OSI modelem.

1.1 ISO/OSI model

Model ISO/OSI (International Organization for Standardization Open Systems Interconnection) je koncepční rámec používaný ke standardizaci a popisu toho, jak různé síťové protokoly a komunikační systémy interagují a spolupracují v počítačových sítích. Poskytuje strukturovaný způsob, jak porozumět různým vrstvám síťové komunikace a diskutovat o nich. [1]



Obrázek 1.1: ISO/OSI model

Model OSI se skládá ze sedmi vrstev, z nichž každá představuje specifický aspekt síťové komunikace. Tyto vrstvy od nejnižší jsou následující:

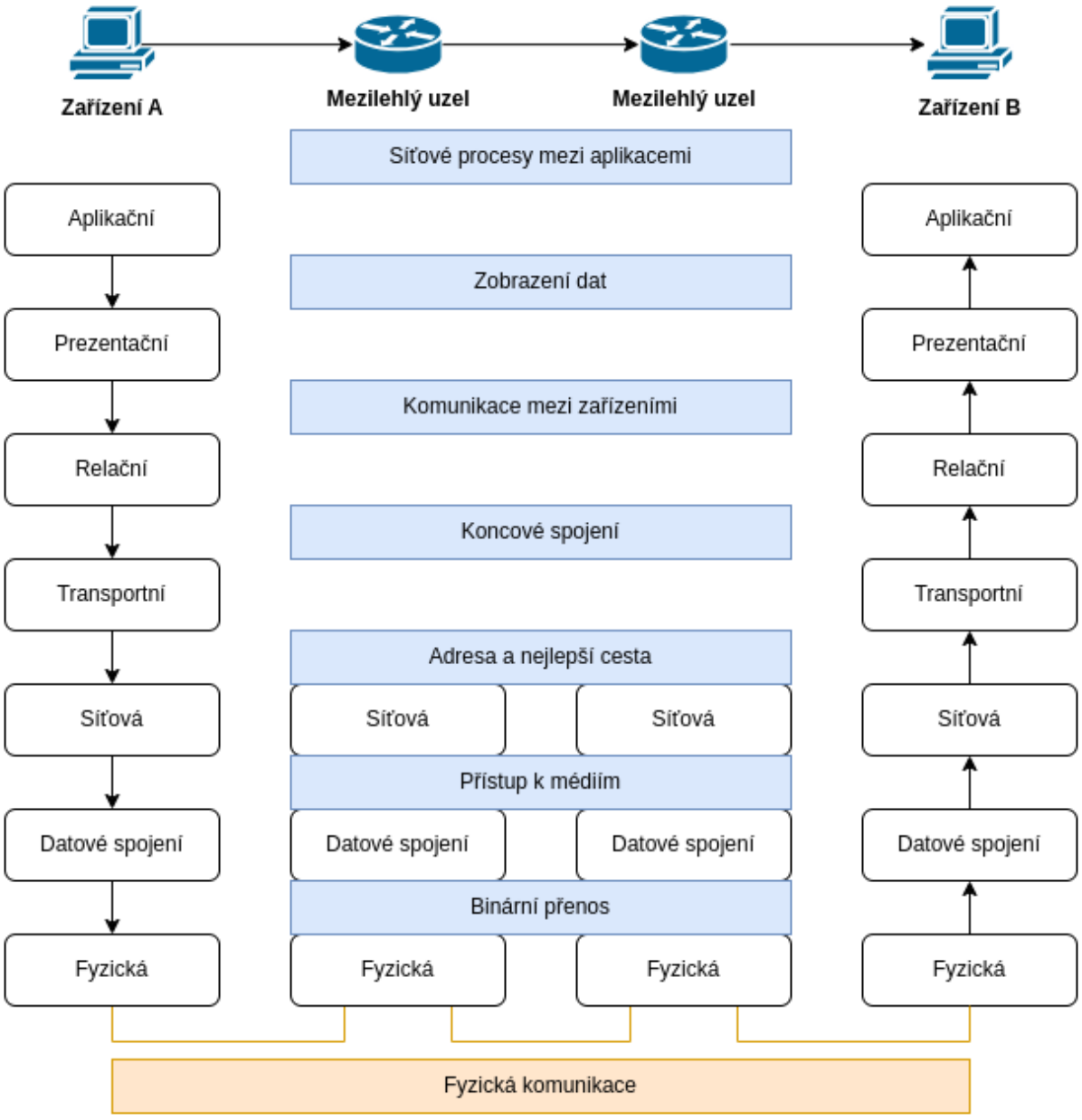
1. **Fyzická vrstva**: Tato vrstva se zabývá fyzickým přenosem nezpracovaných binárních dat přes síťové médium. Definuje charakteristiky, jako jsou úrovně napětí, typy kabelů a přenosové rychlosti. V této vrsvě nalezneme bity jako datovou jednotku. Bit je nejmenší jednotka digitálních dat a představuje nejzákladnější úroveň informací v oblasti výpočetní techniky a telekomunikací. Bit může mít hodnotu 0 nebo 1, což odpovídá dvěma možným stavům v binárním kódu. V této vrsvě se k přenosu těchto bitů používá skutečné fyzické médium (jako jsou měděné dráty, optická vlákna nebo bezdrátové rádiové vlny). [2]
2. **Vrstva datového spojení**: Zodpovídá za vytvoření spolehlivého spojení mezi dvěma přímo propojenými uzly. Zvládá rámování, adresování a detekci chyb a zajišťuje, že data jsou mezi sousedními uzly přenášena bez chyb. V této vrstvě je rámec (frame) jednotka pro přenos dat. Rámec typicky obsahuje rysy synchronizace rámce sestávající ze sekvence bitů nebo symbolů, které indikují přijímači začátek a konec dat užitečné zátěže v proudu symbolů nebo bitů, které přijímá. Pokud je přijímač připojen k systému během přenosu rámců, ignoruje data, dokud nezjistí novou sekvenci synchronizace rámců. Příklady jsou rámce Ethernet, rámce protokolu PPP (Point-to-Point Protocol) a rámce Fibre Channel. [2]
3. **Síťová vrstva**: Tato vrstva spravuje směrování a předávání datových paketů mezi různými sítěmi. Zvládá logické adresování, směrování a řízení provozu. V této vrstvě je používán paket jako datová jednotka. Paket obsahuje jak skutečná přenášená data, tak řídicí informace, jako jsou hlavičky. Packetem se budeme blíže zabívat v sekci 1.3. [3]
4. **Transportní vrstva**: Transportní vrstva zajišťuje komunikaci mezi koncovými body řízením segmentace dat, opětovného sestavení, řízení toku a obnovy chyb. Klíčovými entitami na této vrstvě jsou segmenty a porty. Segment je primární jednotka dat na transportní vrstvě. Představuje kus dat, který je vytvořen transportní vrstvou ve zdrojovém zařízení a je určen k přenosu do cílového zařízení. Transportní vrstva rozděluje data přijatá z vyšších vrstev (např. aplikační vrstva) na menší segmenty pro přenos po síti. Porty jsou logickými koncovými body pro komunikaci a hrají klíčovou roli při identifikaci konkrétních aplikací nebo služeb na zařízení. V kontextu transportní vrstvy existují dva typy portů: zdrojový port a cílový port.
   a) **Zdrojový port**: Identifikuje odesílající aplikaci nebo proces na zdrojovém zařízení.
   b) **Cílový port**: Identifikuje přijímající aplikaci nebo proces na cílovém zařízení.

   Porty jsou nezbytné pro multiplexování a umožňují více aplikacím používat síť současně. Pomáhají transportní vrstvě určit, která aplikace nebo služba na cílovém zařízení by měla přijímat příchozí data.
5. **Relační vrstva**: Tato vrstva spravuje vytvoření, údržbu a ukončení relace mezi aplikacemi. Poskytuje mechanismy pro správu ovládání dialogů a synchronizaci.
6. **Presenční vrstva**: Zodpovídá za překlad dat, šifrování a kompresi. Zajišťuje, že data odesílaná aplikační vrstvou jsou ve formátu, kterému přijímající aplikace rozumí.
7. **Aplikační vrstva**: Nejvyšší vrstva, která přímo interaguje s aplikacemi koncových uživatelů.

Poskytuje platformově nezávislé rozhraní pro softwarové aplikace pro přístup k síťovým službám, jako je e-mail, přenos souborů a vzdálený přístup.

Datový tok z vrstvy do vrstvy

Na obrázku níže můžete vidět, že zařízení **A** posílá zprávu zařízení **B** (přes mezilehlé uzly). Ve zařízení **A** zpráva proudí dolů z aplikační do fyzické vrstvy. Celá zpráva včetně hlavičky je převedena na tok nezpracovaných dat ve fyzické vrstvě zařízení **A**. Nezpracovaná data nyní putují na přijímací konec přes mezilehlé uzly. Na zařízení **B**, která je přijímací stranou, se zpráva přesune z fyzický do aplikační vrstvy. [4]



Obrázek 1.2: Datový tok z vrstvy do vrstvy

Můžeme uvažovat o sedmi vrstvách ISO/OSI modelu jako o třech podskupin. Tyto tři podskupiny jsou:

- Vrstva uživatelské podpory nebo horní tři vrstvy. Zabývá interoperabilitou mezi různými softwarovými systémy.[4]
- Vrstva síťové podpory nebo spodní tři vrstvy. Zabývá fyzickými aspekty přenosu dat od odesílatele k přijímači. Tato podskupina se zabývá fyzickými spojeními, elektrickými specifikacemi, dobou přenosu, fyzickým adresováním atd.[4]
- Transportní vrstva jako rozhraní mezi těmito dvěma podskupinami. Transportní vrstva spojuje dvě podskupiny a zajišťuje, že data odesílaná nižší podskupinou jsou ve formě použitelné pro horní podskupinu.[4]

Horní vrstvy modelu OSI jsou většinou implementovány softwarově, zatímco spodní mohou mít kombinaci hardwaru a softwaru. Výhradně fyzická vrstva je většinou hardwarová. [4]

Model OSI je koncepční rámec a nepředstavuje žádnou konkrétní technologii nebo protokol. Místo toho slouží jako vodítko pro pochopení toho, jak různé síťové protokoly a technologie interagují v rámci vrstvené struktury. Stojí za zmínku, že zatímco model OSI je informativní, mnoho síťových protokolů a systémů v reálném světě striktně nedodržuje jeho sedmivrstvou strukturu. Místo toho mohou kombinovat nebo vynechat určité vrstvy na základě svých požadavků a účinnosti. [1]

Pro analýzu provozu jsou nejdůležitější aspekty v třetí a čtvrté vrstvě. Mezi tyto aspekty patří Routing, IP adresace, Fragmentace, QoS a monitorování procesu navazování a ukončování spojení. Zde můžeme nalézt protokoly IP (Internet Protocol), ICMP, TCP (Transmission Control Protocol) a UDP (User Datagram Protocol) s kterými se nejčastěji setkáme na internetu. Pro internet je klíčový protokol IPv4 (IPv4, RFC 791), na kterém je momentálně postaven, nicméně v budoucnu můžeme očekávat předchod na novější verzi - IPv6 (IPv6, RFC 2460). Představme si tyto protokoly do většího detailu. [5]

1.2 TCP/IP

Transmission Control Protocol/Internet Protocol neboli TCP/IP je sada protokolů, které řídí, jak zařízení komunikují pro bezproblémovou výměnu dat.

TCP/IP je ve svém jádru vrstvená architektura, která představuje pečlivou organizaci, která usnadňuje efektivní přenos informací. Jeho struktura je v souladu s modelem ISO/OSI, který konceptualizuje síťové funkce na sedmi vrstvách. TCP/IP se však zaměřuje především na nižší čtyři vrstvy tohoto modelu, z nichž každá hraje klíčovou roli v procesu vytváření sítí. [6]

Na níže uvedeném diagramu je možné vidět porovnání oproti ISO/OSI modelu.

Obrázek 1.3: TCP/IP model v porovnání s ISO/OSI modelem

1. **Linková vrstva**: Tato vrstva pokládá základy pro fyzické spojení mezi zařízeními. Zahrnuje detaily síťových rozhraní na úrovni hardwaru a zajišťuje, že zařízení jsou správně připojena a mohou komunikovat prostřednictvím síťového média.
2. **Internetová vrstva**: Tato vrstva je klíčová při adresování, směrování a fragmentaci dat do paketů vhodných pro přenos po síti. Ve svém srdci je internetová vrstva zodpovědná za přidělování jedinečných IP adres zařízením, což umožňuje jejich identifikaci v rámci sítě.
3. **Transportní vrstva**: Základní pilíř pro komunikaci mezi koncovými body. Tato vrstva spravuje spolehlivost přenosu dat a rozděluje velké zprávy na menší, spravovatelné pakety. Zajišťuje, že data dorazí neporušená na místo určení, a zahrnuje mechanismy pro kontrolu a opravu chyb. Transportní vrstva navíc reguluje tok dat, zabraňuje přetížení a optimalizuje rychlost přenosu.
4. **Aplikační vrstva**: Tato vrstva slouží jako rozhraní mezi sítí a aplikacemi koncových uživatelů. Tato vrstva poskytuje základní síťové služby přímo uživatelům a aplikacím a usnadňuje úkoly, jako je přenos souborů, e-mailová komunikace a vzdálený přístup.

Modularita a flexibilita TCP/IP přispívá k jeho širokému přijetí, ale jeho význam přesahuje technické atributy. Jedna z jeho klíčových silných stránek spočívá v povaze otevřeného standardu, což znamená, že jeho specifikace jsou veřejně dostupné a neomezují se na vlastnictví. Tato otevřenost hrála klíčovou roli v šíření TCP/IP jako převládajícího síťového protokolu pro Internet. [7]

Prakticky řečeno, když vstoupíte na webovou stránku, pošlete e-mail nebo se zapojíte do jakékoli online aktivity, TCP/IP pracuje v zákulisí a organizuje složitý proces přenosu dat.

1.3 IPv4

IPv4 je zkratka pro internetový protokol verze 4. Slouží jako základní architektura pro komunikaci v rámci počítačových sítí, zejména internetu. Jak již název napovídá, jedná se o čtvrtou iteraci internetového protokolu, což je soubor pravidel a konvencí, kterými se řídí přenos dat a adresování na internetu. IPv4 hraje zásadní roli při identifikaci a směrování síťového provozu a umožňuje bezproblémovou komunikaci mezi různými zařízeními a systémy po celém světě. [8]

Jednou z definujících charakteristik IPv4 je jeho adresní systém. Adresa IPv4 se skládá z 32 bitů, rozdělených do čtyř segmentů, často zobrazovaných ve formátu dotted decimal. Každý z těchto segmentů může obsahovat celočíselnou hodnotu v rozsahu od 0 do 255. Tato struktura nabízí potenciální fond přibližně 4,3 miliardy unikátních adres, což bylo původně považováno za více než dostatečné pro pojmutí rostoucího počtu zařízení připojujících se k internetu. Prudký růst zařízení připojených k internetu, včetně počítačů, chytrých telefonů, chytrých domácích zařízení a dalších, však vedl k vyčerpání dostupných adres IPv4. [8]

V krátkosti zmíním lokální rozsahy IPv4. Lokální rozsahy IPv4 jsou vyhrazeny pro privátní sítě, což umožňuje zařízením v těchto sítích vzájemně komunikovat, aniž by jejich adresy byly přímo vystaveny veřejnému internetu. Tři běžně používané místní rozsahy IPv4, jak jsou definovány v RFC 1918, jsou: [3]

- **10.0.0.0 až 10.255.255.255**: Rozsah 10.0.0.0/8 poskytuje velký fond adres pro privátní sítě. Organizace často využívají tento rozsah pro interní sítě kvůli jeho rozsáhlému adresnímu prostoru.
- **172.16.0.0 až 172.31.255.255**: Rozsah 172.16.0.0/12 nabízí více segmentovaný adresní prostor. Správci systému běžně používají podmnožiny tohoto rozsahu pro různá oddělení nebo účely v rámci organizace.
- **192.168.0.0 až 192.168.255.255**: Řada 192.168.0.0/16 je oblíbená pro domácí sítě a sítě malých firem. Jeho flexibilita umožňuje značné množství zařízení v rámci lokalizovaného prostředí.

Tyto lokální sítě můžeme dále rozdělat na ještě menší sítě pomocí podsítí (subnetting). Jedná se o techniku, která zahrnuje rozdělení větší IP sítě na menší, lépe spravovatelné podsítě. Prostřednictvím podsítí místních rozsahů IPv4 je možné efektivně přidělovat adresy na základě požadavků na jejich síťovou architekturu. [3]

Např.: pracovní síť využívající rozsah 10.0.0.0/8 se může rozhodnout pro podsíťování do menších bloků, jako je 10.1.0.0/16 a 10.2.0.0/16, pro různá oddělení nebo umístění. Tento postup zlepšuje správu a zabezpečení sítě logickým rozdělením sítě do odlišných jednotek.

Použití místních rozsahů IPv4 poskytuje vrstvu zabezpečení tím, že zajišťuje, že interní zařízení nejsou přímo dostupná z veřejného internetu. Překlad síťových adres (NAT) dále zvyšuje zabezpečení tím, že při komunikaci s externími sítěmi překládá místní adresy na jedinou veřejnou IP adresu. [2]

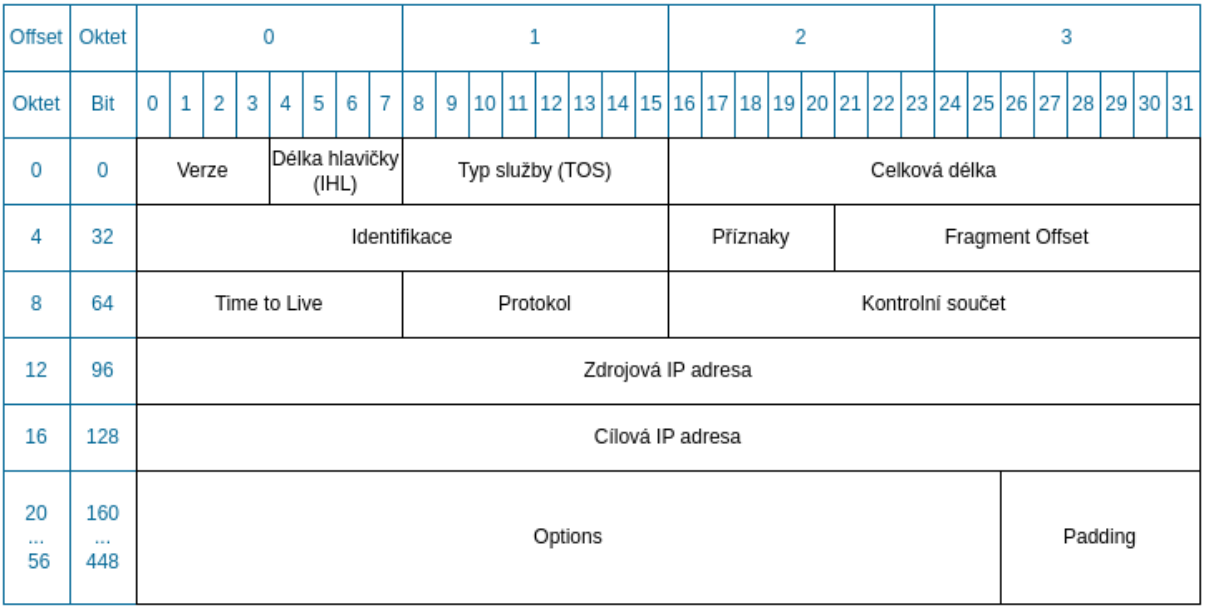
Význam IPv4 přesahuje pouhého adresování. Protokol také popisuje mechanismy směrování a předávání paketů, které umožňují datovým paketům procházet propojenými sítěmi. Každý paket má hlavičku, která obsahuje základní informace, včetně zdrojové a cílové adresy, stejně jako řídicí data pro směrování a zpracování chyb. Tento přístup založený na hlavičkách umožňuje efektivní a přesné doručování dat napříč rozsáhlými a komplexními sítěmi.

Třídy adres

IP adresy jsou rozděleny do několika tříd. Tyto třídy jsou A, B, C, D a E. Každá třída má své specifické rozsahy adres a určení.

- **Třída A (1.0.0.0 až 126.0.0.0)**: První oktet určuje třídu sítě. Rozsah IP adres: 1.0.0.0 až 126.255.255.255. První oktet je rezervován pro identifikaci třídy sítě, zatímco zbývající tři oktety jsou přiděleny konkrétním zařízením v síti.
- **Třída B (128.0.0.0 až 191.255.0.0)**: První dva oktety určují třídu sítě. Rozsah IP adres: 128.0.0.0 až 191.255.255.255. Tato třída je vhodná pro středně velké až velké sítě.
- **Třída C (192.0.0.0 až 223.255.255.0)**: První tři oktety určují třídu sítě. Rozsah IP adres: 192.0.0.0 až 223.255.255.255. Tato třída je vhodná pro menší sítě.
- **Třída D (224.0.0.0 až 239.255.255.255)**: Třída D je rezervována pro multicastové skupiny, což jsou skupiny uzlů, které jsou konfigurovány tak, aby naslouchaly na stejné IP adrese.
- **Třída E (240.0.0.0 až 255.255.255.255)**: Třída E je rezervována pro experimentální nebo výzkumné účely. [2]

Struktura IP paketu



Obrázek 1.4: Struktura IP paketu

Struktura paketů IPv4, často označovaná jako hlavička IPv4, je klíčovou součástí internetového protokolu verze 4. Obsahuje základní informace, které řídí přenos a směrování datových paketů napříč propojenými sítěmi. Hlavička IPv4 je přítomna na začátku každého paketu IPv4 a hraje klíčovou roli při zajišťování přesného a efektivního doručení dat. Přibližme si přehled struktury paketů IPv4:

- **Verze (4 bity)**: Toto pole určuje verzi používaného protokolu IP, což je v tomto případě IPv4. Hodnota je pevně nastavena na „0100“ pro IPv4.
- **Délka hlavičky (4 bity)**: Toto pole udává délku hlavičky IPv4 ve 32bitových slovech. Minimální hodnota je 5, což znamená 20bajtovou hlavičku, a maximální hodnota je 15, což odpovídá 60bajtové hlavičce. Délka hlavičky je vyžadována k nalezení začátku datového užitečného zatížení v paketu.
- **Typ služby (8 bitů)**: Pole Typ služby (TOS) se používá k definování kvality služby a úrovně priority pro zpracování paketů. Lze jej použít ke specifikaci různých úrovní služeb, jako je nízká latence, vysoká propustnost nebo normální služba.
- **Celková délka (16 bitů)**: Toto pole udává celkovou délku paketu IPv4, včetně hlavičky a datového obsahu. Hodnota je vyjádřena v bajtech. Protože maximální velikost paketu IPv4 je 65 535 bajtů, pole celkové délky pokrývá tento rozsah.
- **Identifikace (16 bitů)**: Identifikační pole se používá k jednoznačné identifikaci datagramu. Pomáhá při opětovném sestavení fragmentovaných paketů na přijímací straně.
- **Příznaky (3 bity)**: Tyto tři bity se používají pro řídicí příznaky související s fragmentací paketů. Mezi příznaky patří „Vyhrazeno“, „Nefragmentovat“ a „Další fragmenty“.

- **Fragment Offset (13 bitů)**: Toto pole označuje pozici dat v původním datagramu. Používá se ve spojení s identifikačním polem k opětovnému sestavení fragmentovaných paketů.
- **Time to Live (8 bitů)**: Pole Time to Live (TTL) udává maximální počet skoků (směrovačů), kterými může paket projít, než bude zahozen. Toto pole pomáhá zabránit neomezenému cirkulování paketů v síti.
- **Protokol (8 bitů)**: Toto pole identifikuje protokol datové zátěže, která následuje po hlavičce IPv4. Mezi běžné hodnoty patří 6 pro TCP (Transmission Control Protocol) a 17 pro UDP (User Datagram Protocol).
- **Kontrolní součet (16 bitů)**: Kontrolní součet je hodnota kontrolního součtu vypočtená pro hlavičku IPv4, aby byla zajištěna integrita dat. Pomáhá odhalit chyby v hlavičce během přenosu.
- **Zdrojová IP adresa (32 bitů)**: Toto pole obsahuje 32bitovou adresu IPv4 odesílatele (zdroje) paketu.
- **Cílová IP adresa (32 bitů)**: Toto pole obsahuje 32bitovou adresu IPv4 zamýšleného příjemce (cíl) paketu.
- **Options (Proměnná délka, pokud je k dispozici)**: Toto pole je volitelné a lze jej použít k zahrnutí dalších řídicích informací nebo parametrů souvisejících s paketem.
- **Padding (proměnná délka, v případě potřeby)**: Padding je přidán, aby se zajistilo, že celková délka hlavičky IPv4 je násobkem 32 bitů.

Struktura a pole hlavičky IPv4 pracují v tandemu, aby zajistily přesné doručování dat napříč sítěmi bez ohledu na jejich složitost a rozmanitost. Hlavička poskytuje potřebné informace pro směrovače, aby mohly přijímat rozhodnutí o směrování a pro přijímače, aby správně znovu sestavily fragmentované pakety a zpracovaly zapouzdřená data. [9]

Rychlé vyčerpání dostupných adres IPv4 podnítilo vývoj protokolu IPv6 neboli internetového protokolu verze 6. IPv6 využívá 128bitové adresy, což značně rozšiřuje dostupný adresní prostor v téměř nepředstavitelném rozsahu. Tento krok byl proaktivní reakcí na rostoucí poptávku po internetové konektivitě, zvláště když trend internetu věcí (IoT) nabral na síle. I když byl protokol IPv6 zaveden, aby nahradil protokol IPv4, přechod byl postupný kvůli převládání stávající infrastruktury IPv4 a potřebě kompatibility.

1.4 IPv6

Práce se bude zabývat sledováním provozu na IPv4 síti. Představíme si však klíčové vlastnosti IPv6.

- **Adresní prostor**: IPv4 má 32bitový adresní prostor, který umožňuje přibližně 4,3 miliardy jedinečných adres. IPv6 využívá 128bitový adresní prostor, který poskytuje astronomicky

větší počet unikátních adres (2^{128}, což je přibližně $3.4*10^{38}$ adres). Tím je zajištěna dostatečná dostupnost adresy v dohledné době.

- **Automatická konfigurace**: IPv4 vyžaduje ruční konfiguraci nebo konfiguraci IP adres založenou na DHCP. IPv6 obsahuje vestavěnou podporu pro bezstavovou automatickou konfiguraci adres (SLAAC), která umožňuje zařízením automaticky konfigurovat své adresy IPv6 a síťová nastavení, aniž by se spoléhala na DHCP.
- **Zabezpečení**: IPv6 ve výchozím nastavení obsahuje podporu pro IPSec (Internet Protocol Security), což zvyšuje bezpečnost a důvěrnost dat. IPv6 obsahuje funkce, jako jsou dočasné adresy a rozšíření ochrany osobních údajů pro zlepšení soukromí uživatelů a omezení sledování.
- **Zjednodušení hlavičky**: IPv4 hlavičky mají proměnnou délku a obsahují několik polí, což vede k režii zpracování. Hlavičky IPv6 byly zjednodušeny a opraveny na 40 bajtů, což zlepšuje efektivitu zpracování paketů.
- **Zacházení s fragmentací**: IPv4 umožňuje směrovačům fragmentovat pakety, což vede k potenciálním problémům s opětovným sestavením a výkonem. IPv6 přenáší odpovědnost za fragmentaci na odesílajícího hostitele, což pomáhá snížit zátěž zpracování na směrovačích.
- **Obsluha vícesměrového vysílání**: Vícesměrové vysílání IPv4 používá adresy třídy D a má omezení škálovatelnosti a efektivity. IPv6 multicast využívá vyhrazený rozsah adres a zavádí efektivnější mechanismy pro multicastovou komunikaci.
- **Hierarchie sítě**: IPv4 ve velké míře využívá NAT (Network Address Translation), aby šetřil adresní prostor, což může komplikovat návrh sítě. IPv6 podporuje end-to-end konektivitu a snižuje potřebu NAT, což zjednodušuje síťovou architekturu.

Celkově bylo IPv6 navrženo tak, aby řešilo omezení IPv4 a poskytlo základ pro pokračující rozšiřování internetu a zároveň zahrnovalo vylepšení v adresování, zabezpečení, efektivitě a mobilitě.

Proč je tedy protokol IPv6 používán jen velmi sporadicky? Jedním z hlavních důvodů pomalého zavádění IPv6 je přechod z IPv4 na IPv6. Stávající infrastruktura, zařízení a služby jsou konfigurovány převážně pro IPv4. Migrace na IPv6 vyžaduje značné úsilí a koordinaci, zejména u velkých sítí. [10]

IPv4 a IPv6 nejsou přímo interoperabilní. Během přechodu jsou vyžadovány konfigurace se dvěma zásobníky (podporující protokoly IPv4 i IPv6), což může zvýšit složitost a provozní problémy. Rozšířené používání NAT v IPv4 umožnilo organizacím sdílet jednu veřejnou IP adresu mezi více interními zařízeními. Zatímco IPv6 podporuje end-to-end konektivitu, některé organizace s přechodem váhají kvůli vnímané ztrátě zabezpečení založeného na NAT. [10]

Velkým problémem je také podpora dodavatelů. Hlavní dodavatelé síťových zařízení a softwaru podporují protokol IPv6, starší zařízení a software nemusí mít plnou kompatibilitu s protokolem IPv6, což vytváří problémy s interoperabilitou. Organizace značně investovaly do infrastruktury IPv4, včetně směrovačů, firewallů a aplikací. Přechod na IPv6 vyžaduje aktualizaci nebo výměnu těchto součástí, což může být časově náročné a finančně nákladné. [11] [12]

Přijetí protokolu IPv6 závisí i na poskytovatelích obsahu a online službách, kteří zpřístupní své zdroje prostřednictvím protokolu IPv6. Někteří poskytovatelé zavádějí IPv6 pomalu, což může omezit uživatelské možnosti při přístupu ke zdrojům pouze IPv6. [12]

1.5 ICMP

ICMP je zkratka pro „Internet Control Message Protocol“ (RFC 792). Jedná se o základní síťový protokol používaný v sítích internetového protokolu (IP), včetně internetu. ICMP je primárně určen pro hlášení chyb a poskytování různých diagnostických a provozních informací o stavu sítě. [13]

ICMP se běžně používá k hlášení chyb zjištěných během zpracování paketů IP. Pokud například směrovač přijme paket IP, který nemůže přeposlat, může odeslat zprávu ICMP zpět do zdroje, což znamená, že paket nemohl být doručen. [14]

Je důležité poznamenat, že ačkoli ICMP slouží zásadním účelům pro diagnostiku a správu sítě, lze jej také zneužít ke škodlivým účelům, jako jsou útoky typu denial-of-service (DoS). Některá bezpečnostní opatření zahrnují filtrování nebo kontrolu typů zpráv ICMP, které lze odesílat nebo přijímat, aby se zmírnila potenciální rizika. [13]

Zprávy ICMP mají specifickou strukturu definovanou jejich hlavičkou a v některých případech volitelným datovým obsahem zprávy. Hlavička ICMP je nedílnou součástí každé zprávy ICMP a nese základní informace o typu zprávy, kódu a kontrolním součtu.

Offset	Oktet	0								1								2								3							
Oktet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Typ								Kód								Kontrolní součet															
4	32	Data																															

Obrázek 1.5: Struktura zprávy ICMP

- **Typ (8 bitů)**: Určuje typ zprávy ICMP. Každý typ označuje jiný druh operace ICMP. Například ”Echo Request” a ”Echo Reply” mají různé hodnoty typu.
- **Kód (8 bitů)**: Dodatečně kategorizuje zprávu ICMP. Každý typ může mít přiřazeno více kódů. Kódy poskytují konkrétnější informace o účelu zprávy. Například typ „Destination Unreachable“ má různé kódy označující důvody nedostupnosti cíle.
- **Kontrolní součet (16 bitů)**: Kontrolní součet je vypočítán oproti hlavičce ICMP. Slouží k detekci chyb v hlavičce a datové zprávě během přenosu.

- **Data (proměnná, volitelná)**: Některé typy zpráv ICMP obsahují mimo hlavičku volitelná data. Například zprávy ”Echo Request” a ”Echo Reply” (používané v obslužném programu ”ping”) obsahují data, která jsou obvykle stejná jako obsah původní zprávy. To se používá pro měření času a testování konektivity. [15]

Je důležité si uvědomit, že délka a struktura datové části se může lišit v závislosti na typu zprávy ICMP. Různé typy zpráv ICMP slouží různým účelům, jako je hlášení chyb, testování sítě a diagnostika. V důsledku toho se struktura může mezi různými zprávami ICMP výrazně lišit. Příklady bežných typů zpráv jsou následující:

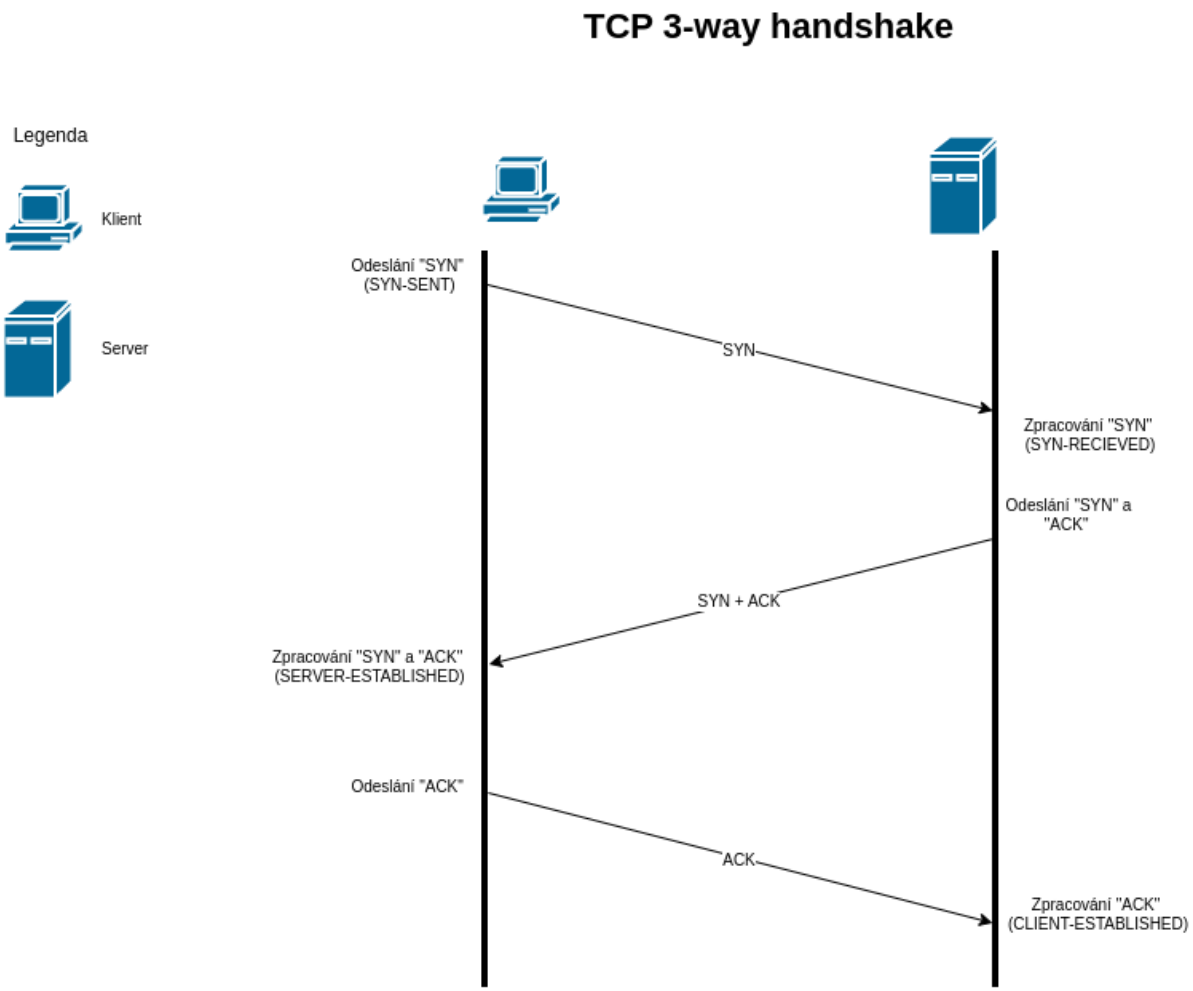
- **Přesměrování - Typ 5**: Přesměrování požaduje odeslání datových paketů alternativní cestou. ICMP Redirect je mechanismus pro směrovače, které přenášejí informace o směrování hostitelům. Zpráva informuje hostitele, aby aktualizoval své směrovací informace (k odesílání paketů alternativní cestou)
- **Překročení času - Typ 11**: Překročení času je generováno bránou, aby se informoval zdroj o vyřazeném datagramu, protože TTL dosáhl nuly. TTL může také odeslat hostitel, pokud se mu nepodaří znovu sestavit fragmentovaný datagram ve svém časovém limitu.
- **Časové razítko - Typ 13**: Časové razítko se používá pro synchronizaci času. Původní časové razítko je nastaveno na čas (v milisekundách od půlnoci), kdy se odesílatel naposledy dotkl paketu.
- **Odpověď s časovým razítkem - Typ 14**: Odpověď s časovým razítkem odpovídá na zprávu časového razítka. Skládá se z původního časového razítka zaslaného odesílatelem. Indikuje, kdy bylo časové razítko přijato.
- **Žádost o masku adresy - Typ 18**: Odpověď masky adresy se používá k odpovědi na zprávu s žádostí o masku adresy s příslušnou maskou podsítě.
- **Nedosažitelný cíl - Typ 0 - 7**: Nedosažitelný cíl je generován hostitelem nebo jeho příchozí bránou, aby informoval klienta, že cíl je z nějakého důvodu nedosažitelný. Důvody pro tuto zprávu mohou zahrnovat: fyzické připojení k hostiteli neexistuje (vzdálenost je nekonečná); uvedený protokol nebo port není aktivní; data musí být fragmentována, ale je zapnutý příznak „nefragmentovat“. Nedosažitelné porty TCP reagují zejména pomocí TCP RST spíše než nedosažitelný cíl typu 3. [15]

1.6 TCP

TCP (Transmission Control Protocol) je základní komunikační protokol používaný v počítačových sítích k zajištění spolehlivého a uspořádaného doručování dat mezi zařízeními, jako jsou počítače, servery a další síťová zařízení. [16]

Navázání spojení

Navázání spojení TCP (Transmission Control Protocol) zahrnuje proces nazývaný trojcestný handshake (3-way handshake). Tento proces se používá k vytvoření spolehlivého spojení mezi klientem a serverem před zahájením skutečného přenosu dat. Trojcestný handshake zajišťuje, že obě strany jsou připraveny vyměňovat si data a že každá strana ví, že ta druhá je dostupná a reagující. [17] [18]



Obrázek 1.6: TCP trojcestný handshake

- **Krok 1. Klient odešle segment SYN (synchronizace)**: Klient (také označován jako odesílatel) zahájí připojení odesláním segmentu TCP s příznakem SYN (synchronizovat) nastaveným na server. Příznak SYN označuje, že klient chce navázat spojení a požaduje synchronizaci se serverem. Klient také vybere počáteční pořadové číslo (ISN) pro své datové segmenty. Toto pořadové číslo pomáhá udržovat pořadí dat během přenosu.
- **Krok 2. Server odpoví pomocí segmentu SYN-ACK (Synchronize-Acknowledge)**: Po přijetí segmentu SYN od klienta server potvrdí požadavek odesláním vlastního segmentu SYN-ACK. Segment SYN-ACK obsahuje příznaky SYN i ACK. Příznak SYN označuje ochotu

serveru navázat spojení a příznak ACK potvrzuje počáteční segment SYN klienta. Server také vybere své vlastní počáteční pořadové číslo (ISN) pro datové segmenty.

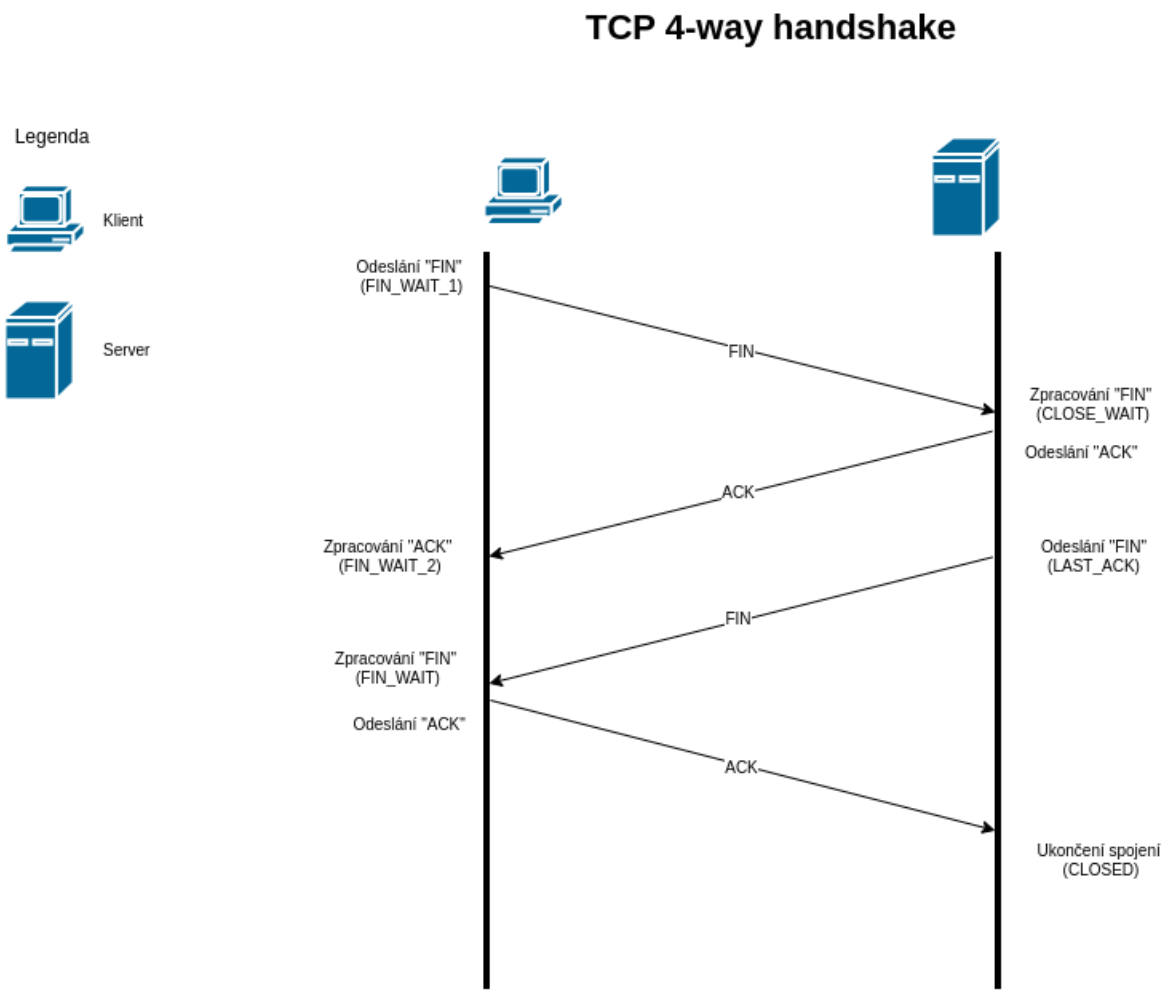
- **Krok 3: Klient odešle segment ACK (potvrzení)**: Po obdržení segmentu SYN-ACK ze serveru klient odpoví segmentem ACK. Segment ACK potvrzuje příjem segmentu SYN-ACK serveru. Je nastaven příznak ACK a pořadové číslo je zvýšeno o jedničku vzhledem k ISN serveru.

V tomto okamžiku je trojcestný handshake dokončen a mezi klientem a serverem bylo navázáno spolehlivé spojení. Obě strany si vyměnily počáteční pořadová čísla, vzájemně potvrdily připravenost komunikovat a potvrdily, že mohou odesílat a přijímat data. [16] [17]

Účelem tříkrokového handshake je zajistit, aby obě strany byly připraveny vysílat a přijímat data bez jakýchkoli nejasností ohledně stavu připojení. Pomáhá také stanovit některé počáteční parametry pro připojení, jako jsou pořadová čísla a velikosti oken (pro řízení toku). Tato spolehlivost a potvrzovací proces jsou klíčové pro zachování integrity a pořádku dat během přenosu přes protokol TCP.

Ukončení spojení

Ukončení připojení TCP zahrnuje proces zvaný čtyřcestný handshake. Tento proces se používá k řádnému ukončení TCP spojení mezi klientem a serverem. Čtyřcestný handshake zajišťuje, že obě strany dokončily odesílání a příjem dat před uzavřením spojení. [18]



Obrázek 1.7: TCP čtyřcestný handshake

- **Krok 1. Klient odešle segment FIN (dokončení)**: Klient (odesílatel) zahájí ukončení spojení odesláním segmentu TCP s nastaveným příznakem FIN (dokončit) na server. Příznak FIN označuje, že klient dokončil odesílání dat a chce ukončit svůj konec připojení.
- **Krok 2. Server odpoví segmentem ACK (potvrzení)**: Po přijetí segmentu FIN od klienta server potvrdí požadavek odesláním segmentu ACK. Příznak ACK potvrzuje příjem FIN segmentu klienta. Server může mít stále data k odeslání, takže okamžitě neuzavře svůj konec připojení.
- **Krok 3. Server odešle segment FIN**: Jakmile server dokončí odesílání dat, odešle klientovi svůj vlastní TCP segment s nastaveným příznakem FIN. To znamená, že server dokončil přenos dat a chce ukončit svůj konec připojení.
- **Krok 4. Klient odpoví pomocí segmentu ACK**: Po přijetí segmentu FIN ze serveru klient potvrdí požadavek odesláním segmentu ACK. Tím potvrdíte požadavek serveru na ukončení spojení.

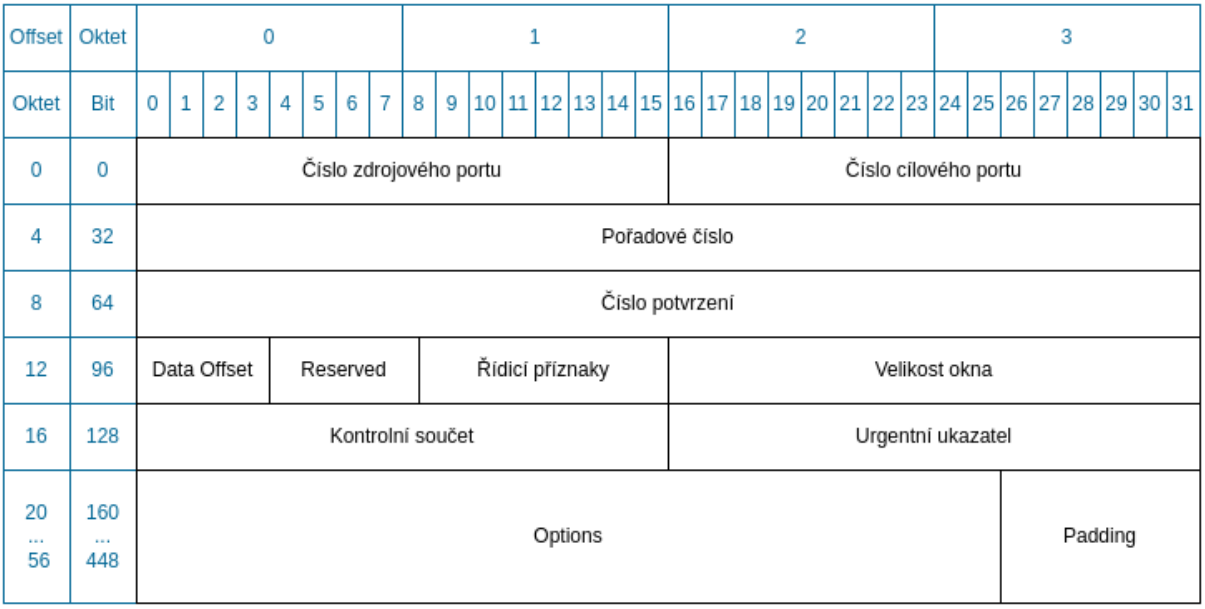
  V tomto okamžiku je čtyřcestný handshake dokončen a TCP spojení je ukončeno. Klient

i server potvrdili příjem segmentů FIN a dokončili příslušné datové přenosy. Připojení je nyní uzavřeno a obě strany mohou uvolnit jakékoli prostředky spojené s připojením. [18]

Účelem čtyřkrokového handshake je zajistit, aby obě strany dokončily své datové přenosy a aby během procesu ukončení nedošlo ke ztrátě dat. Výměnou segmentů FIN a ACK mohou klient a server komunikovat své záměry ukončit připojení a zajistit, aby všechna čekající data byla úspěšně přenesena před úplným uzavřením připojení. To pomáhá udržovat integritu výměny dat a zabraňuje potenciálním záměnám nebo ztrátě dat během ukončení připojení.

Struktura segmentu TCP

Segment TCP je základní jednotkou výměny dat v TCP spojení. Zapouzdřuje část dat spolu s řídicími informacemi nezbytnými pro spolehlivou a uspořádanou komunikaci mezi zařízeními po síti.



Obrázek 1.8: Struktura segmentu TCP

Každý TCP segment se skládá z hlavičky a datové zprávy.

- **Číslo zdrojového portu (16 bitů)**: Toto pole uvádí číslo portu aplikace nebo služby odesílatele. Pomáhá přijímači směrovat segment do správné aplikace na jeho konci.
- **Číslo cílového portu (16 bitů)**: Toto pole udává číslo portu aplikace nebo služby zamýšleného příjemce.
- **Pořadové číslo (32 bitů)**: Pořadové číslo se používá ke sledování pořadí datových segmentů v rámci přenosu. Pomáhá přijímači znovu sestavit data ve správném pořadí.
- **Číslo potvrzení (32 bitů)**: Toto pole nese číslo potvrzení pro další očekávané pořadové číslo, které přijímač očekává. Potvrzuje příjem dat do tohoto pořadového čísla.

- **Data Offset (4 bity)**: Toto pole, známé také jako délka hlavičky, udává délku hlavičky TCP ve 32bitových slovech. Používá se k určení, kde začíná užitečné zatížení dat.
- **Reserved (4 bits)**: Tyto bity jsou rezervovány pro budoucí použití a jsou obvykle nastaveny na nulu.
- **Řídicí příznaky (8 bitů)**: TCP používá řadu řídících příznaků k přenosu informací o účelu segmentu a podmínkách řízení. Některé z klíčových příznaků zahrnují:
  - **URG (Urgent)**: Označuje, že pole Urgentní ukazatel je významné a ukazuje na data, která vyžadují okamžité zpracování.
  - **ACK (Acknowledgement)**: Potvrzuje přijetí dat specifikovaných číslem potvrzení.
  - **PSH (Push)**: Požaduje přijímající stranu, aby doručila data do aplikace co nejdříve, aniž by čekala na plnou vyrovnávací paměť.
  - **RST (Reset)**: Resetuje připojení, pokud je zjištěna chyba nebo abnormální stav.
  - **SYN (Synchronize)**: Zahájí proces navázání spojení.
  - **FIN (Dokončit)**: Zahájí proces ukončení připojení.
- **Velikost okna (16 bitů)**: Velikost okna udává dostupnou vyrovnávací paměť přijímače pro příjem dat. Pomáhá regulovat řízení toku tím, že informuje odesílatele, kolik dat může přenést, než čeká na potvrzení.
- **Kontrolní součet (16 bitů)**: Pole kontrolního součtu obsahuje hodnotu vypočítanou přes TCP hlavičku, datovou zprávu, zdrojové a cílové IP adresy. Používá se pro detekci chyb k zajištění integrity dat během přenosu.
- **Urgentní ukazatel (16 bitů)**: Pokud je nastaven příznak URG, toto pole ukazuje na poslední urgentní datový bajt v datové užitečné zátěži. Používá se k identifikaci dat, která vyžadují okamžitou pozornost.
- **Možnosti (proměnná délka)**: Toto pole obsahuje různé volitelné parametry a řídicí informace, jako je maximální velikost segmentu (MSS), informace o selektivním potvrzení (SACK), časová razítka a další.
- **Padding (proměnná délka)**: Padding se používá k zajištění toho, že hlavička TCP končí na 32bitové hranici.

Struktura segmentu TCP umožňuje spolehlivou, uspořádanou a efektivní výměnu dat mezi zařízeními v síti. Pole hlavičky poskytují nezbytné řídicí informace pro řízení přenosu dat, potvrzení, řízení toku a detekci chyb. Analýzou různých oblastí v rámci segmentu TCP mohou síťoví analytici získat přehled o chování sítě, diagnostikovat problémy a optimalizovat výkon sítě. [17] [18]

Důležité aspekty pro síťovou analýzu

Pokud jde o analýzu sítě, pochopení vlastností TCP (Transmission Control Protocol) je zásadní pro diagnostiku a řešení problémů souvisejících se sítí. Pro analýzu síťového provozu jsou významné

následující vlastnosti:

- **Pořadová čísla a potvrzení**: Pořadová čísla a čísla potvrzení hrají klíčovou roli při sledování pořadí přenášených datových segmentů a potvrzování příjmu dat. Analýza těchto čísel může pomoci identifikovat chybějící, duplicitní nebo nefunkční segmenty, což může poskytnout pohled na přetížení sítě, ztrátu paketů nebo nesprávně nakonfigurovaná síťová zařízení.
- **Velikost okna**: Velikost okna je důležitý parametr pro řízení toku v TCP. Analýza velikosti okna může odhalit, kolik dat je přijímač ochoten přijmout, než bude vyžadovat potvrzení. Malá velikost okna může znamenat přetížení sítě nebo omezené zdroje přijímače.
- **Odhad RTT (Round-Trip Time)**: TCP používá odhad RTT k určení vhodných časových limitů pro opakovaný přenos. Monitorování RTT může pomoci identifikovat odchylky ve zpoždění sítě, které mohou ovlivnit doručování dat a ovlivnit celkový výkon sítě.
- **Frekvence a zpoždění ACK**: Analýza frekvence a zpoždění potvrzení může poskytnout pohled na odezvu přijímače. Zpožděná potvrzení ACK nebo mezery v potvrzovacích vzorcích mohou naznačovat problémy se zpracováním přijímače nebo přetížení sítě.
- **Chování při opakovaném přenosu**: Mechanismus opakovaného přenosu TCP je zásadní pro zajištění spolehlivého doručení dat. Analýza vzorů opakovaného přenosu může pomoci identifikovat segmenty, které nebyly úspěšně doručeny, a poskytnout vodítka o zahlcení sítě, ztrátě paketů nebo vadném síťovém vybavení.
- **TCP příznaky**: TCP segmenty obsahují příznaky, které indikují různé kontrolní podmínky, jako je SYN (navázání spojení), ACK (potvrzení), FIN (ukončení spojení) a další. Analýza těchto příznaků může pomoci diagnostikovat problémy související s nastavením připojení, ukončením nebo jinými událostmi řízení.
- **Délky segmentů**: Monitorování délek segmentů TCP může odhalit vzory v přenosu dat a pomoci identifikovat potenciální neefektivitu, fragmentaci nebo problémy související s konfigurací maximální velikosti segmentu (MSS).
- **Řízení toku a přetížení**: Pochopení toho, jak fungují mechanismy řízení toku a přetížení TCP, může poskytnout pohled na to, jak se protokol přizpůsobuje podmínkám sítě. Analýza úprav velikosti okna a algoritmů řízení přetížení může pomoci diagnostikovat přetížení sítě nebo úzká místa výkonu.
- **Chování zpožděného potvrzení ACK**: Analýza chování zpožděného potvrzení může pomoci identifikovat scénáře, kdy by zpožděná potvrzení ACK mohla ovlivnit celkovou propustnost dat a odezvu.
- **Stavy připojení**: Připojení TCP procházejí během svého životního cyklu různými stavy, včetně UZAVŘENO, POSLECHNUTO, SYN-ODESLANO, ESTABLISHED, FIN-WAIT a dalších. Sledování stavů připojení může pomoci identifikovat zablokovaná připojení, napůl otevřená připojení nebo jiné abnormální chování.

- **Informace záhlaví segmentu**: Analýza různých polí v záhlaví segmentu TCP, jako jsou čísla zdrojových a cílových portů, může poskytnout pohled na aplikace a služby zapojené do komunikace.

Zkoumáním těchto vlastností a chování TCP mohou síťoví analytici přesně určit problémy, řešit problémy s připojením, optimalizovat výkon sítě a zajistit spolehlivý přenos dat v různých síťových prostředích. Nástroje a techniky síťové analýzy se často zaměřují na zachycení a analýzu těchto parametrů specifických pro TCP, aby poskytly komplexní pohled na chování a výkon sítě. [19]

1.7 UDP

UDP je zkratka pro User Datagram Protocol (RFC 768). Jedná se o komunikační protokol používaný v počítačových sítích, který funguje na transportní vrstvě. UDP je navržen tak, aby poskytoval lehkou metodu s nízkou režií pro odesílání datagramů (paketů dat) mezi zařízeními v síti. Na rozdíl od TCP, UDP neposkytuje stejnou úroveň spolehlivosti a kontroly chyb, ale nabízí vyšší přenosové rychlosti a běžně se používá pro aplikace, kde je rychlost a nízká latence důležitější než zaručené doručení. [20] [21]

UDP nevyžaduje připojení, což znamená, že nenavazuje formální spojení mezi odesílatelem a příjemcem před odesláním dat. S každým paketem UDP, známým také jako datagram, se zachází nezávisle a k dosažení svého cíle se může vydat jinou cestou. Na rozdíl od TCP neposkytuje UDP záruky, že data budou doručena úspěšně nebo ve správném pořadí. Nezahrnuje mechanismy pro opakované vysílání ztracených paketů nebo přeskupování paketů mimo pořadí. [21]

Mezi aplikace, které běžně používají protokol UDP, patří:

- **Online hry**: Rychlé online hry vyžadují nízkou latenci, díky čemuž je UDP vhodnou volbou pro rychlé odesílání herních dat, i když občas dojde ke ztrátě paketů.
- **Streamování**: Služby streamování videa a zvuku v reálném čase často používají protokol UDP k poskytování mediálního obsahu s minimálním zpožděním.
- **VoIP**: Služby Voice over IP, jako je Skype nebo Zoom, mohou používat protokol UDP k udržení konverzace v reálném čase.
- **DNS**: Systém DNS (Domain Name System) používá protokol UDP pro rychlé rozlišení názvu domény.
- **IoT (Internet of Things)**: Zařízení IoT, která odesílají malé a časté aktualizace dat, mohou těžit z nízké režie a rychlosti UDP.

Struktura UDP hlavičky

Struktura datagramu UDP je relativně přímočará. Skládá se z hlavičky pevné velikosti následované datovou zprávou.

Offset	Oktet	0								1								2								3							
Oktet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Zdrojový port																Cílový port															
4	32	Délka																Kontrolní součet															

Obrázek 1.9: Struktura UDP hlavičky

- **Zdrojový port (16 bitů)**: Toto pole určuje číslo portu odesílající aplikace nebo procesu. Pomáhá přijímajícímu konci zjistit, která aplikace na straně odesílatele zahájila komunikaci.
- **Cílový port (16 bitů)**: Toto pole udává číslo portu přijímající aplikace nebo procesu na cílovém zařízení. Přesměruje příchozí datagram do správné aplikace.
- **Délka (16 bitů)**: Pole délky udává celkovou délku datagramu UDP, včetně hlavičky a datového obsahu. Měří se v bajtech. Protože minimální velikost hlavičky UDP je 8 bajtů, minimální hodnota pro toto pole je 8. Maximální hodnota je 65 535 bajtů.
- **Kontrolní součet (16 bitů)**: Pole kontrolního součtu je volitelné, ale běžně se používá k detekci chyb v datagramu UDP. Vypočítává se pro celý datagram (včetně záhlaví i dat) a používá se k ověření integrity dat během přenosu. Pokud není pole kontrolního součtu použito, je nastaveno na samé nuly.

UDP má ve srovnání s TCP minimální režii. Nemá rozsáhlé mechanismy handshake, sekvenování a potvrzování, které má TCP. Výsledkem je rychlejší přenos dat. Díky absenci komplexních mechanismů kontroly chyb a obnovy nabízí UDP nižší latenci ve srovnání s TCP. [19] [20] [21]

2 Analýza síťového provozu

2.1 Historický pohled do síťové analýzy

Analýza síťového provozu se v dnešní době stává stále více důležitější pro monitorování síťového provozu. V minulých letech administrátoři monitorovali pouze malý počet síťových zařízení (typicky méně než tisíc počítačů), šířka pásma sítě byla řádově 100 Mbps (megabitů za sekundu). [6]

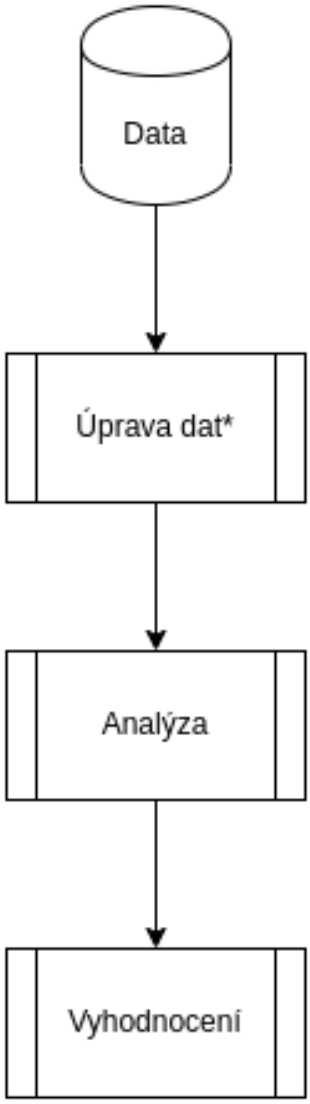
V současné době se administrátoři musí vypořádat s vysokorychlostními sítěmi a různými sítěmi, jako je síť ATM (Asynchronous Transfer Mode) a i bezdrátové sítě. [6]

Asynchronous Transfer Mode (ATM) je síťová technologie, která byla populární na konci 80. let a v průběhu 90. let 20. století. Technologie byla navržena tak, aby podporovala vysokorychlostní komunikaci (dle doby) a efektivní přenos různých typů dat, včetně hlasu, videa a datového provozu, přes jedinou síť. [22]

Síť může být analyzována na různých úrovních, např. na úrovni paketů, na úrovni toku a na úrovni sítě pro správu zabezpečení. Pro analýzu síťového provozu používají výzkumníci různé techniky. Obecný rámec pro analýzu síťového provozu zahrnuje předběžné zpracování, po kterém následuje skutečná analýza a pozorování k odhalení vzorů ze síťových dat. Níže uvedený obrázek ukazuje tři hlavní fáze analýzy síťového provozu. [19] [7]

2.2 Fáze síťové analýzy

Analýzu síťového provozu můžeme rozdělit do následujících fází:



Obrázek 2.1: Fáze analýzy

Data

Získání dat může být provedeno přímo ze síťového prvku, popř. kolektorem napojeným na síťový prvek.

Úprava dat

Úprava dat (preprocessing) je volitelnou fází síťové analýzy. Cílem této fáze je transformovat data do srozumitelného formátu. Skutečná data jsou často neúplná a nekonzistentní (obsahují chyby, odlehlé hodnoty), a proto jsou před analýzou dat využívány metody na jejich úpravu, aby se zlepšila kvalita dat, čímž se napomáhá zvýšit přesnost a účinnost výsledné analýzy. Úpravu dat můžeme rozdělit do dvou metod.

Metoda diskretizace (Discretization method)

Diskretizace je proces mapování spojitých atributů na nominální atributy. Hlavním cílem procesu diskretizace je objevit množinu bodů řezu, které rozdělují rozsah na malý počet intervalů. Každý bod přerušení (cut-point) je skutečná hodnota v rozsahu spojitých hodnot, což rozděluje rozsah na dva intervaly, jeden je větší než bod přerušení a druhý je menší nebo roven hodnotě bodu přerušení. Techniky diskretizace dat lze použít ke snížení počtu hodnot spojitého atributu rozdělením rozsahu atributu do malých intervalů. Popisky intervalů lze použít k nahrazení skutečné hodnoty atributů. Nahrazení hodnot spojitého atributu malým počtem intervalových štítků snižuje a zjednodušuje původní data set. Diskretizační metody lze rozdělit do čtyř kategorií. [6] [7]

- **Metoda pod dozorem a bez dozoru (Supervised and Unsupervised method)**: Metoda pod dozorem a bez dozoru využívají označení tříd prostřednictvím procesu diskretizace. Metody bez dozoru nevyužívají informace o štítcích tříd a generují metody diskretizace sdílením hodnot číselných atributů. [6] [7]
- **Globální a lokální metoda (Global and Local method)**: Globální metody využívají pro diskretizaci celé numerické atributy. Místní metody však při získávání diskretizace používají podmnožinu instancí [6] [3]
- **Top-down rozdělení a Bottom-up sloučení (Top-down splitting and Bottom-up merging)**: Top-down rozdělení diskretizační metody začínají nahoře s některými obecnými informacemi a končí specializovanými informacemi. Metody bottom-up však začínají hlavním počtem dílčích intervalů a kombinují tyto dílčí intervaly, dokud není dosaženo optimálního počtu intervalů. [6] [2]
- **Přímá a inkrementální metoda (Direct and Incremental method)**: Přímé metody rozdělují rozsah hodnot na stejný počet intervalů a uživatelé mohou určit počet intervalů Inkrementální metody začínají jednoduchou diskretizací a pokračují během procedury vylepšení, dokud nedosáhnou dobré diskretizace a poté zastaví proces diskretizace. [6] [7]

Metoda výběru funkcí (Feature Selection method)

Tato metoda se používá hlavně ke snížení velikosti datové sady pro zlepšení analýzy síťového provozu. Mezi tyto metody můžeme zařadit analýzu hlavních komponent, informační entropii a teorii hrubých množin, které jsou nejčastěji používané pro skutečnou analýzu síťového provozu. [6] [7]

- **Analýza hlavních komponent (PCA, principal component analysis)**: PCA v podstatě transformuje původní vlastnosti datové sady do nové sady nekorelovaných proměnných nazývaných hlavní komponenty. Tyto hlavní složky jsou lineárními kombinacemi původních vlastností a jsou seřazeny podle důležitosti při vysvětlení rozptylu přítomného v datech. [6] [7]

- **Informační entropie**: V kontextu teorie informace entropie kvantifikuje množství informací nebo „překvapení“ obsažených v daném souboru dat. Čím vyšší je entropie, tím jsou data nejistější nebo nepředvídatelnější a naopak. Používá se k charakterizaci průměrného množství informací potřebných k popisu události čerpané z rozdělení pravděpodobnosti. [6] [7]
- **Teorie hrubých množin (Rough Set theory)**: Teorie je užitečná zejména v oblasti umělé inteligence, strojového učení, analýzu dat a rozpoznávání vzorů. Hlavní myšlenkou teorie hrubých množin je reprezentovat znalosti pomocí množin objektů a jejich atributů. V hrubé sadě je objekt definován svými charakteristikami nebo atributy a tyto atributy jsou rozděleny do dvou typů: [6] [7]
  - Určité atributy: Jedná se o atributy, které jednoznačně určují objekt. Jinými slovy, pokud mají dva objekty stejnou hodnotu pro všechny určité atributy, jsou s ohledem na tyto atributy považovány za stejné. [6] [23]
  - Nejisté (nebo neurčité) atributy: Tyto atributy neurčují objekt jednoznačně. Různé objekty mohou mít pro tyto atributy stejnou hodnotu, což vnáší nejistotu do reprezentace znalostí. [6] [3]

Analýza

Analýza síťového provozu lze rozdělit do čtyř hlavních kategorií - techniky shlukování, klasifikace, hybridních a asociačních pravidel. [7]

Technika shlukování

Technika shlukování, také známe pod pojmem clustering je proces rozdělení dat do skupin podle určitých charakteristik dat. Každá skupina, nazývaná cluster, se skládá z členů, kteří jsou si dosti podobní a členové z různých shluků se od sebe liší. Clusterovací metody se používají pro vytváření skupin síťových dat pro analýzu síťového provozu. Zjistili jsme, že výzkumníci používají několik algoritmů shlukování dat. Hledání správného nejlepšího algoritmu v této kategorii však není jednoduché. [7]

M. Vijayakumar navrhl bi-section K-Means algoritmus pro analýzu síťového provozu. Název jasně napovídá, že algoritmus je založený na algoritmu K-means. Vhodný shluk je identifikován a dále rozdělen pomocí hierarchických technik k získání dalších shluků. Podle M. Vijayakumara je tento bi-section K-Mean clustering lepší než čistě hierarchická metoda clusteringu z hlediska analýzy provozu. [24]

Jeffrey Erman navrhl tři shlukovací algoritmy, K-means, Auto class a DBSCAN pro klasifikaci síťového provozu. J. Erman shromáždil trasování paketů ze trasování Auckland IV a trasování univerzity v Calgary s různým časovým intervalem. Použil algoritmy Kmeans, Auto class a DBSCAN k vyhodnocení účinnosti každého algoritmu pro klasifikaci provozu. Z výsledku dospěl k závěru, že algoritmus Auto Class fungoval lépe pro klasifikaci síťového provozu. [25]

Klasifikační techniky

Klasifikace je forma analýzy dat, která vezme každou instanci datové sady a přiřadí ji konkrétní třídě. Analýza síťového provozu založená na klasifikaci se pokouší klasifikovat veškerý provoz jako normální nebo škodlivý. Úkolem klasifikace je snížit počet falešně pozitivních (detekce normálního síťového provozu jako abnormálního) a falešně negativních (detekce škodlivého síťového provozu jako normální). V této podsekci uvedu několik klasifikačních algoritmů pro analýzu dat, které výzkumníci používají. [3]

- **Support Vector Machine (SVM)**

  Support vector machine je metoda učení pod dohledem používaná pro klasifikaci a regresi. Support Vector Machine byl vytvořen pomocí Vapnik support Vector Machine pro trénování a testování velkého množství vysoce rozměrných dat. SVM je velmi nákladný z hlediska času a spotřeby paměti. Řada současných studií uvádí, že výsledky SVM poskytují vyšší přesnost s ohledem na výkon než jiné klasifikační přístupy. [26]

  Ghanshyam Prasad Dubey navrhl přístupy RST a inkrementální SVM k detekci narušení. Dubey experimentoval s datovým souborem KDD. Výběr významných atributů ze souboru dat o síťovém provozu je dokončen pomocí metody RST a následně zpracován přístupem SVM pro učení a testování. Z analýzy dat je prezentováno srovnání mezi inkrementální SVM a neinkrementální SVM. Podle Dubey přístup inkrementálního SVM zvýšil výkon pro detekci narušení. Přístupy RST a inkrementální SVM jsou účinné pro snížení prostorové hustoty dat. [27]

- **Neuronová síť**

  Neuronová síť se skládá z kolekce procesních neuronů, které jsou vzájemně propojené a transformují sadu vstupů na sadu požadovaných výstupů. Výsledek transformace je rozpoznán podle charakteristik neuronů a vah spojených s korelací mezi nimi. Zdokonalením propojení mezi uzly je síť schopna získat požadované výstupy. [7]

  Khaled Al-Nafjan představil modulární neuronovou síť (MNN) pro detekci narušení. Al--Nafjan použil KDD dataset a nejvýznamnější atributy z celého souboru dat o síťovém provozu získali pomocí metody PCA. Použil MNN k detekci narušení sítě. Z analýzy výsledku vyhodnotil, že MNN s přístupem PCA snížilo RMSE (střední kvadratická odchyblka) na 0,1. [28]

  Shilpa Lakhina PCA pro identifikaci jakéhokoli druhu nových útoků. Přístup PCA se používá ke snížení rozměrnosti souboru dat pro zlepšení přesnosti. Test a porovnání se prováděl na datovém souboru KDD. Dále vybral 8 atributů z celé sady 41 atributů. Výsledky klasifikace pomocí hybridního modelu s 41 atributy a 8 atributy porovnal a dospěl k závěru, že klasifikace po předzpracování PCA poskytuje lepší výkon. [29]

- **Rozhodovací stromy**

  Rozhodovací strom je výkonné a oblíbené strojové učení pod dozorem pro klasifikační problémy. Rozhodovací strom se používá v řadě aplikací v reálném životě, jako je lékařská

diagnostika, předpověď počasí, schvalování úvěrů a detekce narušení atd. Rozhodovací strom může díky velikosti stromu používat velký objem dat s mnoha atributy. Je nezávislý na velikosti datové sady. Rozhodovací strom se skládá z uzlů, listů a hran. Každý uzel je označen atributem on proti všem atributům, podle kterých mají být data rozdělena. Každý uzel má určitý počet hran podle možných hodnot atributu. Listy jsou označeny rozhodovací hodnotou pro klasifikaci dat. Každý rozhodovací strom odpovídá sadě pravidel, která klasifikuje data podle atributů datové sady. Budovací algoritmy rozhodovacích stromů mohou nejprve vytvořit strom a poté jej oříznout pro efektivnější klasifikaci. Pomocí techniky prořezávání mohou být části stromu odstraněny nebo kombinovány, aby se zmenšila celková velikost stromu, čas a spotřeba paměti rozhodovacího stromu závisí na velikosti souboru dat. [6]

Jiong Zhang použil algoritmus náhodného lesa bez dozoru k detekci odlehlých hodnot v sadě dat síťového provozu. Detekce odlehlých hodnot se používá pro účely narušení sítě. Zhang experimentoval s datovým souborem KDD. Dospěl k závěru, že přístup k detekci odlehlých hodnot bez dozoru měl za následek méně falešně pozitivních případů ve srovnání s algoritmy klasifikace pod dohledem neuronové sítě (NN) a SVM. [30]

- **Hybridní modely**

  Hybridní modely jsou kombinací dvou nebo více přístupů pro analýzu síťového provozu. Hybridní model dosahuje dobrých výsledků v analýze síťového provozu. [23]

- **Asociační pravidla**

  Asociační pravidlo považuje každý pár atribut/hodnota za položku. Kolekce položek označovaných jako sada položek v jediném síťovém požadavku. Asociační pravidla se používají k identifikaci vzoru nebo vztahu mezi atributem databáze. [7]

Vyhodnocení

Pro vyhodnocování je možné použít mnoho různých metrik. Pro měření výkonu klasifikátoru pro různé datové sady se používají metriky míry detekce, četnosti falešných pozitivních nálezů, přesnosti a časových nákladů. K vyjádření přesnosti predikce existuje řada metrik. Použité metriky pomocí matoucí matice. [2]

- **True Negatives (TN)**: Celkový počet správně klasifikovaných normálních paketů.
- **True Positives (TP)** Celkový počet správně klasifikovaných škodlivých paketů.
- **False Negatives (FN)**: Falešná negativa je celkový počet škodlivých paketů nesprávně klasifikovaných jako normální pakety.
- **False Positives (FP)**: Falešně pozitivní je celkový počet normálních paketů nesprávně klasifikovaných jako škodlivé pakety.
- **Míra detekce (MD)**: Je poměr celkového počtu detekovaných útoků dělený celkovým počtem falešně pozitivních plus celkový počet skutečně negativních.

- **Míra přesnosti (MP)**: Je podíl celkového počtu TP dělený celkovým počtem TP plus celkový počet FP.
- **Míra odvolání (MO)**: Je poměr celkového počtu TP dělený celkovým počtem TP plus celkový počet FN.
- **Celková rychlost (CR)**: Je podíl celkového počtu TP pulzů celkový počet TN dělený celkovým počtem TP plus celkový počet FP plus celkový počet plus celkový počet TN.
- **Citlivost (C)**: Je podíl celkového počtu TP dělený celkovým počtem FP.
- **Specifičnost (S)**: Je podíl celkového počtu TN dělený celkovým počtem FN.
- **Přesnost (P)**: Je poměr celkového počtu TP plus celkového počtu TN dělený celkovým počtem FP plus celkovým počtem FN.
- **Procento úspěšné predikce (PUP)**: Je poměr celkového počtu klasifikovaných úspěšných instancí dělený celkovým počtem skutečných instancí.

[7]

2.3 Klíčové aspekty síťové analýzy

Analýza síťového provozu má dopad i do jiných odvětví informačních technologií. Je možné ji využít i v jiných ohledech než při vyšetřování narušení bezpečnosti či anomálií sítě.

- **Monitorování výkonu**: Síťoví analytici sledují výkon sítě, aby zajistili její optimální fungování. Měří metriky, jako je využití šířky pásma, latence, ztrátovost paketů a propustnost, aby identifikovaly potenciální úzká místa nebo oblasti pro zlepšení. [6]
- **Odstraňování problémů a diagnostika**: Když se objeví problémy se sítí, analytici používají specializované nástroje k diagnostice a izolaci problémů. Mohou analyzovat soubory protokolu, zkoumat síťový provoz a používat analyzátory síťových protokolů (sniffery paketů) k určení zdroje problému. [2]
- **Plánování kapacity**: Analýzou trendů využití sítě a historických dat mohou síťoví analytici předpovídat budoucí požadavky na kapacitu. To pomáhá při přijímání informovaných rozhodnutí o upgradech nebo rozšiřování sítě za účelem splnění rostoucích požadavků. [7]
- **Síťové inženýrství**: Síťová analýza pomáhá při optimalizaci toku síťového provozu a správě kvality služeb (QoS), aby bylo zajištěno, že kritické aplikace obdrží nezbytnou šířku pásma a prioritu. [7]
- **Analýza protokolů**: Síťoví analytici často používají analyzátory protokolů k zachycení, dekódování a analýze síťových paketů. To jim umožňuje porozumět komunikaci mezi zařízeními a aplikacemi, což jim pomáhá identifikovat problémy nebo neefektivitu. [3]
- **Vizualizace sítě**: Vizuální reprezentace, jako jsou mapy topologie sítě nebo vývojové diagramy, pomáhají pochopit složité sítě a jejich propojení.

- **Forenzní analýza**: V případě bezpečnostního incidentu nebo narušení je pro digitální forenzní analýza nezbytná síťová analýza, která rekonstruuje sled událostí a identifikuje hlavní příčinu.
- **Optimalizace a vylepšení**: Na základě zjištění analýzy mohou síťoví analytici navrhnout a implementovat změny ke zlepšení výkonu sítě, bezpečnosti a celkové efektivity. [6]

2.4 Techniky skenování portů

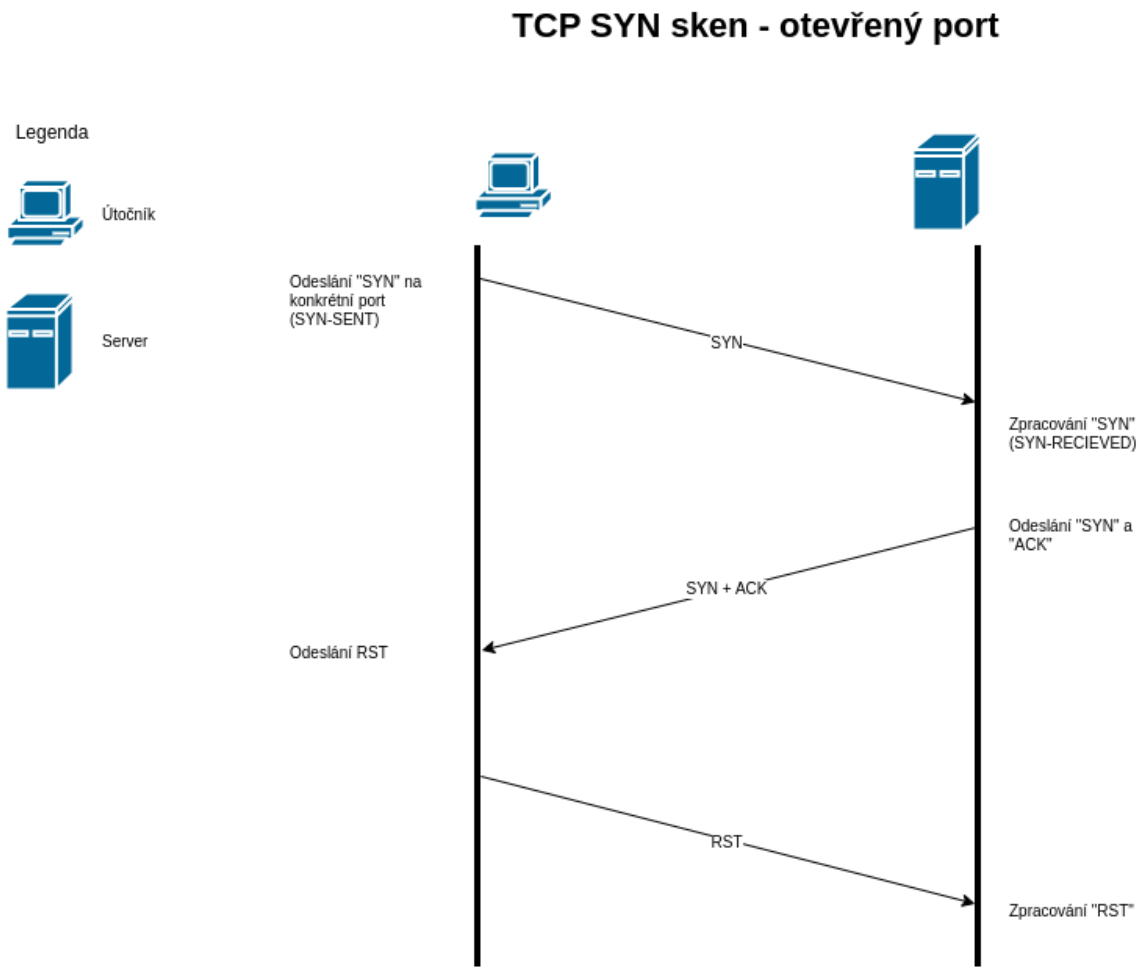
Skenování portů je klíčovou technikou v oblasti síťové analýzy, která umožňuje identifikovat otevřené porty na cílovém zařízení či síti. Při síťové analýze se techniky skenování portů využívají k detekci dostupných služeb a jejich konfigurací. Tyto techniky mohou být aktivní, což znamená přímý pokus o komunikaci s porty zařízení, nebo pasivní, kdy se analyzují síťové komunikace a sledují se odpovědi zařízení na neiniciované dotazy. Skenování portů poskytuje důležité informace pro bezpečnostní odborníky a správce sítí, protože umožňuje identifikovat potenciální bezpečnostní slabiny a monitorovat aktivity v síti. Zároveň může být použito i k identifikaci neautorizovaných pokusů o přístup či útoky, což posiluje celkovou síťovou bezpečnost.

V předchozích odstavcích jsme se ponořili do základních protokolů a síťových metadat, které pokládají základ pro efektivní komunikaci a výměnu dat v počítačových sítích. S tímto porozuměním nyní můžeme prozkoumat techniky skenování síťových portů. Techniky skenování portů se liší ve složitosti a utajení, od základních metod, jako je skenování TCP connect, až po pokročilejší přístupy, jako je skenování SYN, skenování FIN a skenování XMAS. Každá technika nabízí zřetelné výhody, pokud jde o rychlost a nenápadnost. [6]

TCP SYN

Tato technika je jedna z nejčastěji používaných a to z dobrého důvodu. Lze jí provádět rychle a skenovat tisíce portů za sekundu. SYN sken je relativně nenápadný protože nikdy nedokončuje připojení TCP. Funguje také proti jakémukoli vyhovujícímu TCP stacku, spíše než v závislosti na zvláštnostech konkrétních platforem. Umožňuje také jasné a spolehlivé rozlišení mezi otevřeným, uzavřeným a filtrovaným stavem. [6] [31]

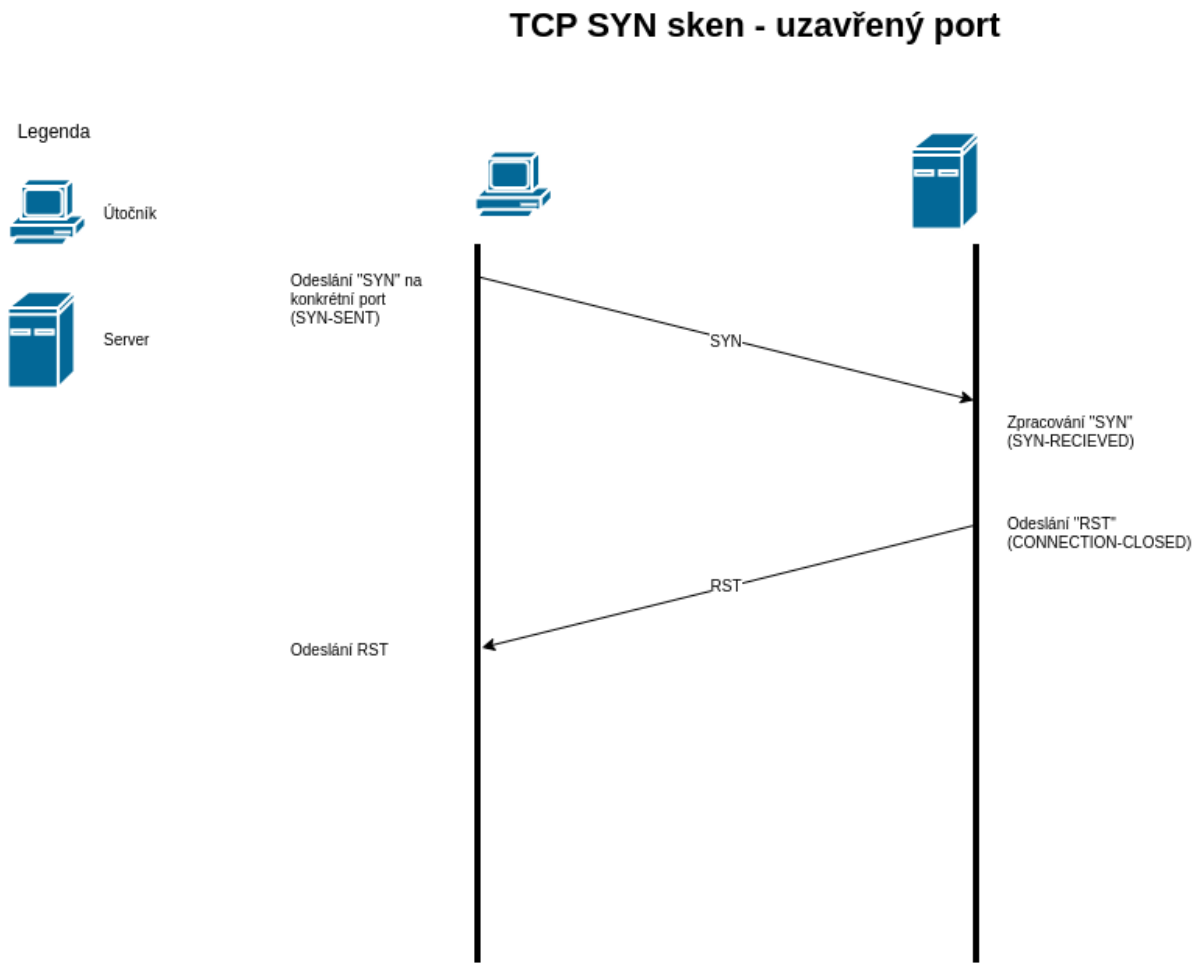
I když je skenování SYN docela snadné bez znalosti TCP na nízké úrovni. Ukažme si tento sken na diagramu na úrovni paketů. Nejprve chování vůči otevřenému portu.



Obrázek 2.2: TCP SYN sken - otevřený port

V prvním kroku dojde k odeslání TCP paketu s příznakem SYN - toto je první běžný krok pro navázání spojení. Protože je cílový port otevřený, server udělá druhý krok odesláním odpovědi s příznaky SYN a ACK útočníkovi. Pokud by se jednalo o běžné spojení, útočník by dokončil trojcestný handshake odesláním ACK. Protože nechceme navázat spojení (a poté se starat o jeho uzavření), hacker odešle RST packet. Toto řekne serveru, aby zapomněl (resetoval) pokus o připojení. Protože trojcestný handshake není nikdy dokončeno, skenování SYN se někdy nazývá polootevřené skenování. [31]

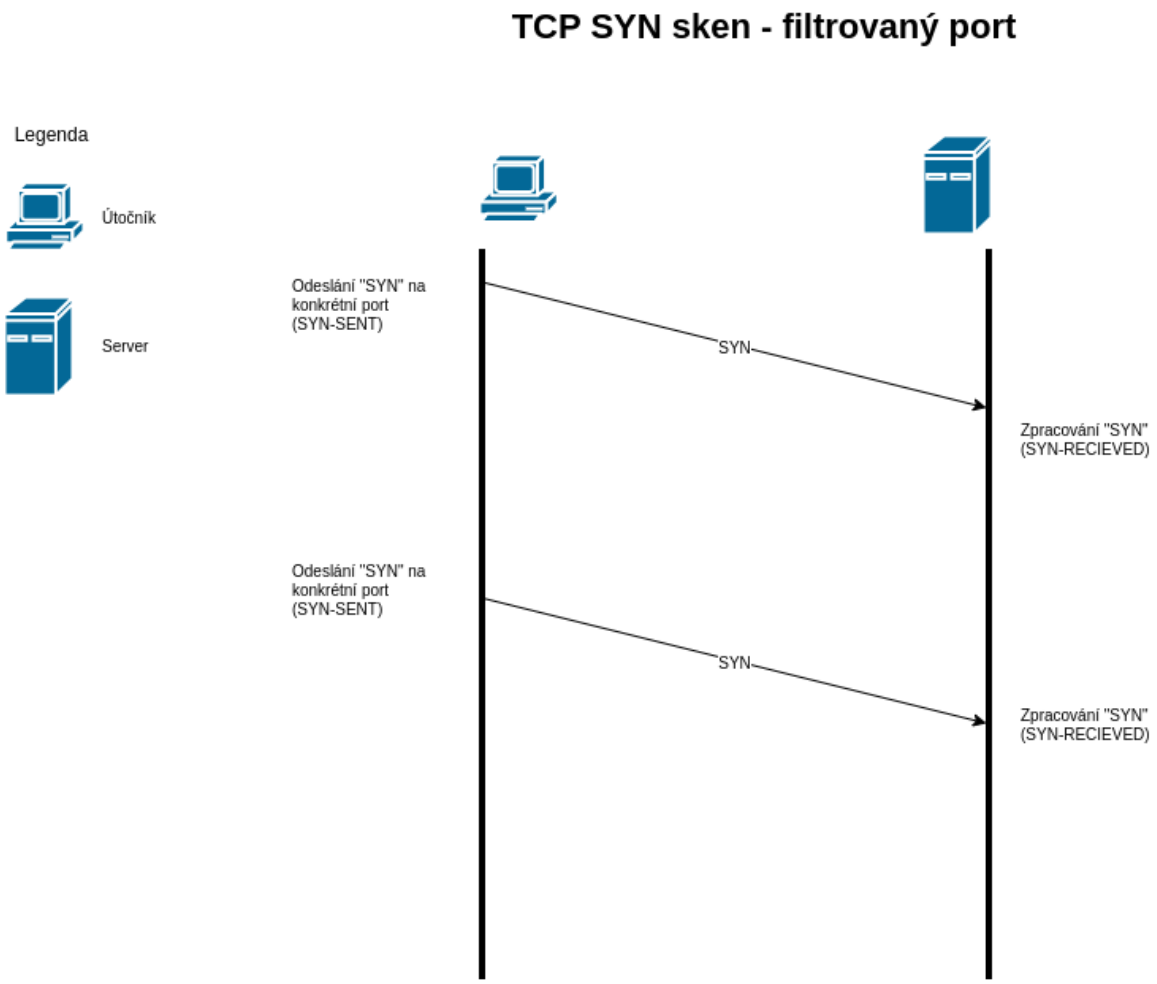
Ukažme si případ, kdy daný port není otevřen.



Obrázek 2.3: TCP SYN sken - uzavřený port

První krok je stejný, ale na místo zpětného přijetí SYN/ACK, vrací server RST. Zde máme jasný příznak, že port je uzavřen a žádná další komunikace ohledně tohoto portu není nutná. [31]

Ukažme si ještě poslední variantu s filtrovaným portem.

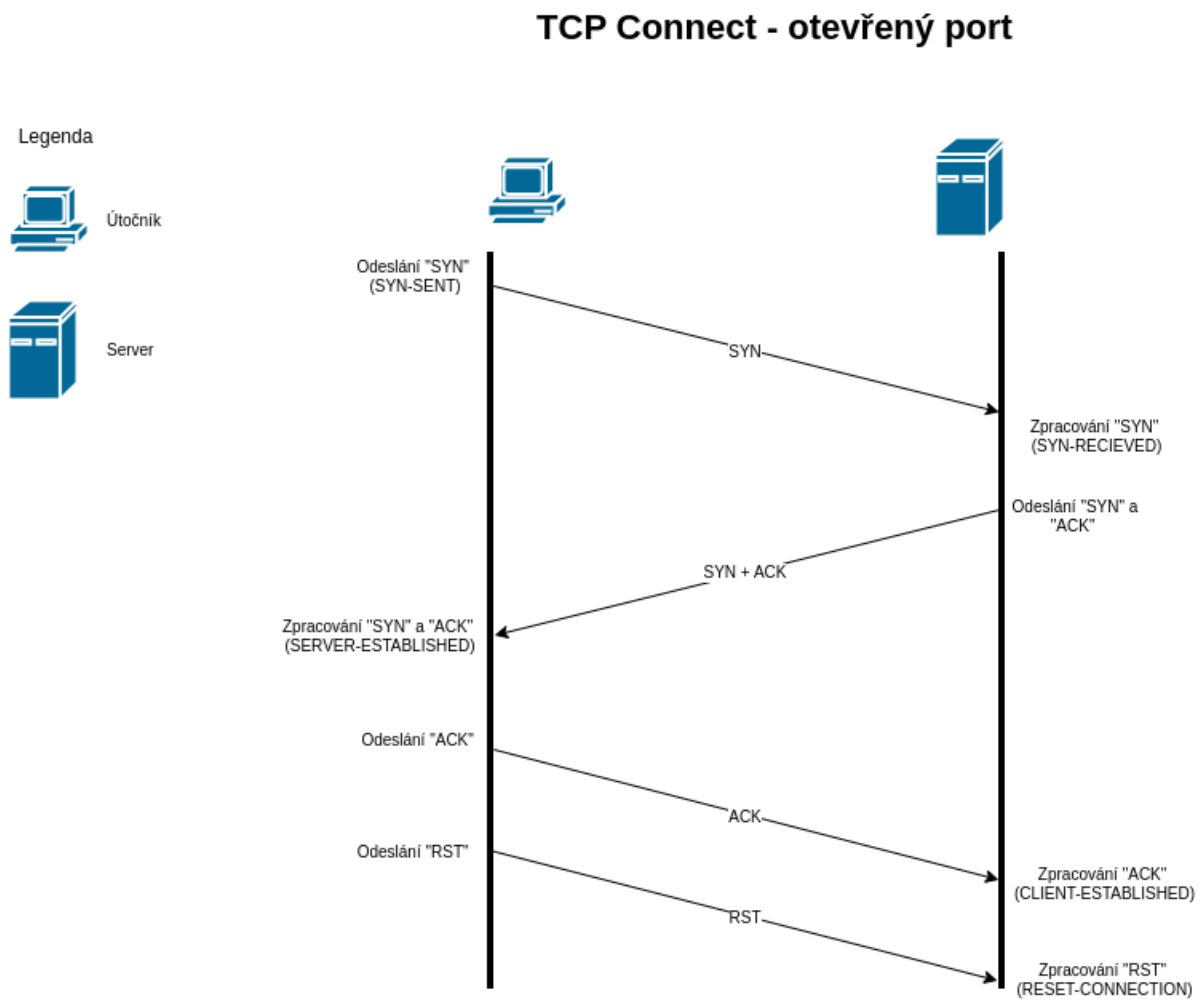


Obrázek 2.4: TCP SYN sken - filtrovaný port

První krok je stále stejný, ale na místo odpovědi od serveru nic nedostaneme. Port (server), který nereaguje, je obvykle filtrován (blokován firewallem). Zkusíme znovu provést první krok, abychom se přesvědčili a vyloučili výpadek sítě. Pokud server ani po několikanásobném odeslání SYN packetu neodpoví, jedná se o filtrovaný port. [31]

TCP Connect

Tato technika je velice podobné TCP SYN s rozdílem, že TCP connect dokončí trojcestný handshake. Jedná se tedy o stejné systémové volání, které používají např. webové prohlížeče, klienti P2P a většina dalších síťových aplikací k navázání spojení. Oproti předchozí technice, sken trvá déle a vyžaduje více paketů k získání stejných informací. [32]



Obrázek 2.5: TCP Connect - otevřený port

Zatímco tento příklad skenování připojení zabral téměř dvakrát tolik paketů než skenování SYN, rozdíly v šířce pásma jsou jen zřídka tak výrazné. Velká většina portů ve velkém skenování bude uzavřena nebo filtrována. Proč tedy využívat tuto techniku, když se zdá být absurdní oproti TCP SYN? [32]

- **Úplnost**: Skenování pomocí TCP connect je spolehlivější při detekci otevřených portů ve srovnání se skenováním SYN. Zavádí úplné tříkrokové navázání komunikace s cílovým systémem, což poskytuje jasnou indikaci, zda je port otevřený, uzavřený nebo filtrovaný.
- **Viditelnost**: Některé systémy a firewally jsou nakonfigurovány tak, aby s pakety SYN zacházely odlišně, což může vést k nepřesným výsledkům. Skenování připojení TCP je méně pravděpodobné, že bude těmito konfiguracemi ovlivněno.
- **Protokolování a auditování**: Vzhledem k tomu, že je navázáno úplné připojení, skenování připojení TCP může generovat komplexnější protokoly v cílovém systému. To může být výhodné pro účely auditu a sledování. Toto se nezdá jako výhoda pro útočníka, pokud nepoužije další metodiku spoofingu MAC adresy nebo IP adresy.

[6] [32]

TCP FIN, NULL, PSH, URG

Dle RFC 793 - „pokud je cílový stav portu 'není otevřen', příchozí segment, který neobsahuje RST, způsobí odeslání RST jako odpověď.". Při skenování systémů vyhovujících tomuto textu RFC bude mít v případě uzavřeného portu každý paket neobsahující bity SYN, RST nebo ACK za následek vrácený RST. V případě otevřeného portu, neobdrží žádnou odpověď. Pokud není zahrnut žádný z těchto tří bitů, jakákoliv kombinace ostatních tří (FIN, PSH a URG) je v pořádku. [33]

Klíčovou výhodou těchto typů skenování je, že se mohou propašovat přes určité nestavové firewally a směrovače pro filtrování paketů. Takové firewally se snaží zabránit příchozím TCP spojením (a zároveň povolit odchozí) blokováním jakýchkoli TCP paketů s nastaveným bitem SYN a vymazaným ACK. Tato konfigurace je natolik běžná, že příkaz linuxového firewallu iptables nabízí k její implementaci speciální volbu '–syn'. Skenování NULL, FIN, PSH anebo URG vymaže bit SYN a obejde tak přímo tato pravidla. [33]

Velkou nevýhodou je, že ne všechny systémy dodržují RFC 793. Řada systémů odesílá odpovědi RST sondám bez ohledu na to, zda je port otevřený nebo ne. To způsobí, že všechny porty budou označeny jako uzavřené. Hlavní operační systémy, které to dělají, jsou Microsoft Windows, mnoho zařízení Cisco a IBM. [6] [33]

TCP ACK

Tento sken se liší od ostatních dosud představených v tom, že nikdy neurčuje otevřené anebo filtrované porty. Používá se k mapování sad pravidel brány firewall, určení, zda jsou stavové či nikoli, a které porty jsou filtrovány. Při skenování nefiltrovaných systémů vrátí otevřené i uzavřené porty paket RST. Toto znamená, že jsou dosažitelné ACK paketem, ale není určeno, zda jsou otevřené nebo filtrované. Porty, které neodpovídají nebo odesílají určité chybové zprávy ICMP zpět, jsou označeny jako filtrované. [6]

Jedním z nejzajímavějších použití skenování ACK je rozlišení mezi stavovými a bezstavovými firewally. Firewally, které blokují sondu, na druhou stranu obvykle neodpovídají nebo pošlou zpět chybu ICMP nedosažitelnosti cíle. Toto rozlišení umožňuje zjistit, zda jsou ACK pakety filtrovány. Sada filtrovaných portů hlášená skenováním Nmap ACK je často menší než pro skenování SYN na stejném počítači, protože skenování ACK je obtížnější filtrovat. Mnoho sítí umožňuje téměř neomezená odchozí připojení, ale chtějí blokovat internetové hostitele v inicializaci připojení zpět k nim. Blokování příchozích paketů SYN (bez nastaveného bitu ACK) je snadný způsob, jak toho dosáhnout, ale stále umožňuje průchod všem paketům ACK. Blokování těchto ACK paketů je obtížnější, protože neříkají, která strana zahájila spojení. Nevýhodou je, že ke svému fungování vyžadují více zdrojů a restartování firewallu ve stavu může způsobit ztrátu stavu zařízení a ukončení všech navázaných spojení, která přes něj procházejí. [34] [7]

TCP FIN/ACK

TCP FIN/ACK je také přezdívaný jako „Maimonův sken“ (Maimon), pojmenován po svém objeviteli Urielu Maimonovi. Techniku popsal ve vydání časopisu Phrack č. 49 (listopad 1996). Tato technika je přesně stejná jako skenování NULL, FIN, URG anebo PSH, až na to, že sonda je FIN/ACK. Podle RFC 793 (TCP) by měl být paket RST generován jako odpověď na takovou sondu, ať už je port otevřený nebo zavřený. Uriel si však všiml, že mnoho systémů odvozených od BSD jednoduše zahodí paket, pokud je port otevřený. [35]

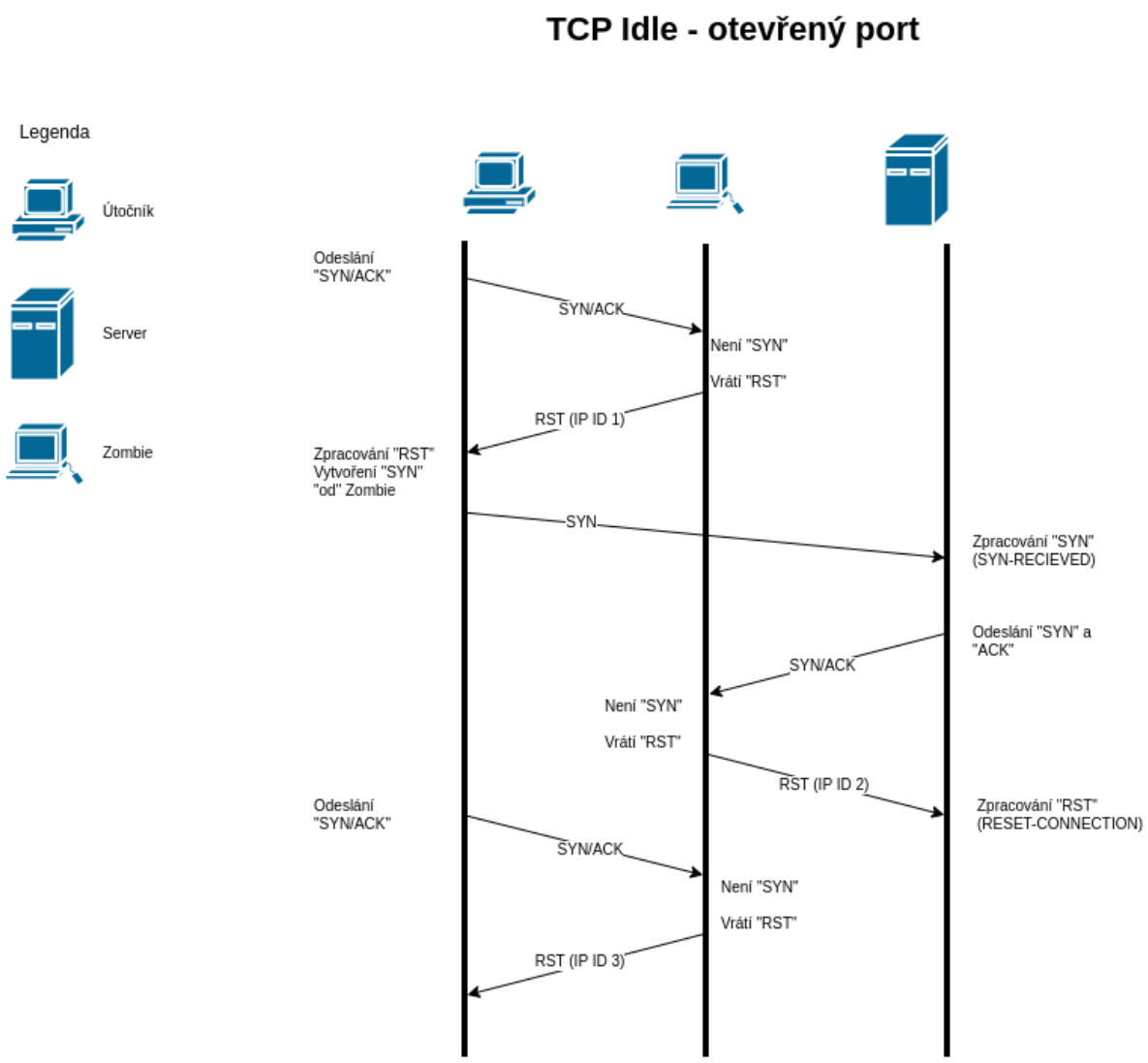
Zatímco tato možnost byla v roce 1996 docela užitečná, moderní systémy tuto chybu vykazují jen zřídka. Odesílají RST zpět pro všechny porty, takže každý port vypadá uzavřený. [35]

TCP Idle

TCP idle sken, také známý jako „zombie scan“, je skenovací technika používaná ke shromažďování informací o otevřených portech v cílovém systému bez přímého odesílání paketů z útočníkovy IP adresy. Tato metoda je navržena tak, aby byla nenápadná a vyhýbala se některým základním systémům IDS a firewallům tím, že využívá chování určitých typů síťových zařízení a jejich zpracování požadavků na připojení TCP. [36]

Tato technika se opírá o koncept nazvaný „IP ID sekvence predikce“. Každý IP paket odeslaný přes síť obsahuje ve své hlavičce pole identifikace (ID). Toto ID se obvykle používá k opětovnému sestavení fragmentovaných paketů na přijímací straně. Zajímavou vlastností je, že některé operační systémy nebo síťové prvky zvyšují toto IP ID pro každý odeslaný paket, bez ohledu na to, zda je paket součástí legitimního TCP spojení nebo ne. [6]

Na níže uvedeném diagramu je zobrazen proces kontroli otevřeného portu.



Obrázek 2.6: TCP Idle - otevřený port

Rozdělme tento proces to tří sub-procesů.

1. **Identifikace zombie**

   Útočník potřebuje v síti najít systém, který je zároveň online a má otevřený port. Tento systém se bude chovat jako ”zombie”, přes který budou poslány pakety. Systém zombie by měl mít předvídatelné chování sekvence IP ID, což znamená, že jeho IP ID se předvídatelně zvyšuje s každým odeslaným paketem, bez ohledu na cíl nebo účel paketu. Útočník nejrpve odešle SYN/ACK paket na zombie. Protože se nejedná o standartní způsob spojení (chybí SYN), zombie odpoví zprávou RST. [36]

2. **Podvodný SYN paket**

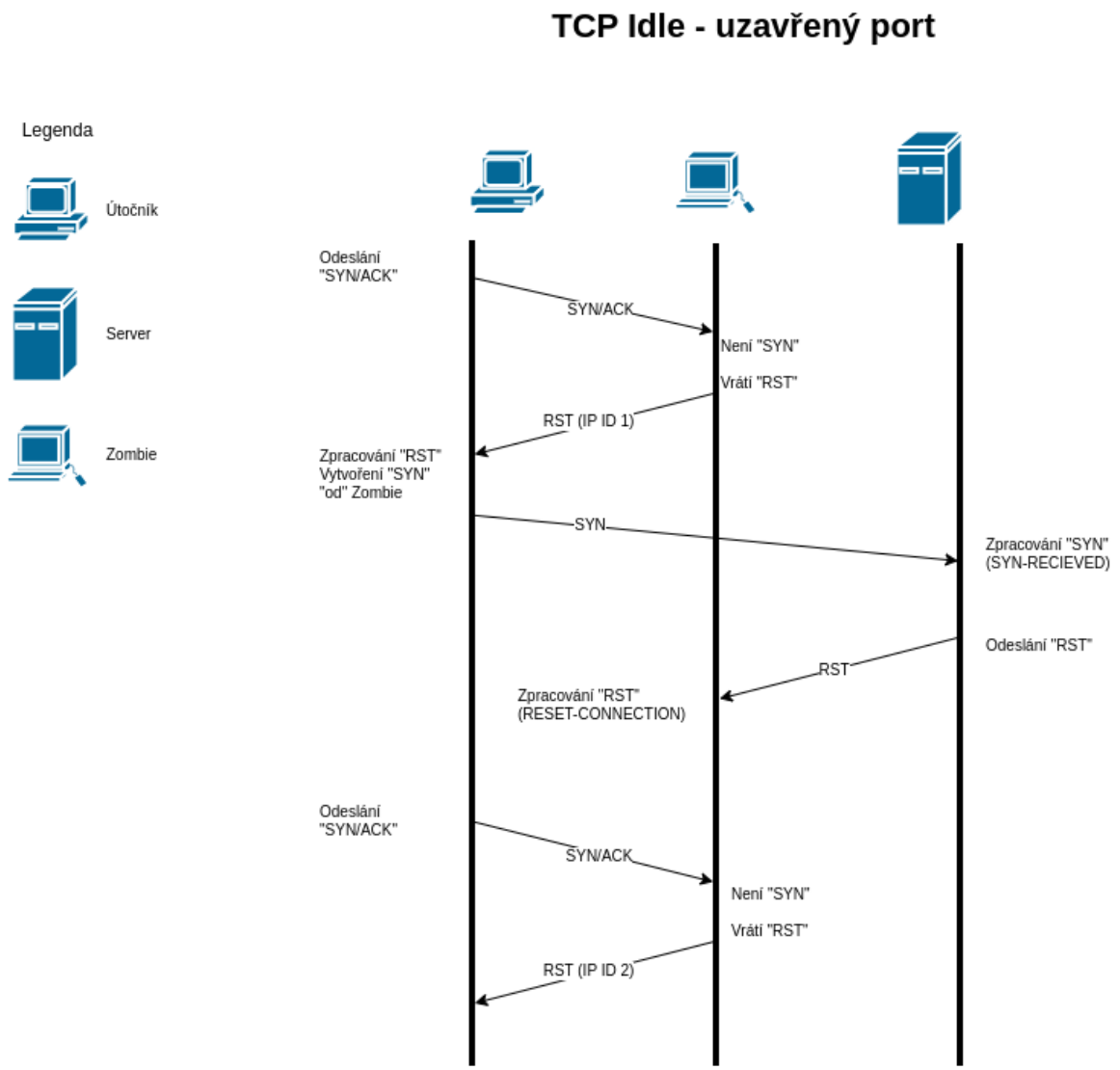
   Útočník připraví SYN paket, který bude odeslán na IP adresu serveru. Cílem tohoto paketu je prozkoumat konkrétní port. Zdrojová IP adresa v paketu je nastavena na IP adresu zombie.

Díky tomu server věří, že paket pochází od zombie. Pořadové číslo v paketu je nastaveno na hodnotu založenou na pozorované sekvenci IP ID zombie. To je pro sken zásadní. [36]

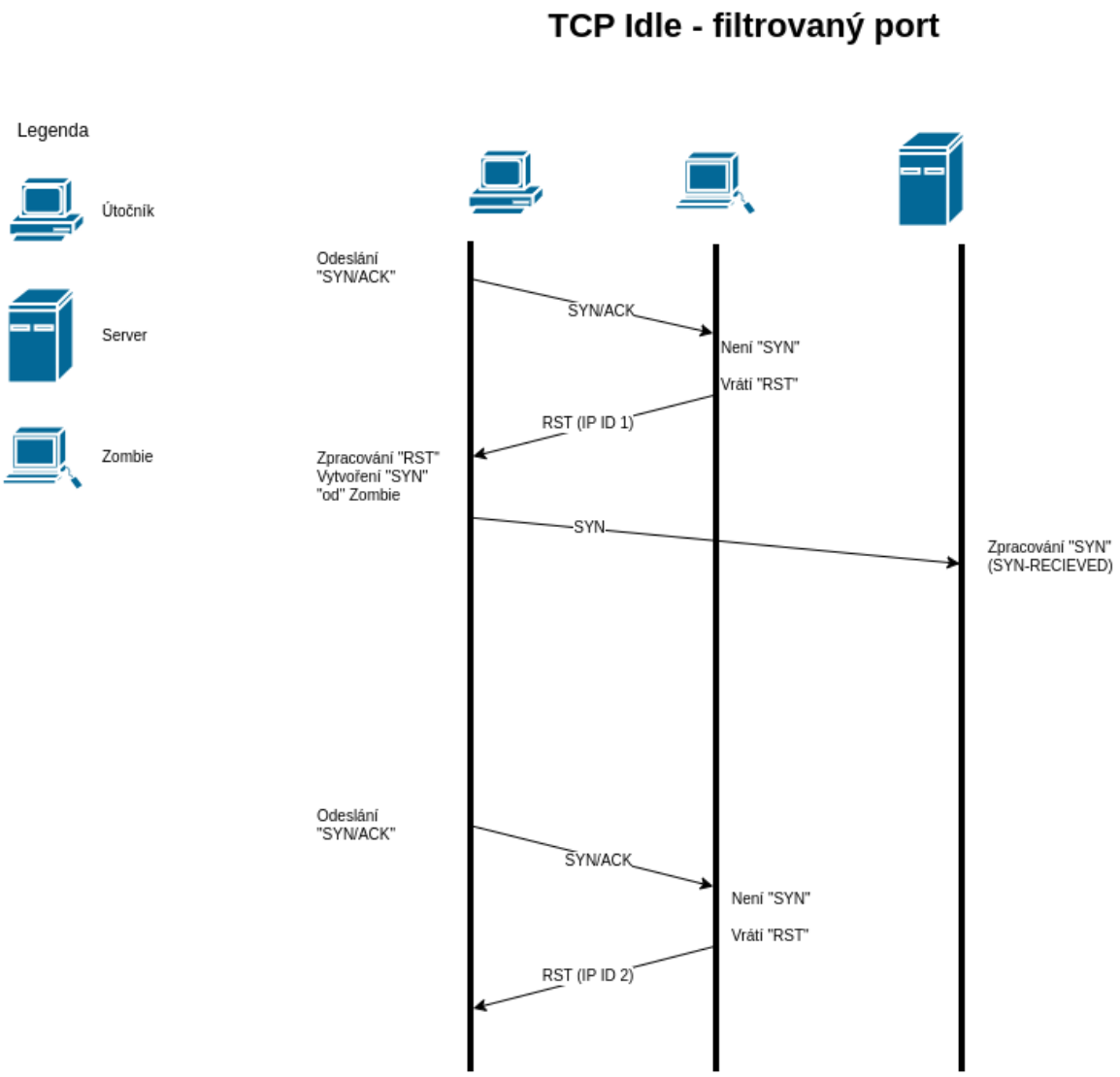
3. **Analyzování odpovědi**

   Útočník znovu odešle SYN/ACK na zombie a po vrácení RST zombie je IP ID zvýšeno o 2. Znamená to, že testovaný port je otevřený. [36]

Pro sken uzavřeného a filtrované portu je proces podobný.



Obrázek 2.7: TCP Idle - uzavřený port



Obrázek 2.8: TCP Idle - filtrovaný port

Všimněme si, že v obou těchto případech je IP ID zvýšené pouze o 1. Pro útočníka tedy není možné rozpoznat zdali se jedná o filtrovaný nebo uzavřený port. [36]

Jedinečnou výhodou idle skenování je, že jej lze použít k proražení určitých firewallů a směrovačů filtrující pakety. Filtrování zdrojové adresy IP je běžný (i když slabý) bezpečnostní mechanismus pro omezení strojů, které se mohou připojit k citlivému hostiteli nebo síti. Firemní databázový server může například povolit pouze připojení z veřejného webového serveru, který k němu přistupuje. Nebo může domácí uživatel povolit pouze připojení SSH (interaktivní přihlášení) ze svých pracovních strojů. Klíčovým faktorem je, že výsledky skenování obsahují seznam otevřených portů z pohledu hostitele zombie. Normální skenování proti výše uvedenému databázovému serveru nemusí ukázat žádné otevřené porty, ale provádění nečinného skenování při použití adresy IP webového serveru jako zombie by mohlo odhalit vztah důvěryhodnosti tím, že se porty služeb souvisejících s databází zobrazí jako otevřené. [36] [3]

Idle sken se může zdát jako dokonalá nenápadná technika skenování, nicméně systémy IDS budou obecně posílat výstrahy s tvrzením, že proti nim zombie stroj zahájil skenování. Další nevýhodou je dlouhá doba skenování, mnohem déle než většina ostatních typů skenování. A největší problém je, že útočník musí být schopen podvrhnout pakety, jako by přicházely od zombie, a nechat je dosáhnout cílového stroje. Mnoho poskytovatelů internetových služeb nyní implementuje filtrování odchozích kanálů, aby zabránili tomuto druhu falšování paketů. [6]

UDP sken

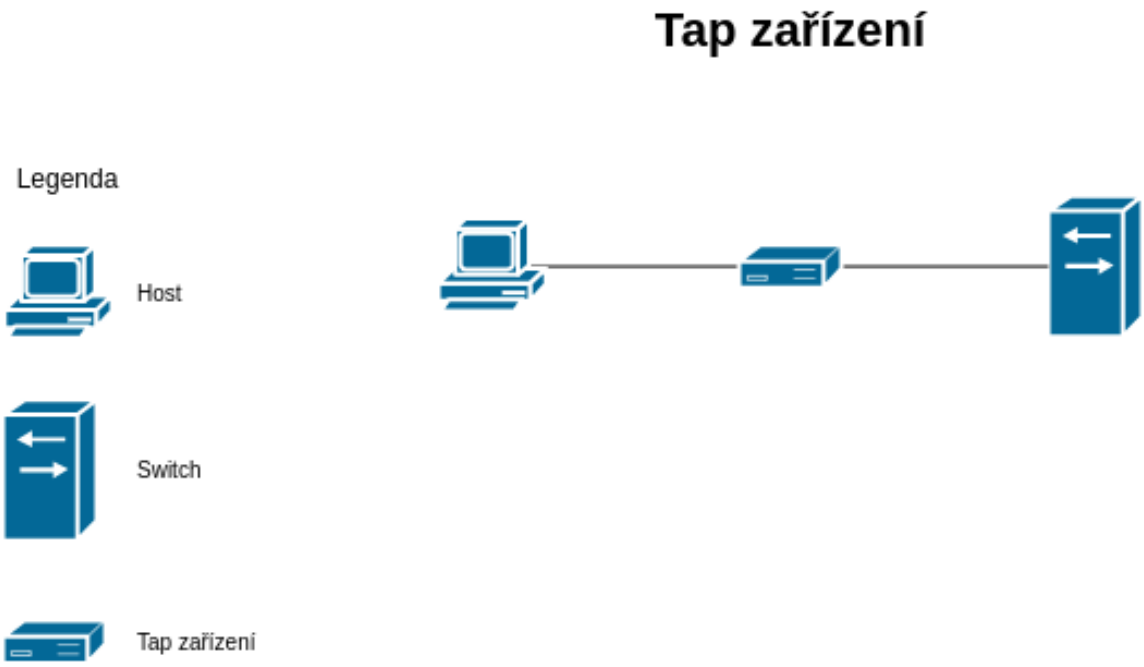
Zatímco většina populárních služeb na internetu běží přes protokol TCP, služby UDP jsou široce používány pro DNS, SNMP, DHCP apod. Sken UDP je založen na odeslání prázdné hlavičky na konkrétní port. Pokud cílový systém odpoví paketem UDP, znamená to, že port je otevřený a na tomto portu může být spuštěna služba. Pokud cílový systém odpoví chybovou zprávou ICMP, například „ICMP Port Unreachable“, znamená to, že port je uzavřen a nenaslouchá na něm žádná služba. V posledním případě se nemusí objevit žádná odpověď. Toto může znamenat, že je port otevřený nebo filtrovaný firewallem nebo zařízením pro zabezpečení sítě. [37]

Velkou výzvou pro skenování UDP je rychlé provedení. Otevřené a filtrované porty jen zřídka posílají odpověď, tudíž je nutné odeslat segment i vícekrát. Ještě větším problémem jsou často uzavřené porty. Obvykle pošlou zpět chybu nedostupnosti portu ICMP. Ale na rozdíl od paketů RST odesílaných uzavřenými porty TCP v reakci na skenování SYN nebo připojení, mnoho služeb standardně omezuje zprávy o nedostupnosti portu ICMP. [37]

Tap zařízení

Tap zařízení, často označované jako „síťová odbočka“, je hardwarové zařízení používané v sítích k zachycení a monitorování síťového provozu. Poskytuje způsob, jak zachytit a zkontrolovat datové pakety procházející segmentem sítě, aniž by došlo k narušení normálního provozu této sítě. Na rozdíl od tradičních metod monitorování sítě, které se spoléhají na softwarová řešení nebo použití portů síťových přepínačů v promiskuitním režimu, poskytují síťové odposlechy nerušivý a spolehlivý způsob, jak zachytit síťová data. [7]

Síťové odbočky fungují na jednoduchém principu vytváření kopii síťového provozu, aniž by zasahovaly do původního datového toku. Zařízení je obvykle vloženo mezi dvě síťová zařízení, jako je switch a host. Tap zařízení disponuje vstupním portem, výstupním portem a často také monitorovací portem, což umožňuje datový tok odeslílat přímo jeho vlastníkovi. [6]



Obrázek 2.9: Tap zařízení

Hlavní výhodou tohoto zařízení je jeho neviditelnost na síti. Tyto zařízení často nemají či nedostanou IP adresu a proto jsou pro vzdáleného správce neidentifikovatelné. Odbočky neodesílají testovací pakety ani aktivně nekontrolují síťové porty. Ve výsledku je méně pravděpodobné, že spustí bezpečnostní výstrahy nebo budou blokovány obranou sítě. To z nich dělá tajnější možnost monitorování a analýzy zabezpečení.

Zatímco síťové odbočky nabízejí neuvěřitelnou výhodu v tichém chodu, je důležité si uvědomit, že toto může být zneužito i útočníkem.

V této kapitole jsme se seznámili s technikami skenování portů, které byly vysvětleny i z hlediska útočníka. Útočník byl zde poprvé zmíněn z důvodu většího pochopení jednotlivých technik. Tato práce se nezabývá zneužitím těchto metodik, nicméně je potřeba si uvědomit, že veškeré výhody či nevýhody jednotlivých metodik mohou být zneužity právě daným aktérem.

3 Cesta vývoje moderních bezpečnostních prvků

3.1 Firewall

Firewall je hardwarové zařízení pro zabezpečení sítě nebo softwarová aplikace určená k monitorování, filtrování a řízení příchozího a odchozího síťového provozu na základě sady předdefinovaných bezpečnostních pravidel.

Tato zařízení se pravidelně používají k tomu, aby se nedůvěryhodní internetoví klienti nebo provoz nedostali do vnitřního systému a přesto umožnili vnířním klientům přístup k internetu. Jsou zásadní, protože poskytují samostatný vstupní a výstupní bod, kde lze vynutit bezpečnost a kontrolu nad síťovým tokem dat.

Dále poskytují možnost získávání důležitých, kritických dat pro analyzování síťového toku nebo po útoku do sítě.

Firewall řídí datový tok prostřednictvím sady kontrolních modelů, nazývaných jako politika síťového provozu. Jedná se o „bariéru“ mezi důvěryhodným systémem a jinými nedůvěryhodnými systémy např. internetem nebo webovým fórem zahrádkářů. [38]

Historie - důležité milníky ve vývoji FW

Termín firewall vznikl z hašení požáru a požární averze, kde je firewall vytvořená hranice, aby se zabránilo šíření plamene. [38]

Než se firewally koncem 80. let 20. století rozrostly, hlavním typem zabezpečení systému byly seznamy řízeného přístupu (ACL) implementované ve směrovačích. ACL zjistily, kterým IP umístěním byl povolen nebo odepřen přístup do systému. Rozvoj internetu a následná rozšířená dostupnost systémů znamenala, že tento druh opatření již nestačil na to, aby zabránil hackerské aktivitě, protože v hlavičkách packetů jsou obsažena pouze základní data. [39]

1988 – DEC Packet-Filter Firewall

V roce 1988 vyvinula společnost Digital Equipment Corporation (DEC) první generaci technologie firewallu s názvem Packet-Filter Firewall. Firewally s paketovým filtrem kontrolovaly pakety

přenášené mezi počítači v síti. Pokud paket neodpovídal pravidlům paketového filtru, byl buď zahozen, nebo odmítnut. Paket mohl projít, pokud odpovídal veškerým pravidlům filtrování. Filtrování bylo založeno na řadě mechanismů, jako jsou zdrojové a cílové síťové adresy, použité protokoly a čísla portů na obou koncích. [39]

Tento typ firewallu nehledí na stav připojení paketu. Neudržuje tedy stav. Proto se jim říká bezstavové firewally. Nazývají se také firewally síťové vrstvy. [39]

1989 – AT&T Bell Labs Stateful Firewall

V roce 1989 vyvinula společnost AT&T Bell Labs druhou generaci technologie firewallu s názvem Circuit Level Gateway, což byl první stavový Firewall. Stavový firewall uchovává informace o aktivních relacích a stavech připojení - tzn. zaznamenává všechna spojení, která jím procházejí. [39]

Tyto firewally používají informace o stavu připojení ke správě filtrování paketů. Pokud se přenášený paket neshoduje s aktivním spojením, je vyhodnocen podle sady pravidel filtrování vytvořené pro vytváření nových spojení. Pokud packet odpovídal pravidlům, mohl se přenést. Protože byl sledován stav připojení, byly tyto firewally nazývány stavovými firewally. [39]

Stavové firewally monitorují příchozí a odchozí pakety a také stavy připojení a poté tyto informace ukládají do tabulek dynamických stavů. Po navázání připojení mohou procházet pouze ty pakety, které jsou spojeny s připojeními uvedenými v tabulkách dynamických stavů. Relace uložené v této tabulce vyprší, pokud po definovanou dobu neprošel žádný provoz. Tím se zabrání zaplnění tabulky. [38]

Stavové firewally jsou druhým typem firewallů síťové vrstvy. Tyto firewally fungovaly také na transportní vrstvě. [38]

1991 – DEC Application Layer Firewall

V roce 1991 vydala společnost Digital Equipment Corporation (DEC) třetí generaci technologie firewall s názvem Application Layer Firewall se svým produktem nazvaným DEC SEAL (Secure External Access Link). Tyto firewally běží na aplikační vrstvě. Proto jsou schopny kontrolovat všechna data putující do a ze všech běžících softwarů. Hlavním cílem těchto firewallů je chránit počítače před malwarem. [39]

Jak název napovídá, Application Layer Firewall spravuje provoz aplikací, jako jsou webové prohlížeče a další, které se připojují k internetu a odesílají nebo přijímají data. Spravuje také provoz na FTP, Telnet a HTTP.

2004 – IDC zavádí termín Unified Threat Management (UTM)

V roce 2004 vytvořila International Data Corporation (IDC) termín v zabezpečení sítě nazvaný Unified Threat Management (UTM). UTM firewall je bezpečnostní systém pro ochranu sítě

v reálném čase. Jde o evoluci tradičního firewallu v komplexní řešení zabezpečení sítě. UTM využívá technologie jako Network Firewall, Web Filtering, Gateway Antivirus, IPS, Anti-Spam, VPN atd. k ochraně sítí před hrozbami. [38]

2009 – Gartner definuje Next-Generation FireWall (NGFW)

V roce 2009 Gartner představil koncept firewallu nové generace (NGFW). Firewall nové generace (NGFW) využívá koncepty tradičního firewallu spolu s novějšími technologiemi, jako je Network Firewall, IPS, Deep Packet Inspection (DPI), Sandboxing, Application Control, URL Filtering, Advanced Malware Protection, Profilování sítě, zásady identity, VPN a mnoho jiných. [39]

Tradiční firewally

Tradiční firewally, také přezdívané legacy firewally řídí síťový provoz mezi důvěryhodnou sítí a nedůvěryhodnou nebo veřejnou sítí. Tyto firewally jsou často v praxi používáné, protože se relativně snadno obsluhují a udržují. Jedná se převážně o levnější varianty a disponují dobrou propustností dat, proto převládají na trhu více než dvě desetiletí. [38]

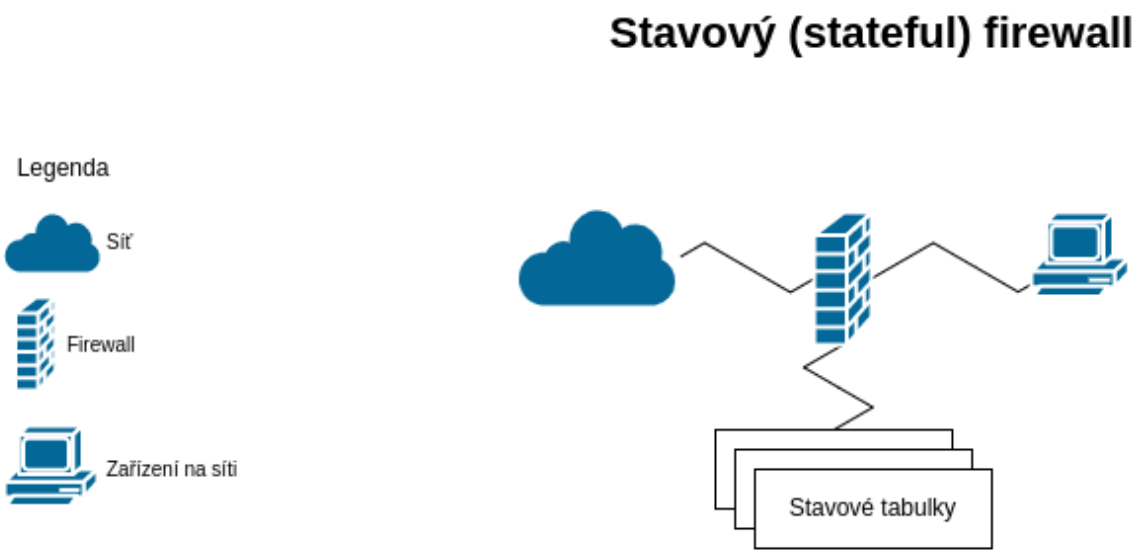
Tradiční firewally obvykle používají bezstavový (stateless) nebo stavový (stateful) přístup firewallu.

Stavové (stateful) firewally

O stavovém firewallu jsme se již zmínili v 3.1. Stavový firewall může také sledovat, jak se data chovají a katalogizovat vzorce chování. Pokud kontrola datových paketů odhalí podezřelé chování, i když tento druh chování nebyl ručně zadán správcem, firewall jej dokáže rozpoznat a hrozbu řešit. Stavový firewall lze použít na okraji sítě nebo uvnitř, jako je tomu v případě interního segmentačního firewallu (ISFW), který chrání konkrétní segmenty sítě v případě, že se dovnitř dostane škodlivý kód. [40]

Stavové firewally jsou vysoce kvalifikované v odhalování neoprávněných pokusů nebo falešných zpráv. Tyto brány firewall nepotřebují pro správnou komunikaci mnoho otevřených portů.

Nevýhodou tohoto přístupu jsou chyby zabezpečení, které mohou hackerovi umožnit kompromitovat a převzít kontrolu nad bránou firewall, která není aktualizována nejnovějším verzem softwaru. Dále útoky typu Man-in-the-middle mohou představovat větší zranitelnost oproti bezstastavovými firewally.



Obrázek 3.1: Stavový firewall

Stavové brány firewall uplatňují svůj přístup v závislosti na stavu připojení. V TCP řídí stav připojení čtyři části - SYN, ACK, FIN a RST. Poté, co brána firewall potvrdí určitý typ provozu, připojí se k tabulce stavů a bude se volně pohybovat. Provoz, který neprovede handshake, již nebude existovat. [18] [40]

Stavové firewally mohou obsahovat funkcionalitu na úrovni okruhu (circuit-level). Fungují stejným způsobem jako bezstavové firewally, ale mají výhodu v tom že mohou jít až do vrstvy 5 ISO/OSI, ve srovnání s bezstavovými firewally, které fungují pouze na vrstvě 3 a 4 ISO/OSI. Výsledkem je větší flexibilita firewallu a přísnější bezpečnostní pravidla. [41]

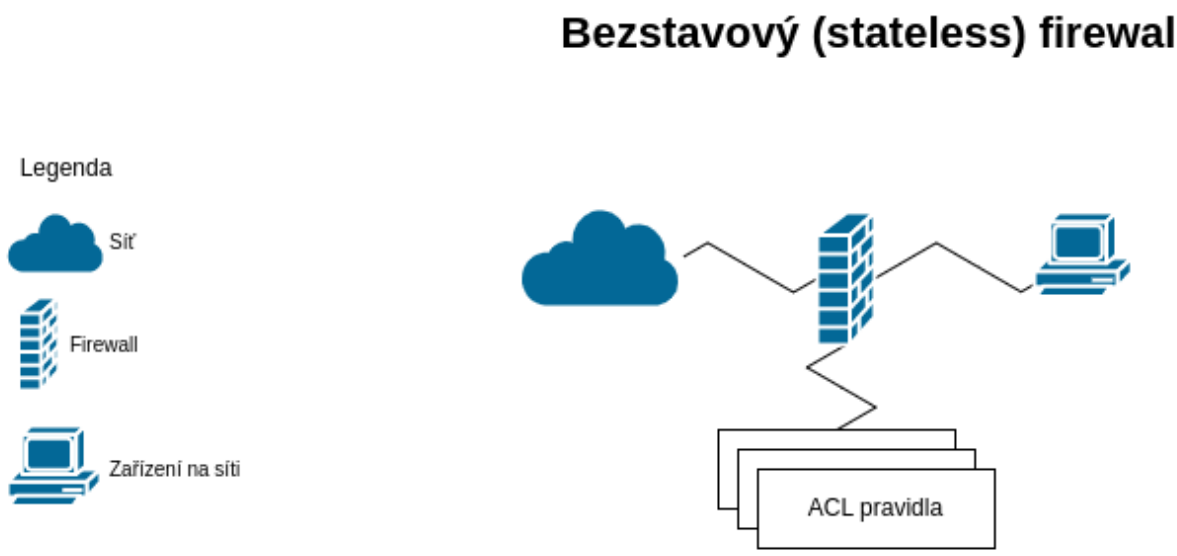
Bezstavové (stateless) firewally

Bezstavové brány firewall využívají zdroj, cíl a další parametry datového paketu, aby zjistily, zda data představují hrozbu. Tyto parametry musí zadat buď správce, nebo výrobce pomocí předem nastavených pravidel. [42]

Pokud datový paket překročí parametry toho, co je považováno za přijatelné, bezstavový firewall protokol identifikuje hrozbu a poté omezí nebo zablokuje data, která v něm jsou uložena.

Bezstavový firewall funguje na vrstvě 3 a 4 ISO/OSI. Bezstavové firewally jsou méně spolehlivé než stavové firewally při kontrole jednotlivých datových paketů. Přesto v případě obrany silnějšího útoku můžou být lepší, protože zablokují útok pohledem na vzor. Bezstavové firewally neumí rozlišovat mezi různými síťovými provozy, tedy HTTP, HTTPS, FTP, SSH atd. [41]

Nevýhodou tohoto přístupu je, že se nezkoumá celý paket, ale místo toho rozhodne, zda paket vyhovuje stávajícím bezpečnostním pravidlům. [42]



Obrázek 3.2: Bezstavový firewall

Osobní (personal) firewally

Osobní firewally jsou navrženy a implementovány tak, aby chránily jedno zařízení před neoprávněným přístupem. Osobní firewally mohou také integrovat další funkce, jako je monitorování síťového provozu, analýza chování a IDS. [43]

Zatímco osobní firewally mají obrovský smysl na trhu SOHO (small office/homeoffice) a domácích uživatelů, protože poskytují ochranu koncového uživatele i kontrolu nad politikou, v podnicích jsou problémy složitější. Největším problémem podnikových uživatelů s ohledem na osobní firewally je schopnost poskytnout pro firewall centralizovaný mechanismus kontroly bezpečnostních politik. Potřeba centralizovat kontrolu politik je zásadní pro používání osobních firewallů v podnikovém prostředí, aby se minimalizovala administrativní zátěž. [43]

Personální firewally, což je většina firewallů na aplikační úrovni (softwarové), jsou dnes nabízeny jako komerční a open-source řadou společností a poskytovatelů, např. Symantec nebo Malwarebytes. Přestože osobní firewally jsou docela schopné analyzovat a předcházet hrozbám z vnější sítě, podniky potřebují centralizovat a prosazovat své vlastní zásady, a proto jsou hardwarové síťové firewally v takových prostředích obvykle povinné. [43] [42]

Podniky nasazují kromě osobních firewallů také síťové firewally, aby prosadily a centralizovaly své vlastní zásady a řídily tok provozu mezi vnější a vnitřní sítí.

Síťové brány firewall

Síťové firewally jsou umístěny v přední linii sítě a slouží jako komunikační spojení mezi důvěryhodnou interní sítí a nedůvěryhodnou sítí. [42]

Síťové firewally jsou běžně klasifikovány do tří typů podle způsobu ochrany sítě:

- **Filtrování paketů**

  Firewally s touto funkcí provádějí pouze velmi základní operace, jako je zkoumání hlavičky paketu, ověřování IP adresy, portu nebo obou dvou a udělování anebo zakazování přístupu bez provedení jakýchkoli změn. Díky této jednoduchosti ovládání mají výhodu jak rychlosti, tak účinnosti. Filtrované pakety mohou být příchozí, odchozí nebo obojí, v závislosti na typu směrovače. Další výhodou je, že svou práci vykonávají tiše, nezávisle na znalostech nebo pomoci uživatele, tj. mají dobrou transparentnost. [6] [40]

  Pakety lze filtrovat na základě některých nebo všech následujících kritérií: zdrojová IP adresa, cílová IP adresa, zdrojový port TCP/UDP a cílový port TCP/UDP. Firewall tohoto typu může blokovat připojení do a z konkrétních zařízení, sítí anebo portů. Jsou relativně levné, protože používají software, který se již nachází ve směrovači, a poskytují dobrou úroveň bezpečnosti. [6]

- **Circuit proxy**

  Hlavní rozdíl mezi circuit proxy a filtrováním paketů je ten, že adresátovi musí všichni komunikátoři adresovat své pakety. Za předpokladu, že byl povolen přístup, circuit proxy nahradí původní adresu (svou vlastní) adresou zamýšleného cíle. Tento přístup má nevýhodu v tom, že si nárokuje zdroje potřebné k provedení změn v hlavičce, a výhodu skrývá IP adresu cílového systému. [6] [40]

- **Aplikační proxy**

  Tento přístup je složitější oproti filtrování paketů nebo circuit proxy. Aplikační proxy rozumí aplikačnímu protokolu a datům, a zachycuje veškeré informace určené pro danou aplikaci. Na základě množství informací dostupných pro rozhodování může aplikační proxy autentizovat uživatele a posoudit, zda některá data mohou představovat hrozbu. Cenou za tuto komplexnější funkci je to, že uživatelé nebo klienti musí být na ně často přenastaveni, což je někdy komplikovaný proces s následnou ztrátou transparentnosti. Aplikační proxy se označují jako proxy služby a hostitelské stroje, které je provozují, jako aplikační brány. [6] [40]

Limitace firewallu

Profesionálové v oblasti informační bezpečnosti se často ocitají v rozporu s mylnými představami a populárními názory vytvořenými z neúplných dat. Některé z těchto názorů pramení spíše z naděje než ze skutečnosti, jako je například myšlenka, že vnitřní zabezpečení sítě lze vyřešit jednoduše nasazením firewallu. I když je pravda, že firewally hrají důležitou a ústřední roli při udržování bezpečnosti sítě a každá organizace, která je ignoruje, činí tak na vlastní nebezpečí. Vědět, co firewally neumí, je stejně důležité jako vědět, co umí. Následují omezení, kterých by si člověk měl být vědom. [44]

- Firewall je ze své podstaty obranou prostředí a není zaměřen na boj s útočníkem uvnitř sítě, a proto není užitečným protiopatřením proti uživateli, který zneužívá autorizovaný přístup k doméně. [42] [44]
- Firewall není skutečnou obranou proti problémům se škodlivým kódem, jako jsou viry, Trojské koně a jiné, i když některé jsou schopni skenovat kód a hledat v něm příznačné znaky.
- Konfigurace pravidel pro filtrování paketů bývá komplikovaným procesem, v jehož průběhu může snadno dojít k chybám vedoucím k bezpečnostním trhlinám. Testování nakonfigurovaných pravidel bývá zdlouhavý a obtížný proces kvůli nedostatkům současných testovacích nástrojů.

Zranitelnosti firewallu

Firewally vytvářejí digitální bariéru mezi sítí a potenciálními hrozbami. Přetrvává však kritická zranitelnost, kterou žádná sofistikovaná technologie nedokáže plně vyřešit - lidský prvek. Uživatelé, i přes ochranné vrstvy firewallů, zůstávají významnou bezpečnostní hrozbou a jejich činy často nevědomky otevírají dveře škodlivým entitám.

Jedním z nejúčinnějších nástrojů v arzenálu útočníka je sociální inženýrství. Tato technika se opírá o manipulaci lidské psychologie za účelem získání neoprávněného přístupu k informacím nebo systémům. Využitím důvěry a psychologických triků mohou úročníci oklamat uživatele, aby odhalili citlivá data nebo poskytli přístup k chráněným souborům. Ať už prostřednictvím phishingových e-mailů, podvodných telefonních hovorů nebo dokonce osobních interakcí, sociální inženýrství využívá lidské slabosti a dělá z uživatelů nevědomé spoluviníky při prolomení obrany firewallu. [2]

Lidská slabost hraje klíčovou roli v úspěchu útoků sociálního inženýrství. Strach, zvědavost a touha být nápomocný jsou emoce, které lze využít. Například zdánlivě neškodný e-mail požadující naléhavou akci nebo známý hlas na druhém konci telefonní linky mohou vyvolat reakce, které ohrožují bezpečnost. Školení uživatelů k rozpoznání těchto taktik a podpora kultury uvědomělé v oblasti kybernetické bezpečnosti v organizacích je zásadní pro zmírnění této lidské zranitelnosti.

Síťová analýza se ukazuje jako mocný nástroj pro pochopení a řešení bezpečnostní hrozby, kterou představují uživatelé. Pečlivým zkoumáním vzorců chování uživatelů mohou správci sítě identifikovat anomálie, které mohou naznačovat potenciální narušení bezpečnosti. Podezřelé aktivity, nepravidelné přihlašovací vzory nebo neočekávané přenosy dat mohou být varovným signálem, který podnítí další vyšetřování. Síťová analýza nejen pomáhá při odhalování potenciálních hrozeb, ale také umožňuje implementaci proaktivních opatření k posílení firewallu proti zranitelnostem zaměřeným na člověka.

Mezi běžné síťové útoky odepírající přístuponost služeb (DOS či DDOS útoky) patří:

Poštovní bomby

Poštovní bomba je útok navržený k zahlcení doručené pošty nebo zablokování e-mailového serveru odesláním velkého počtu e-mailů konkrétní osobě nebo systému. Cílem je zaplnit místo na disku příjemce na serveru nebo přetížit server, aby přestal fungovat.

Útoky poštovních bomb jsou obvykle iniciovány záměrně botnetem, útočníkem nebo skupinou útočníků. Škody způsobené poštovní bombou se mohou pohybovat od drobných nepříjemností až po úplný výpadek služeb. Tyto útoky mohou trvat několik hodin, pokud není vynaloženo žádné úsilí k filtrování, zmírňování nebo blokování útočícího provozu.

Existuje mnoho typů poštovních bomb, níže jsou uvedené nejčastější příklady:

- Přílohové (Attachement-based). K tomuto útoku dochází, když je odesláno více e-mailů s velkými přílohami. Jsou navrženy tak, aby rychle přetížily úložný prostor serveru a způsobily, že nebude reagovat. [45]
- Propojení seznamů (List linking). Jedná se o útok, který používají útočníci k přihlašování cílených e-mailů do více e-mailových předplatitelských služeb. Cílem je nepřímo zahltit e-mailové adresy předplaceným obsahem. To je možné, protože mnoho předplatitelských služeb nevyžaduje žádné ověření. Je obtížné bránit se útokům na propojení seznamů, protože provoz pochází z legitimních zdrojů. [45]
- Hromadná pošta (Mass mailing). Hromadná pošta je druh poštovní bomby, která není vždy úmyslná. Například místo kliknutí na jednu e-mailovou adresu může uživatel omylem vybrat všechny a omylem odeslat e-mail na stovky nebo tisíce cílených e-mailových adres. Záměrné hromadné poštovní bomby jsou často iniciovány pomocí botnetů nebo škodlivých skriptů. Útočnící mohou například automatizovat vyplňování online formulářů s cílovou e-mailovou adresou jako žádající/zpáteční adresou. [45]
- Odpovědět všem (Reply all). Když uživatel odpoví kliknutím na „Odpovědět všem“ na rozsáhlý seznam e-mailových adres namísto pouze původního odesílatele, doručené pošty jsou zaplaveny e-maily. Toto může mít za následek řetězový efekt v případě, že někteří uživatelé používají automatizované zprávy o nepřítomnosti nebo dovolené. [45]

Ping flood

Jedná se o útok, při kterém útočník znepřístupní zařízení tím, že jej zahltí požadavky na odezvu ICMP, známé také jako pingy.

Normálně se požadavky ping používají k testování konektivity dvou zařízení měřením doby zpáteční cesty od okamžiku odeslání požadavku ICMP do okamžiku přijetí odpovědi ICMP. Během útoku se však používají k přetížení cílové sítě datovými pakety.

Útok zahrnuje zaplavení sítě cílového zařízení pakety požadavků s vědomím, že síť odpoví stejným počtem paketů odpovědí.

Obrana proti tomuto typu útoku je poměrně jednoduchá. Je nutné nakonfigurovat firewall tak, aby nepovoloval pingy z venkovní sítě (avšak ICMP požadavky z vnitřní sítě jsou ponechány povolené). Plošné blokování požadavků ping však může mít nezamýšlené důsledky, včetně neschopnosti diagnostikovat problémy se serverem. [46]

Softwarové chyby

Zde softwarovou chybou může být myšlena chyba přímo v softwaru výrobce firewall, hostované aplikace společnosti, která je dostupná i mimo vnitřní síť anebo lokálně nainstalovaná aplikace na zařízení. [6]

Existuje několik způsobů, jak zjistit informace o používaných službách či aplikací cílového zařízení např. sociálním inženýřstvím, prostřednictvím canary tokenů ale i náhodným zkoušením.

Na bežném webu je řada webových portálů, které tyto zranitelnosti zveřejnují. Důležitým pojmem u zranitelnosti je CVE. CVE, zkratka pro Common Vulnerabilities and Exposures, je seznam veřejně odhalených chyb v počítačové bezpečnosti. Když někdo odkazuje na CVE, myslí tím bezpečnostní chybu, které bylo přiděleno ID číslo CVE. [47]

Bezpečnostní upozornění vydaná prodejci a výzkumníky téměř vždy zmiňují alespoň jedno CVE ID. CVE pomáhají IT profesionálům koordinovat jejich úsilí při stanovení priorit a řešení těchto zranitelností, aby byly počítačové systémy bezpečnější. [47]

Jakmile dostane zranitelnost svoje CVE, je potvrzena MITRE corporation, která zodpovídá za správu CVE a kontrolu zranitelností, je zranitelnost potvrzena a výrobci softwaru by měli chybu co nejrychleji odstranit anebo nabídnout konsumerům alternativu řešení. [47]

Dle průzkumů rizik [48] [49] a [50] více než polovina všech narušení jsou dnes způsobena některým legitimním uživatelem, který je již za firewallem.

Tradiční předpoklad, že všichni uvnitř firewallu jsou přátelští a všichni mimo něj potenciálně nepřátelští, se nyní stává poněkud zastaralým. Internetová konektivita se rozšířila, extranety mohou umožnit přístup zvenčí do oblastí chráněných firewally a některé stroje vyžadují větší přístup ven než jiné, což často zahrnuje změnu interní IP adresy. Další hrozbou je použití end-to-end šifrování, protože firewall není schopen procházet šifrováním.

Bez ohledu na kryptografii je používání firewallů v řadě organizací hluboce zakořeněno a je nedílnou součástí jejich nastavení zabezpečení a bude tomu tak ještě několik let.

Dalším faktorem je neustálý vývoj nových funkcí a služeb, které jsou v současné době neustále přidávány do firewallů. Ty snižují řadu výše uvedených omezení a zvyšují flexibilitu firewallu a zároveň mu umožňují zachovat si svou původní funkci bez narušení. Zde je několik příkladů, které ilustrují tento bod, jsou:

- Návrh distribuovaného firewallu [51] využívajícího IPSEC (IP Security) a nástroje pro správu systému, který zachovává centrální řízení přístupové politiky a zároveň snižuje nebo odstraňuje jakoukoli závislost na topologii.

- Phoenix's Adaptive Firewall Technology [52] poskytuje samoadaptující kontrolu síťového přístupu, čímž vytváří efektivní síťovou bezpečnostní politiku zkoumáním každého paketu a přizpůsobováním pravidel „on-the-fly" na základě informací v paketu procházejícím síťovým rozhraním.

3.2 Detekce a prevence narušení síťového provozu

V této kapitole se v krátkosti seznámíme se systémy pro detekci a prevenci narušení síťového provozu.

IDS

Systémy pro detekci narušení síťového provozu (Intrusion Detection Systems, IDS) byly vytvořeny, aby reagovaly na rostoucí potřebu lepší kybernetické bezpečnosti a pomohly chránit počítačové sítě a systémy před různými hrozbami a zranitelnostmi. S tím, jak se počítačové sítě a systémy staly více rozšířenými a propojenými, rostl počet a sofistikovanost kybernetických hrozeb. Neoprávněný přístup, narušení dat, malware a další škodlivé aktivity představovaly značná rizika pro jednotlivce i organizace. Tento systém byl vyvinut za účelem proaktivně detekovat a reagovat na tyto hrozby. [53]

Oproti tradičním firewallům jsou tyto systémy navrženy tak, aby filtrovaly a řídily příchozí a odchozí síťový provoz na základě předem definovaných pravidel. Samotné firewally však nemohou detekovat nebo zabránit všem typům útoků nebo neoprávněnému přístupu. IDS doplňují firewally aktivním monitorováním síťového provozu a systémových aktivit, zda nevykazují známky narušení, a to i po počáteční obraně firewallu.

Tyto systémy dokážeme rozdělit na síťově zaměřené IDS (Network-based IDS, NIDS) a hostově zaměřené IDS (Host-based IDS, HIDS). NIDS monitoruje síťový provoz a hledá podezřelé vzory nebo značky v datech. Funguje na úrovni sítě a je často strategicky umístěn v rámci síťové infrastruktury. HIDS se zaměřuje na jednotlivé hostitelské stroje nebo servery, sleduje jejich systémové protokoly, soubory a aktivity. Je více přizpůsoben konkrétním systémům a dokáže detekovat hrozby, které se vyskytují interně na hostiteli. [53]

Metody detekce IDS

- **Detekce na základě značek**

  Tato technika se opírá o předdefinované vzory nebo značky, které jsou charakteristické pro známý malware, exploity nebo pokusy o narušení. Když se síťový provoz nebo systémové aktivity shodují s těmito předdefinovanými značkami, IDS vyšle výstrahu indikující potenciální bezpečnostní hrozbu. [54]

Detekce se opírá o databázi známých značek útoků. Tyto značky jsou v podstatě vzory nebo sekvence dat, které jsou jedinečné pro konkrétní typy útoků nebo škodlivých aktivit. Pro bezpečný chod je nutné neustále aktualizovat tuto databázi, aby obsahovala nové značky útoků. Tato detekce není účinná proti novým nebo dříve neviditelným hrozbám, jako jsou zranitelnosti zero-day nebo útoky, které byly pečlivě vytvořeny tak, aby se vyhnuly detekci značek. [54]

Jedním z problémů s detekcí založenou na značkách jsou falešné poplachy. K falešným poplachům dochází v případě, kdy legitimní provoz nebo aktivity jsou nesprávně identifikovány jako hrozby, protože se shodují se značkou. To může vést ke zbytečným upozorněním a administrativní režii. [54]

- **Detekce na základě anomálií**

  Detekce anomálií stanoví základní linii normálního chování sítě nebo systému a poté označí jakékoli odchylky od této základní linie jako potenciální narušení. To je užitečné pro detekci dříve neznámých nebo zero-day útoků. Pokud však základní linie není přesně definována, může to vést k falešně pozitivním vyhodnocením. [54] [6]

  Jakmile je stanovena základní linie, monitorovací systém nepřetržitě sleduje a shromažďuje data ze sítě, systému nebo aplikace v reálném čase. Tato data zahrnují různé parametry, jako je počet pokusů o přihlášení, přístupy k souborům a jiné. [54]

  Nalezené odchylky mohou být náhlé špičky nebo poklesy v síťovém provozu, neobvyklé značky spotřeby zdrojů, neočekávané přístupy nebo jakékoli chování, které se výrazně liší od toho, co je považováno za normální. Detekce je založena na statistické analýze nebo modelech strojového učení, které porovnávají aktuální data s historickými výchozími daty. [54] [23]

  Tato detekce je skvělým doplňkem detekcí na základě značek, tím že vyniká v identifikaci nových a sofistikovaných útoků a hrozeb zevnitř.

- **Hybridní detekce**

  Hybridní detekce je přístup, který kombinuje silné stránky detekce na základě značek a detekce na základě anomálií a poskytuje komplexnější a efektivnější kybernetickou bezpečnost. Tento přístup má za cíl zlepšit přesnost detekce hrozeb a zároveň minimalizovat falešně pozitivní a falešně negativní vyhodnocení. [54] [7]

  Detekce založená na značkách je náchylná k falešným poplachům, protože může spustit upozornění na jakoukoli aktivitu, která je shodná se značkou, i když se jedná o běžný provoz. Detekce založená na anomáliích pomáhá snižovat počet falešných poplachů tím, že bere v úvahu anomálie kontextu a chování. Tímto je také zvýšena přesnost, pokud obě detekce upozorní na stejnou aktivitu, je pravděpodobnější, že se jedná o skutečnou hrozbu. [54]

  Hybridní detekční systémy často upřednostňují výstrahy na základě závažnosti a úrovně spolehlivosti detekce. Výstrahy s vysokou spolehlivostí, které jsou označeny jak analýzou

podpisu, tak analýzou anomálií, získají okamžitou pozornost, zatímco výstrahy s nižší spolehlivostí mohou projít dalším šetřením. [54]

Správa a konfigurace systému hybridní detekce vyžaduje pečlivé plánování a odborné znalosti, aby se dosáhlo správné rovnováhy mezi detekcí na základě podpisů a anomálií. Jemné doladění je nezbytné pro zajištění optimálního výkonu a minimalizaci falešných výstrah.

IPS

Systém pro prevenci narušení síťového provozu je forma zabezpečení sítě, která slouží k prevenci identifikovaných hrozeb. Systémy prevence narušení nepřetržitě monitorují síť, hledají možné škodlivé incidenty a zachycují o nich informace. IPS hlásí tyto události správcům systému a provádí preventivní opatření, jako je uzavření přístupových bodů a konfigurace firewallů, aby se zabránilo budoucím útokům. [55]

Typy prevence

IPS využívají různé techniky prevence k identifikaci a zmírnění síťových hrozeb a útoků.

- **Prevence založená na značce**

  Toto je nejtradičnější a nejrozšířenější metoda. Spoléhá na databázi známých značek nebo vzorů útoků. Když se příchozí síťový provoz shoduje se známou signaturou, IPS podnikne kroky k zablokování nebo zmírnění hrozby. Prevence založená na značkách je účinná proti známému malwaru a dobře zdokumentovaným technikám útoků, ale může se potýkat s útoky zero-day, dříve neznámými zranitelnostmi. [56] [57]

- **Prevence založená na anomáliích**

  Prevence založená na anomáliích se zaměřuje na identifikaci odchylek od normálního chování sítě. Namísto spoléhání se na předdefinované signatury stanoví základní linii „normální“ síťové aktivity a upozorní nebo provede akci, když zjistí významné odchylky. Detekce anomálií může být účinná při odhalování nových hrozeb, ale může také generovat falešné poplachy, pokud není správně vyladěna. [56] [57]

V oblasti síťové bezpečnosti a analýzy jsou IDS a IPS dvě odlišné, ale úzce související systémy, používané k ochraně sítí a systémů před kybernetickými hrozbami.

IDS je primárně navržen tak, aby detekoval a upozorňoval na podezřelé nebo potenciálně škodlivé aktivity vyskytující se v síti nebo na hostiteli. Monitoruje síťový provoz nebo systémové protokoly, aby identifikoval potenciální bezpečnostní incidenty, aniž by podnikal přímou akci k jejich blokování nebo prevenci. IPS na druhou stranu hrozby nejen detekuje, ale také jim aktivně předchází. Dokáže automaticky zablokovat nebo zmírnit bezpečnostní incidenty v reálném čase, jako je blokování škodlivého síťového provozu nebo ukončení podezřelých připojení. [2] [54]

IDS je pasivní povahy. Pozoruje a podává zprávy o bezpečnostních událostech, ale nepodniká žádné přímé kroky k jejich zastavení. Jeho primární úlohou je poskytovat výstrahy bezpečnostním správcům, kteří se pak rozhodnou, jak na hrozby reagovat. IPS je aktivní a může provádět automatické akce k prevenci nebo blokování bezpečnostních hrozeb. Může například zahazovat pakety, uzavírat síťová připojení nebo překonfigurovat pravidla firewall v reálném čase, aby zastavil probíhající útok. [54]

IDS může generovat falešné poplachy, protože se zaměřuje na detekci anomálií nebo vzorů, které odpovídají známým značkám útoku. IPS může také vytvářet falešné poplachy, ale riziko je vyšší, protože blokuje provoz pomocí automatických akcí. To může potenciálně narušit legitimní síťový provoz, pokud není správně nakonfigurován. [54]

V praxi se často používají kombinaci řešení IDS a IPS k dosažení bezpečnostního přístupu. IDS pomáhá při odhalování a analýze hrozeb, zatímco IPS přidává aktivní vrstvu obrany, která zabraňuje a zmírňuje útoky v reálném čase. [6]

IDS a IPS jsou nezbytné pro analýzu sítě, protože poskytují proaktivní detekci hrozeb, pomáhají při reakci na incidenty a pomáhají udržovat silnou pozici zabezpečení tváří v tvář vyvíjejícím se hrozbám a zranitelnostem. Hrají klíčovou roli v moderních strategiích zabezpečení sítí.

3.3 Next-Generation Firewall

Důvod vzniku firewallu nové generace

S rozvojem internetu se rozvíjí i množina bezpečnostích hrozeb a hackerských útoků. Do aplikací se začaly implementovat bezpečností opatření v obdobě např. šifrování k zajištění soukromí a ochrany dat. Toto šifrování může být odesíláno i přes nedůvěryhodné sítě za předpokladu, že šifrovací klíč byl vykomunikován bezpečnou komunikací.

Organizace také současně využívaly různé druhy bezpečnostních řešení, např. IPS, proxy a další komplexní zařízení spolu s tradičními firewally, aby zlepšily své bezpečnostní politiky. Takovéto sítě obsahující komplexně vzájemně propojené bezpečnostní zařízení jsou problematické na správu a jejich údržba je z dlouhodobého hlediska nákladná, nemluvě o problémech týkajících se škálovatelnosti. [58]

V boji proti rostoucímu množství hrozeb jsou v organizacích nasazovány brány firewall nové generace (Next-Generation Firewall, dále jen NGFW), které bezpečnostním týmům nebo administrátorům sítě poskytují nové možnosti zlepšovat a prosazovat bezpečnostní zásady společnosti.

Definice NGFW

NGFW je pokročilé zařízení nebo software pro zabezpečení sítě, které kombinuje tradiční funkce brány firewall s dalšími bezpečnostními funkcemi a funkcemi určenými k ochraně moderních sítí

a aplikací. NGFW jsou navrženy tak, aby mimo jiné poskytovaly vylepšenou detekci a prevenci hrozeb, kontrolu aplikací a hloubkovou kontrolu paketů. [59]

NGFW umí vše, co běžné firewally. To ještě navíc zahrnuje:

- **Rozsah ochrany** NGFW poskytují pokročilé bezpečnostní funkce, jako je hloubková kontrola paketů (DPI), povědomí o aplikacích a sledování identity uživatelů. Mohou identifikovat a řídit aplikace a služby na aplikační vrstvě a nabídnout podrobnější kontrolu nad síťovým provozem. [58]
- **Povědomí o aplikaci** NGFW mohou rozpoznávat a ovládat aplikace a služby, což umožňuje správcům definovat zásady na základě konkrétních aplikací nebo kategorií aplikací. To pomáhá při prosazování přesnější kontroly přístupu. [58] [60]
- **Povědomí o uživateli a identitě** NGFW se mohou integrovat se systémy správy identit a ověřovat uživatele, což umožňuje vynutit zásady založené na identitách jednotlivých uživatelů nebo atributech zařízení.
- **IPS a IDS** NGFW často obsahují funkci IPS a IDS, které umožňují detekovat a blokovat škodlivý provoz a chování v reálném čase. [58]
- **Filtrování obsahu** NGFW obvykle obsahují funkce filtrování obsahu, které umožňují řídit přístup k webům a aplikacím na základě kategorií obsahu. [58]
- **SSL/TLS dešifrování** Mnoho NGFW dokáže dešifrovat a kontrolovat provoz šifrovaný SSL/TLS, což pomáhá odhalovat skryté hrozby. [58]
- **Přehledy a analýzy** NGFW nabízejí robustní protokolovací, reportovací a analytické nástroje, které poskytují přehled o síťové aktivitě a bezpečnostních incidentech. [60]

NGFW jsou navrženy tak, aby ve srovnání s tradičními firewally poskytovaly pokročilejší a podrobnější možnosti zabezpečení, díky čemuž se lépe hodí pro moderní výzvy v oblasti zabezpečení sítě. Výběr mezi firewallem a NGFW však závisí na konkrétních bezpečnostních potřebách organizace a na složitosti jejího síťového prostředí. [59]

Filtrování paketů

Pakety obsahují data, která vstupují do sítě, brány firewall je kontrolují a blokují nebo jim umožňují zabránit pronikání škodlivého obsahu, jako je útok malwaru. Všechny firewally mají tuto schopnost filtrování paketů. [61]

Filtrování paketů funguje tak, že kontroluje zdrojové a cílové IP adresy, porty a protokoly spojené s každým paketem – jinými slovy, odkud každý paket pochází, kam směřuje a jak se tam dostane. Firewally povolují nebo blokují pakety na základě tohoto hodnocení a odfiltrují nepovolené pakety. [61]

Útočníci se například někdy pokoušejí zneužít zranitelnosti související s protokolem RDP (Remote Desktop Protocol) odesíláním speciálně vytvořených paketů na port používaný tímto protokolem,

port 3389. Firewall však může paket zkontrolovat a zjistit, na který port jde a blokovat všechny pakety směrované na tento port, pokud nepocházejí z konkrétně povolené IP adresy. To zahrnuje kontrolu síťového provozu na vrstvách 3 (pro zobrazení zdrojové a cílové IP adresy) a 4 (pro zobrazení portu) ISO/OSI. [60]

Hluboká kontrola paketů (DPI)

NGFW vylepšují filtrování paketů tím, že místo toho provádí hloubkovou kontrolu paketů (DPI). Stejně jako filtrování paketů, DPI zahrnuje kontrolu každého jednotlivého paketu, aby byla vidět zdrojová a cílová IP adresa, zdrojový a cílový port a tak dále. Všechny tyto informace jsou obsaženy v záhlavích vrstvy 3 a 4 paketu ISO/OSI. [6]

DPI ale také kontroluje tělo každého paketu, nejen hlavičku. Konkrétně DPI kontroluje těla paketů na přítomnost malwarových signatur a dalších potenciálních hrozeb. Porovnává obsah každého paketu s obsahem známých škodlivých útoků. [62]

3.4 Zajímavé technologie na trhu

Na závěr této kapitoly zmíníme několik zajímavých technologií dodavatelů NGFW.

Palo Alto Networks

Palo Alto Networks PA-Series NGFW je navržen tak, aby poskytoval komplexní zabezpečení sítě, které přesahuje tradiční firewally. Mezi klíčové technologie patří: [63]

- **App-ID** Jednou z charakteristických vlastností PA-Series NGFW je její schopnost identifikovat a řídit aplikace procházející sítí, bez ohledu na port, protokol nebo únikové techniky. [63]
- **User-ID** Funkce User-ID umožňuje řadě PA identifikovat uživatele v síti, i když se pohybují mezi fyzickým a virtualizovaným prostředím. Přidružením síťového provozu ke konkrétním uživatelům mohou organizace implementovat podrobnější bezpečnostní zásady, které zajistí ochranu citlivých dat a zabránění neoprávněnému přístupu. [63]
- **Content-ID** Content-ID zodpovídá za skenování obsahu v aplikacích za účelem zjištění a blokování malwaru, phishingových útoků a dalšího škodlivého obsahu. Využívá pokročilé techniky prevence před hrozbami, aby zajistil, že škodlivá data budou identifikována a neutralizována dříve, než mohou způsobit škodu. [63]
- **WildFire** WildFire je pokročilá služba analýzy hrozeb Palo Alto Networks, která automaticky analyzuje a identifikuje neznámé a potenciálně škodlivé soubory. Když je soubor odeslán do WildFire, je spuštěn v zabezpečeném prostředí a je analyzováno jeho chování. Pokud

je to považováno za škodlivé, informace o hrozbách jsou sdíleny v celém ekosystému Palo Alto Networks. [63]

- **Dešifrování SSL** Vzhledem k nárůstu šifrovaného provozu je schopnost dešifrovat a kontrolovat provoz SSL/TLS klíčová. PA-Series NGFW nabízí dešifrování SSL, které organizacím umožňuje kontrolovat zašifrovaný obsah, zda neobsahuje potenciální hrozby, aniž by došlo k ohrožení zabezpečení. [63]
- **Panorama Management** Platforma pro správu Panorama umožňuje centralizovanou správu a monitorování více zařízení NGFW řady PA, čímž usnadňuje velkým podnikům udržovat konzistentní bezpečnostní zásady a získat komplexní přehled o stavu zabezpečení sítě. [63]

Cisco Firepower NGFW

Cisco Systems, Inc., běžně známá jako Cisco, je jednou z nejvlivnějších a nejtransformativnějších společností v technologickém sektoru. Společnost, kterou v roce 1984 založili Leonard Bosack a Sandy Lerner, sehrála nedílnou roli při utváření způsobu, jakým se svět spojuje a komunikuje. Pojďme se podívat ma technologie, které řada Firepower nabízí. [64]

- **Threat Intelligence** Cisco Firepower NGFW využívá rozsáhlou síť zdrojů informací o hrozbách k identifikaci a reakci na vznikající hrozby. Neustále aktualizuje svou databázi hrozeb, aby si udržela náskok před neustále se vyvíjejícím prostředím hrozeb. Tato inteligence o hrozbách v reálném čase umožňuje firewallu detekovat a blokovat škodlivé aktivity v rané fázi, čímž se snižuje riziko úspěšných kybernetických útoků. [65]
- **Application Visibility and Control** Jednou z výjimečných funkcí Cisco Firepower NGFW je Application Visibility and Control (AVC). AVC poskytuje možnost vidět a pochopit, jaké aplikace běží v jejich sítích. To zahrnuje identifikaci aplikací odpovědných za provoz, jejich vzorce používání a jejich dopad na výkon sítě. Viditelnost síťového provozu je zásadní pro diagnostiku síťových problémů, optimalizaci výkonu a informovaná rozhodnutí o správě sítě. AVC používá různé metody pro klasifikaci síťového provozu do konkrétních kategorií aplikací. Toho lze dosáhnout pomocí technik, jako je hloubková kontrola paketů, analýza toku a identifikace na základě podpisu. Díky kategorizaci síťového provozu na základě aplikací mohou správci získat přehled o tom, které aplikace spotřebovávají síťové zdroje a potenciálně způsobují problémy. Jakmile je síťový provoz klasifikován, je možné implementovat zásady pro řízení chování konkrétních aplikací. Můžeme například rozhodnout upřednostnit kritické obchodní aplikace a omezit šířku pásma přidělenou nepodnikatelským aplikacím, aby zajistila optimální výkon. AVC často pracuje ve spojení s mechanismy Quality of Service (QoS). QoS umožňuje správcům přidělovat síťové zdroje, jako je šířka pásma, latence a ztráta paketů, aby bylo zajištěno, že kritické aplikace obdrží potřebné síťové zdroje pro hladký a spolehlivý provoz. Řešení AVC poskytují podrobné reportovací a analytické funkce. Uživatelé mají přístup k údajům o využití aplikací, výkonu sítě a bezpečnostních

incidentech. Tato data jsou nezbytná pro rozhodování na základě dat a optimalizaci síťových zdrojů. [65]

- **Integrace s Cisco Talos** Cisco Firepower NGFW těží z integrace s Cisco Talos, přední službou pro analýzu hrozeb. Tato spolupráce poskytuje nepřetržité aktualizace a odbornou analýzu vznikajících hrozeb. Pomáhá zůstat o krok napřed před kyberzločinci tím, že využívá rozsáhlou globální síť Cisco pro informace o hrozbách. [65]

Fortinet FortiGate NGFW

FortiGate je vlajkovou lodí firewallů společnosti Fortinet, která si získala reputaci pro svou všestrannost a efektivitu v obraně proti dnešním komplexním a sofistikovaným kybernetickým hrozbám. Tyto firewally nové generace nabízejí širokou škálu bezpečnostních služeb a funkcí, díky čemuž jsou nezbytnou součástí zabezpečení sítě v moderním digitálním prostředí. [66]

- **Unified Threat Management (UTM)** Zařízení FortiGate jsou navržena tak, aby konsolidovala bezpečnostní funkce do jediné, snadno spravovatelné platformy. To zjednodušuje správu zabezpečení a snižuje náklady a složitost související s údržbou více bezpečnostních řešení. [67]
- **Security Fabric** FortiGate se hladce integruje s Security Fabric. Jedná se o komplexní bezpečnostní ekosystémem, který zlepšuje viditelnost a kontrolu v celé síti. To umožňuje konzistentní bezpečnostní zásady a sdílení informací o hrozbách. [68]

SonicWall SuperMassive

SonicWall je společnost založená v roce 1991, vybudovala inovaci v odvětví kybernetické bezpečnosti. Společnost je známá svými špičkovými řešeními zabezpečení sítě, zabezpečení přístupu, cloudového zabezpečení a zabezpečení koncových bodů. Závazek společnosti SonicWall udržet si náskok před vznikajícími hrozbami z ní učinil preferovanou volbu pro organizace, které hledají robustní opatření v oblasti kybernetické bezpečnosti. [69]

- **Advanced Threat Protection Service** Jednou z výjimečných vlastností SuperMassive NGFW je jeho sandboxová technologie. Vytváří izolované prostředí pro spouštění a analýzu podezřelých souborů v reálném čase. Sledováním chování těchto souborů může brána firewall přesně určit, zda jsou škodlivé. Tento proaktivní přístup k detekci hrozeb zajišťuje, že útoky zero-day, které využívají zranitelnosti dříve, než jsou známy, jsou rychle identifikovány a neutralizovány. [70]
- **Detekce a blokování příkazů a řízení botnetu** SonicWall poskytuje schopnost identifikovat a blokovat příkazy a řídit provoz pocházející z botů v místní síti na IP a domény, které jsou identifikovány jako šířící malware. [71]

4 SW pro analýzu síťového provozu

Moderní podnikové sítě se stávají stále složitějšími, protože přibývají četná osobní zařízení, množství cloudových aplikací, přístupové body, virtuální servery a další. Pro správce je důležité pečlivé řízení šířky pásma a kvality služeb, implementovat nástroje pro vyrovnávání zatížení pro vysokou dostupnost, vytvářet redundantní prostředí pro zálohování a převzetí služeb při selhání a zajistit, aby síť fungovala optimálním způsobem. [6]

Uprostřed této složitosti správcům často chybí jasný přehled o jejich síti. Zde přicházejí na scénu nástroje síťové analýzy. Na počátku sítí byly používány tradiční metody monitorování sítě, jako je Simple Network Management Protocol (SNMP) či sniffování paketů k provádění základních úkolů, shromažďování omezených dat a nezávislé práci. [7]

Moderní nástroje síťové analýzy shromažďují data z mnoha zdrojů a analyzují tato data v reálném čase proti různým prahovým hodnotám. Tyto nástroje umožňují např. sledování využití šířky pásma a jeho úzká místa, kontrola rychlosti a dostupnost Wi-Fi, odhalování bezpečnostních hrozeb anebo i komplexní troubleshooting při chybném připojení. [6]

Některé nástroje síťové analýzy také navrhují možné způsoby, jak zlepšit výkon sítě. Ve srovnání s tradičními nástroji nabízejí tato řešení proaktivní monitorování a včasnou reakci ke zmírnění hrozeb a řešení síťových problémů. Tyto nástroje také pomáhají při pokročilém plánování kapacity a přidělování určitého rozpočtu na síťové úkoly. Pojďme se nyní podívat na několik nástrojů pro analýzu sítového provozu. [6] [7]

4.1 Wireshark

Wireshark je analyzátor síťových protokolů nebo aplikace, která zachycuje pakety ze síťového připojení. Wireshark je nejčastěji používaným snifferem paketů na světě. [72]

Koncem 90. let pracoval Gerald Combs, absolvent informatiky na University of Missouri–Kansas City, pro malého poskytovatele internetových služeb. Komerční produkty analýzy protokolů v té době měly cenu kolem $1500 a neběžely na primárních platformách společnosti (Solaris a Linux), takže Gerald začal psát Ethereal a vydal první verzi kolem roku 1998. Ochrannou známku Ethereal vlastní společnost Network Integration Services. [73]

V květnu 2006, Combs přijal práci s CACE Technologies s Loris Degioanni. Combs stále držel autorská práva na většinu zdrojového kódu Etherealu (a zbytek byl redistribuovatelný pod GNU GPL), takže jako základ pro Wireshark použil subverzi Etherealu. Nevlastnil však ochrannou

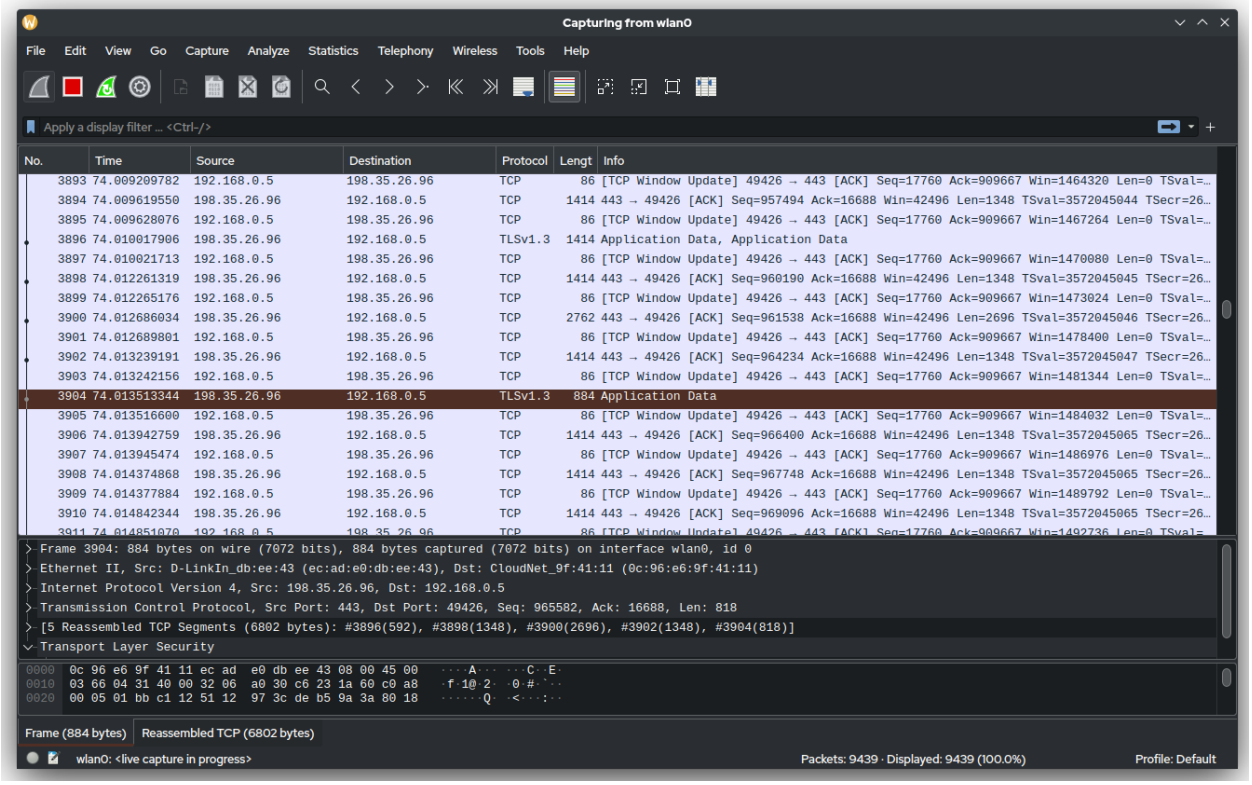
známku Ethereal, a tak změnil název na Wireshark. V roce 2010 Riverbed Technology koupil CACE a stal se tak primárním sponzorem pro Wireshark. Vývoj Etherealu se tak zastavil. [73]

Wireshark v průběhu let vyhrál několik průmyslových ocenění, včetně eWeek, InfoWorld a PC Magazine. Je také nejlépe hodnoceným snifferem paketů v průzkumu nástrojů zabezpečení sítě Insecure.Org a byl projektem měsíce SourceForge v srpnu 2010. [72]

Combs nadále udržuje celkový kód Wireshark a vydává nové verze softwaru. Webová stránka produktu uvádí více než 2000 přispívajících autorů. [72]

Wireshark má mnoho využití, včetně řešení problémů se sítěmi, které mají problémy s výkonem. Správci často používají Wireshark ke sledování připojení, zobrazení obsahu podezřelých síťových transakcí a identifikaci výpadků síťového provozu. [72]

Wireshark je bezpečný nástroj používaný vládními agenturami, vzdělávacími institucemi, korporacemi, malými podniky a neziskovými organizacemi. Navíc lze Wireshark použít jako výukový nástroj či pro vývoj nových či ladění protokolů. [72]



Obrázek 4.1: Wireshark [74]

Klíčové funkce

Zde jsou některé z mnoha funkcí, které Wireshark nabízí:

- **Živé zachycení z mnoha různých síťových médií** Wireshark dokáže zachytit provoz z mnoha různých typů síťových médií, včetně Ethernetu, bezdrátové sítě LAN, Bluetooth, USB a dalších. Konkrétní podporované typy médií mohou být omezeny několika faktory,

včetně vašeho hardwaru a operačního systému. Wireshark dešifruje strukturu každého zachyceného paketu a rozloží jej na jednotlivé části. Identifikuje a zobrazuje různé hlavičky protokolů, včetně Ethernetu, IP, TCP, UDP a dalších. Tento hierarchický přístup umožňuje analytikům porozumět celému procesu přenosu dat. Kromě informací v hlavičce jde Wireshark hlouběji a poskytuje podrobné informace o užitečné zátěži paketů. Může například extrahovat a zobrazovat obsah požadavků a odpovědí HTTP, dotazů a odpovědí DNS a dalších dat na aplikační vrstvě. To je neocenitelné pro diagnostiku problémů v konkrétních síťových protokolech. [75]

- **Analýza konverzace**

  Zobrazení konverzace aplikace Wireshark umožňuje analytikům sledovat interakce mezi koncovými Endpointy. Poskytuje statistiky o vzorcích komunikace, pomáhá identifikovat anomálie a úzká místa výkonu.

  - **Grafy toků** Grafy toků vizualizují vztahy mezi pakety, což usnadňuje pochopení složitých interakcí v síti. Tyto grafy mohou odhalit vzorce komunikace a potenciální problémy
  - **Podrobné informace** - Funkce „expertních informací“ Wiresharku upozorňuje na potenciální problémy v zachycených datech, jako jsou chybně tvarované pakety nebo problémy specifické pro protokol. Analytici mohou tyto obavy rychle identifikovat a řešit.
  - **Analýza VoIP** - Wireshark obsahuje specifické nástroje pro analýzu provozu Voice over IP (VoIP), díky čemuž je neocenitelný pro hodnocení kvality hovorů VoIP a diagnostiku problémů v systémech hlasové komunikace. [75]

- **Přizpůsobení a rozšiřitelnost**

  - **Skriptování Lua** - Wireshark podporuje skriptování pomocí programovacího jazyka Lua. To umožňuje vytvářet vlastní disektory pro proprietární protokoly, automatizovat opakující se úlohy a přizpůsobovat Wireshark jejich specifickým potřebám.
  - **Pluginy třetích stran** - Komunita Wireshark vyvinula četné pluginy a disektory pro specializované úkoly. Tato rozšíření mohou dále vylepšit možnosti Wiresharku a zefektivnit specifické analýzy. [75]

- **Možnost importu paketů z textových souborů obsahujících hex výpisů dat paketů**
- **Zobrazení packetů s podrobnými informacemi o protokolu**
- **Ukládání zachycených dat a následného včetně exportu**
- **Bohaté filtrování**
- **Software s otevřeným zdrojovým kódem**

  Wireshark je projekt s otevřeným zdrojovým kódem a je vydán pod licencí GNU General Public License (GPL). Wireshark můžete volně používat na libovolném počtu počítačů

bez obav o licenční klíče nebo poplatky a podobně. Veškerý zdrojový kód je navíc volně dostupný pod licencí GPL. [75]

Limitace

Wireshark nemůže za normálních okolností zachytit provoz ze všech ostatních systémů v síti. V sítích, které používají zařízení switch, může snímat provoz pouze mezi vaším místním počítačem a vzdáleným systémem, se kterým komunikuje.

I když Wireshark může zobrazit poškozené pakety a použít barevné kódování, nemá skutečné výstrahy - Wireshark není systém detekce narušení.

Wireshark nemůže pomoci s dešifrováním, pokud jde o šifrovaný provoz.

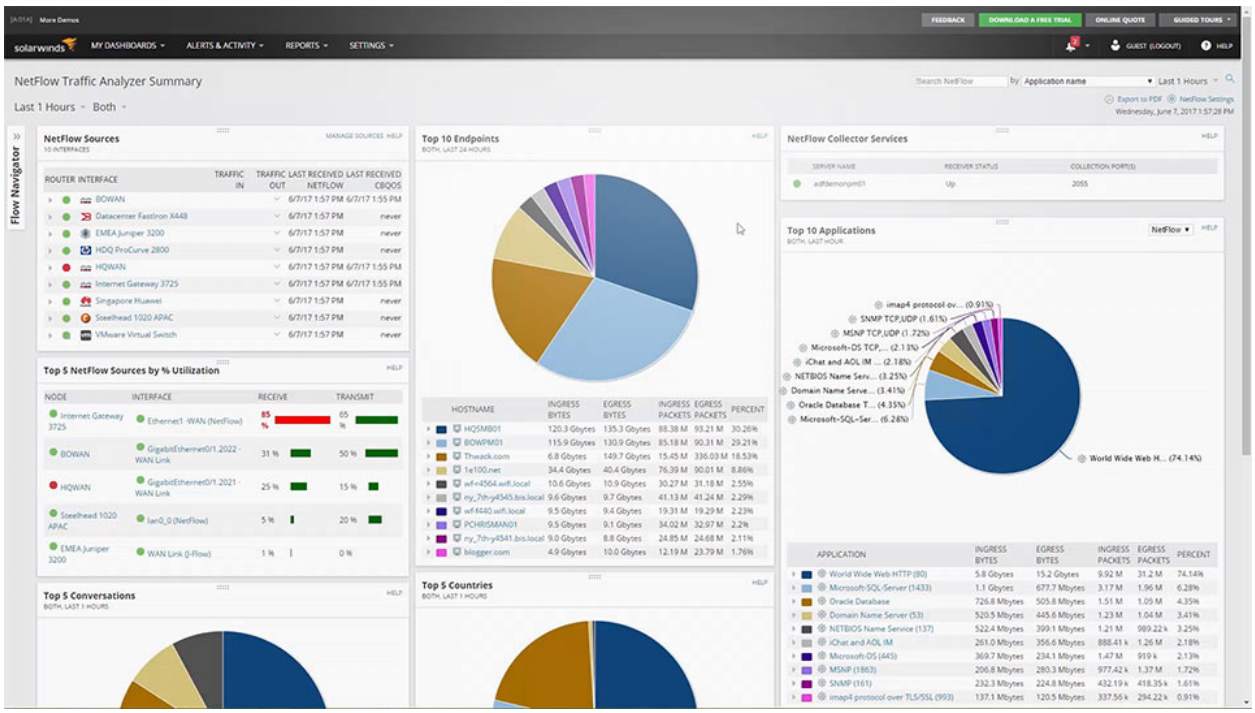
Wireshark vyniká v poskytování podrobné analýzy na úrovni paketů, dekódování protokolů a široké škály funkcí pro správce či analytiky. Jeho open source povaha, rozsáhlá komunitní podpora a kompatibilita napříč platformami z něj činí výkonnou volbu pro analýzu síťového provozu, pouze však na síti, na které je stroj připojen.

4.2 SolarWinds NetFlow Traffic Analyzer

Oproti Wiresharku, NetFlow Traffic Analyzer je komplexněší nástroj pro monitorování, analýzu a optimalizaci síťového provozu.

Než prozkoumáme NetFlow Traffic Analyzer, pojďme si stručně představit NetFlow. NetFlow je síťový protokol původně vyvinutý společností Cisco Systems. Umožňuje síťovým zařízením, jako jsou směrovače a přepínače, shromažďovat a exportovat informace o tocích IP provozu v reálném čase. Tok představuje sekvenci paketů, které sdílejí společné charakteristiky, jako jsou zdrojové a cílové IP adresy, porty a protokoly. NetFlow poskytuje cenná data pro pochopení vzorců síťového provozu, využití zdrojů a potenciálních bezpečnostních hrozeb. [76]

SolarWinds NetFlow Traffic Analyzer je navržen tak, aby bezproblémově spolupracoval s protokolem NetFlow, díky čemuž je kompatibilní s celou řadou síťových zařízení. [77]



Obrázek 4.2: SolarWinds NetFlow Traffic Analyzer [78]

Klíčové funkce

- **Sledování provozu v reálném čase**

  Jednou z primárních funkcí NetFlow Traffic Analyzeru je monitorování síťového provozu v reálném čase. Shromažďuje data o provozních tocích procházejících sítí a nabízí přehled o tom, kdo komunikuje, jaké služby nebo aplikace používá a jak velkou šířku pásma spotřebovává.

- **Statistiky využití šířky pásma**

  Pochopení toho, jak se využívá šířka pásma sítě, je zásadní pro optimalizaci výkonu sítě. NetFlow Traffic Analyzer poskytuje podrobnosti o využití šířky pásma. Pomáhá určit, které aplikace, uživatelé nebo zařízení jsou hlavními spotřebiteli síťových zdrojů, což umožňuje proaktivní správu šířky pásma. [79]

- **Analýza historických dat**

  Kromě monitorování v reálném čase nástroj ukládá historická data NetFlow. Tato historická perspektiva umožňuje správcům sítě analyzovat vzorce provozu a trendy v čase. Můžete přesně určit, kdy a kde dochází ke špičkám provozu, což pomáhá při plánování kapacity a přidělování zdrojů.

- **Vlastní upozornění a prahové hodnoty**

  Proaktivní správa sítě je usnadněna pomocí vlastních výstrah a prahových hodnot. Správci sítě mohou konfigurovat výstrahy na základě konkrétních podmínek sítě nebo vzorců

provozu. Když jsou tyto prahové hodnoty překročeny nebo jsou splněna určitá kritéria, systém generuje výstrahy, které zajišťují rychlé reakce na problémy se sítí.

- **Bezpečnostní monitorování**

  Zabezpečení sítě je hlavním zájmem organizací. NetFlow Traffic Analyzer také hraje roli v bezpečnostním monitorování. Analýzou vzorců provozu a anomálií může pomoci odhalit potenciální bezpečnostní hrozby nebo neoprávněný přístup, což přispívá k robustnějšímu zabezpečení. [77]

- **Reporting a dashboardy**

  Vizualizace dat o síťovém provozu je snadná díky přizpůsobitelným sestavám a řídicím panelům. Uživatelsky přívětivé rozhraní SolarWinds vám umožňuje vytvářet vizuální reprezentace metrik síťového provozu, což usnadňuje sdělování poznatků zúčastněným stranám.

- **Integrační schopnosti**

  SolarWinds je známý svým přístupem vstřícným k integraci. NetFlow Traffic Analyzer lze bezproblémově integrovat s dalšími produkty SolarWinds a nástroji třetích stran. Toto umožní vybudovat komplexní ekosystém správy sítě přizpůsobený potřebám organizace.

- **Škálovatelnost**

  Ať už jde o malou síť nebo velké podnikové prostředí, NetFlow Traffic Analyzer je navržen tak, aby se přizpůsobil požadavkům sítě. Vyhovuje sítím různé velikosti a složitosti. [79]

Limitace

Zatímco SolarWinds NetFlow Traffic Analyzer je výkonný nástroj pro monitorování a analýzu síťového provozu, jako každý software, má omezení. Tato omezení mohou v určitých situacích ovlivnit jeho účinnost.

- **Závislost na zařízeních kompatibilních s NetFlow**

  SolarWinds NetFlow Traffic Analyzer spoléhá na síťová zařízení, která podporují NetFlow nebo jiné protokoly založené na toku. Pokud síťové vybavení nepodporuje tyto protokoly, nelze shromažďovat data o toku, takže nástroj bude neúčinný. [77]

- **Limited Packet-Level Detail**

  NetFlow poskytuje agregovaná data toku, což znamená, že postrádá úroveň detailů, jakou mají nástroje pro analýzu na úrovni paketů, jako je Wireshark. Pokud uživatel požaduje hloubkovou kontrolu paketů pro řešení problémů nebo forenzní analýzu, bude muset NetFlow Traffic Analyzer doplnit o nástroje pro zachycení paketů. [79]

- **Zpoždění dat v reálném čase**

  Data NetFlow jsou typicky shromažďována a pravidelně odesílána ze síťových zařízení do analyzátoru. Při přijímání dat v reálném čase může docházet ke zpožděním, což může ovlivnit schopnost okamžitě reagovat na problémy se sítí.

- **Režie na síťových zařízeních**

  Povolení NetFlow na síťových zařízeních může představovat určitou režii, protože zařízení potřebují zpracovávat a exportovat toková data. U vysoce zatížených zařízení to může potenciálně ovlivnit výkon zařízení.

- **Omezená viditelnost šifrovaného provozu**

  Data NetFlow nemusí poskytovat přehled o obsahu šifrovaného provozu. Jsou dostupná metadata o šifrovaných tocích, ale nelze kontrolovat skutečná data v šifrovaných paketech. [76]

- **Složitost integrace**

  Zatímco produkty SolarWinds jsou známé svými integračními schopnostmi, nastavení a údržba integrace NetFlow Traffic Analyzer s různými síťovými zařízeními a dalšími systémy může být složitá a může vyžadovat pokročilou konfiguraci.

- **Nedostatek hloubkové analýzy protokolu**

  I když poskytuje cenné informace o tocích a vzorcích provozu, NetFlow Traffic Analyzer nenabízí možnosti hloubkové analýzy protokolů. Nepomůže podrobně řešit problémy na úrovni protokolu.

- **Náklady**

  SolarWinds NetFlow Traffic Analyzer je placený produkt a cena se může lišit v závislosti na rozsahu sítě a specifických funkcích, které jsou požadovány. Menším organizacím to může připadat relativně drahé.

Můžeme si nyní porovnat tento nástroj oproti Wiresharku.

Wireshark je nástroj, který je lépe určen pro potřebu hloubkové analýzy na úrovni paketů pro odstraňování problémů, ladění protokolů nebo forenzní analýzu. Je to nezbytné pro identifikaci a řešení složitých problémů se sítí v reálném čase.

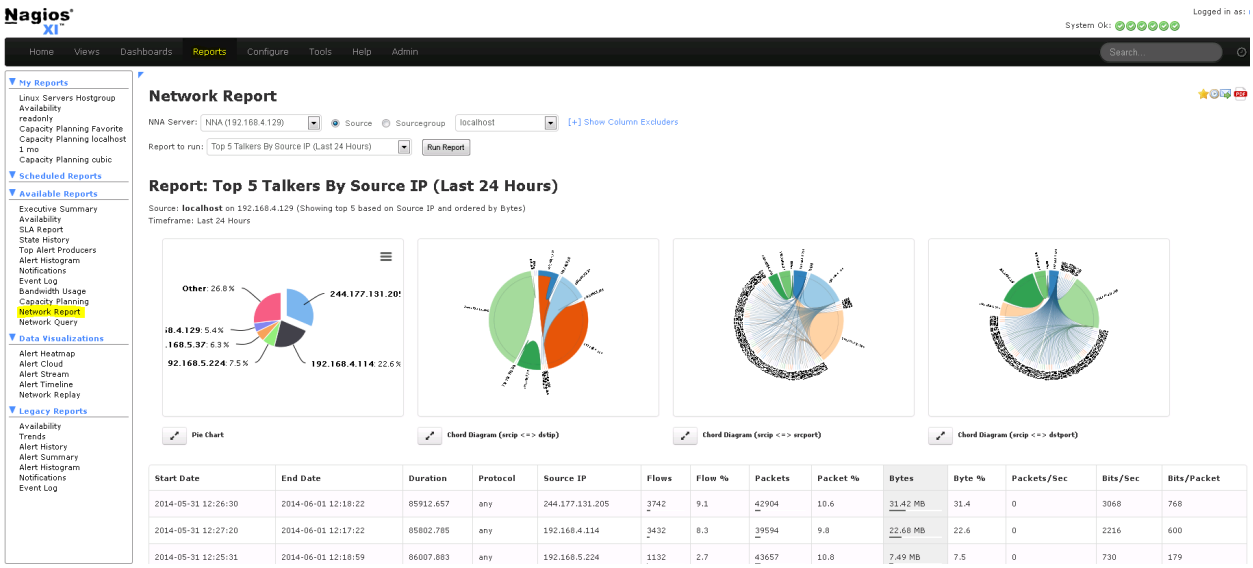
SolarWinds NetFlow Traffic Analyzer je lepší, když je potřeba širšího přehledu o vzorcích síťového provozu, využití šířky pásma a historických trendech. Je to cenné pro dlouhodobé plánování kapacity, monitorování zabezpečení a optimalizaci výkonu sítě.

V praxi, v mnoha případech správci sítě používají oba nástroje a využívají Wireshark pro analýzu paketů v reálném čase, když se objeví problémy, a SolarWinds NetFlow Traffic Analyzer pro průběžnou správu výkonu sítě. Tento přístup poskytuje komplexní pohled na stav sítě a zajišťuje, že jste dobře připraveni na řešení jakýchkoli problémů, které mohou nastat. [80]

Správná volba nakonec závisí na požadavcích na správu a analýzu sítě a kombinace obou nástrojů může nabídnout nejúplnější řešení síťové analýzy.

4.3 Nagios Network Analyzer

Nagios Network Analyzer je software pro ekosystém Nagios, který přináší schopnost monitorovat a analyzovat síťový provoz. Jedná se o všestranný nástroj pro monitorování a analýzu sítě vyvinutý s cílem poskytnout komplexní pohled na výkon a zabezpečení sítě. Hraje klíčovou roli v širší sadě Nagios, která je známá svými monitorovacími schopnostmi. Zejména Network Analyzer se specializuje na zkoumání síťového provozu a nabízí vhled do různých aspektů analýzy i správy sítě. [81]



Obrázek 4.3: Nagios Network Analyzer [82]

Klíčové funkce

Pojďme prozkoumat klíčové funkce a možnosti, díky kterým je Nagios Network Analyzer cenným aktivem pro správce sítí a IT profesionály:

- **Analýza síťového provozu**

  Srdcem Nagios Network Analyzer je jeho schopnost zachytit a analyzovat síťový provoz v reálném čase. Tato schopnost umožňuje správcům získat hluboké znalosti o tom, jak data procházejí jejich síťovou infrastrukturou, což je zásadní pro optimalizaci výkonu sítě a identifikaci úzkých míst.

- **Sledování šířky pásma**

  Efektivní správa šířky pásma je zásadní pro zajištění efektivní alokace síťových zdrojů. Network Analyzer umožňuje organizacím monitorovat využití šířky pásma, identifikovat aplikace nebo uživatele náročné na šířku pásma a činit informovaná rozhodnutí k optimalizaci síťových zdrojů.

- **Bezpečnostní analýza**

Narušení bezpečnosti a hrozby jsou v digitálním prostředí všudypřítomné obavy. Nagios Network Analyzer pomáhá organizacím detekovat a reagovat na bezpečnostní incidenty identifikací anomálií, pokusů o narušení a potenciálních zranitelností v rámci sítě. Tento proaktivní přístup k zabezpečení je zásadní pro ochranu citlivých dat. [81]

- **Výstrahy a upozornění**

  Network Analyzer umožňuje správcům nastavit vlastní výstrahy a upozornění na základě předem definovaných prahových hodnot. Tento proaktivní výstražný systém zajišťuje, že problémy se sítí jsou okamžitě vyřešeny, čímž se minimalizují prostoje a narušení.

- **Historická data a reporting**

  Ukládání historických dat sítě je zásadní pro sledování trendů, přijímání informovaných rozhodnutí a řešení problémů s historickým výkonem. Možnosti uchovávání dat Network Analyzer usnadňují generování zpráv a analýz v průběhu času, což pomáhá při plánování kapacity a dlouhodobé optimalizaci sítě. [81]

- **Integrace**

  Nagios Network Analyzer se hladce integruje s ostatními produkty Nagios a nástroji třetích stran. Tato integrační schopnost umožňuje organizacím vytvářet jednotné, komplexní řešení pro monitorování a správu sítě přizpůsobené jejich specifickým potřebám. [81]

- **Compliance a audit**

  Splnění norem je pro mnoho organizací nejvyšší prioritou. Network Analyzer v tomto ohledu pomáhá tím, že poskytuje data pro účely auditu a zajišťuje, že síťové aktivity splňují bezpečnostní a regulační standardy. [81]

Limitace

Jako každý software i Network Analyzer má svá omezení.

- **Cena**

  Nagios Network Analyzer je komerční nástroj, což znamená, že s jeho licencováním a průběžnou podporou jsou spojeny náklady. Tyto náklady mohou být omezením pro malé organizace nebo organizace s omezeným rozpočtem.

- **Složitost**

  Přestože Nagios Network Analyzer nabízí vysoký stupeň přizpůsobení a flexibility, jeho počáteční nastavení a konfigurace mohou být komplikované, zejména pro uživatele, kteří jsou v ekosystému Nagios noví.

- **Intenzivní na zdroje**

  Nagios Network Analyzer může být náročný na zdroje, zejména při monitorování a analýze velkého objemu síťového provozu. To může vyžadovat, aby organizace investovaly do

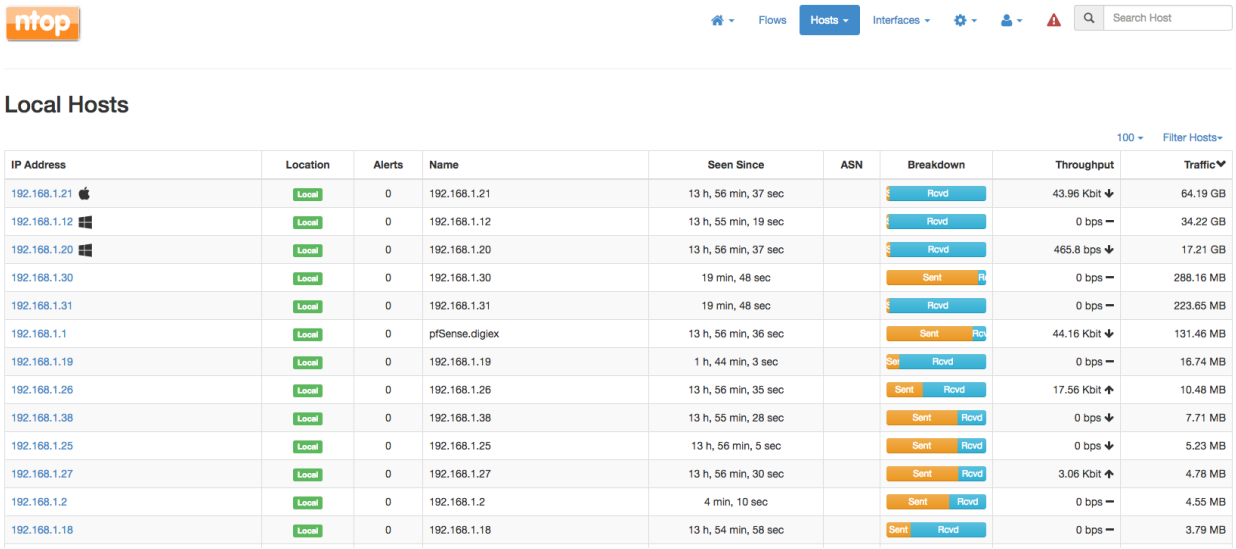
robustní hardwarové infrastruktury, aby zvládly zátěž. Nagios Network Analyzer je navržen tak, aby poskytoval analýzu síťového provozu v reálném čase, ale zpracování a analýza velkých objemů dat v reálném čase může být náročná na systémové zdroje. To může vést ke zpožděním při zpracování a analýze dat, případně k chybějícím kritickým událostem. [81]

- **Závislost na ekosystému Nagios**

  Aby bylo možné uvolnit svůj plný potenciál, Nagios Network Analyzer se často integruje s dalšími produkty Nagios nebo nástroji třetích stran. Tato závislost na ekosystému Nagios může omezit efektivitu nástroje pro organizace, které preferují více samostatné řešení. [81]

4.4 ntop

ntop (zkratka pro network top) je účinný nástroj pro analýzu síťového provozu, který rozšiřuje možnosti firewallu pfSense tím, že poskytuje monitorování a analýzu sítě v reálném čase. [83]



Obrázek 4.4: ntop[84]

Klíčové vlastnosti

- **Analýza provozu v reálném čase**

  pfSense poskytuje nezbytný firewall a možnosti směrování, zatímco ntop je doplňuje tím, že nabízí viditelnost síťového provozu procházejícího firewallem v reálném čase. Tato synergie zajišťuje, že je možné monitorovat a reagovat na síťové události, jakmile k nim dojde. Ntop vyniká zobrazováním informací o síťovém provozu v reálném čase, identifikací hlavních mluvčích a používaných protokolů. [85]

  - **Granulární analýza**

Schopnost ntop rozdělit síťový provoz na jemné detaily, což pomůže získat přehled o povaze síťového provozu. Tyto informace jsou neocenitelné pro rozhodování na základě dat ohledně optimalizace šířky pásma, výkonu aplikací a bezpečnostních opatření.

- **Historická analýza**

  Když jsou historická data sítě vyžadována pro vyšetřování, shodu nebo analýzu výkonu, do hry vstupují možnosti uchovávání dat ntop. Efektivně ukládá historická data a zpřístupňuje je pro budoucí použití.

- **Profilování provozu**

  ntop poskytuje podrobné informace o typech provozu v síti, což pomůže pochopit, které aplikace a protokoly spotřebovávají největší šířku pásma.

- **Statistiky využití šířky pásma**

  Monitorováním spotřeby šířky pásma na úrovni zařízení nebo IP adresy pomáhá identifikovat a řešit potenciální problémy s šířkou pásma. [85]

- **Reporting**

  Generuje podrobné zprávy o využití sítě a vizualizace pro informované rozhodování a účely prezentace.

- Kombinací robustních funkcí firewallu pfSense s analýzou provozu ntop je možné proaktivně identifikovat a zmírnit bezpečnostní hrozby a neobvyklé chování sítě. Výstražný systém ntop vás může upozornit na podezřelé aktivity.

Limitace

Přestože pfSense a ntop nabízejí výkonné možnosti monitorování sítě a zabezpečení, mají určitá omezení a úvahy, které je třeba mít na paměti.

- **Požadavky na zdroje**

  Jak pfSense, tak ntop mohou být náročné na zdroje, zejména při řešení velkého množství síťového provozu. [85]

- **Křivka učení**

  Konfigurace a doladění pfSense i ntop může být složité, zejména pro uživatele, kteří jsou v oblasti zabezpečení a monitorování sítě noví.

- **Hardwarová kompatibilita**

  Ne všechny karty síťového rozhraní (NIC) jsou již po vybalení podporovány pfSense a kompatibilita se může lišit. [85]

- **Škálovatelnost**

  I když jsou pfSense a ntop vhodné pro malé až středně velké sítě, nemusí se bez pečlivého plánování a alokace zdrojů dobře škálovat do extrémně velkých prostředí nebo prostředí s vysokým provozem. [83]

- **Podpora**

  I když je k dispozici komunita uživatelů a dokumentace pro pfSense i ntop, oficiální možnosti podpory jsou omezené, zejména pro bezplatné verze. [85]

- **Kompatibilita s jiným softwarem**

  Integrace pfSense a ntop s jiným softwarem nebo nástroji může vyžadovat další konfiguraci a testování, aby byla zajištěna kompatibilita a zabránilo se konfliktům.

Kombinace pfSense a ntop nabízí výkonné řešení pro jednotlivce i organizace. Zatímco pfSense vytváří robustní firewall a směrovací infrastrukturu, ntop poskytuje základní nástroje pro viditelnost a analýzu.

Ve výsledku bychom hledali takové řešení, které obsahuje „klasickou funkcionalitu analýzy síťového provozu“, je jednodušše integrovatelné do již stávající infrastruktury a spolupracuje s ostatními síťovými zařízeními či softwarem.

Na trhu existuje mnoho dalších softwarů pro analýzu síťového provozu, většina má stejné funkcionality.

5 Vlastní řešení

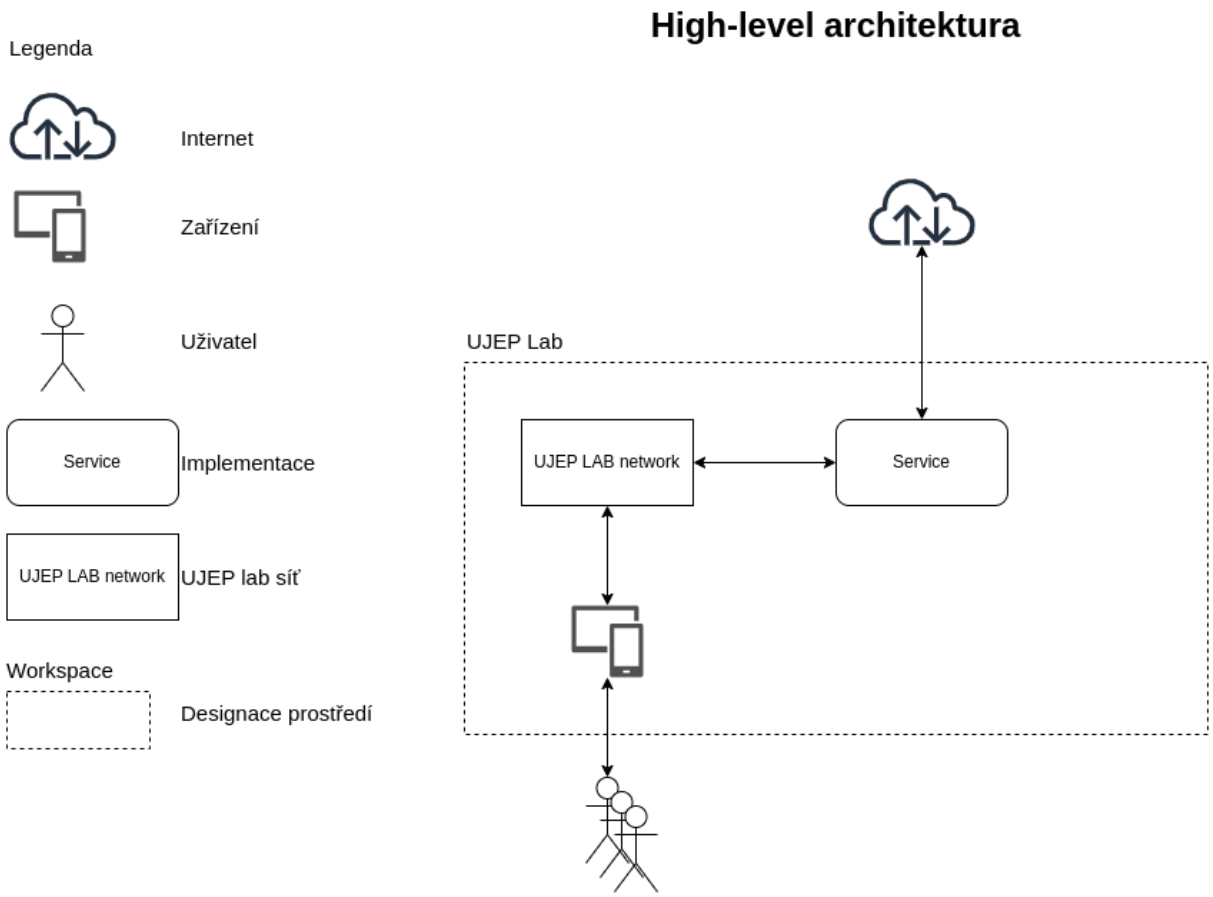
Pro začátek si určíme cíle a očekávání vlastního řešení.

- Řešení musí být postaveno na NGFW a bude obsahovat veškeré QoL funkcionality NGFW; jako jsou:
  - Přehledný dashboard
  - Reporting
  - Bohatý nástroj pro analýzu dat
- Kolektor dat nesmí výrazně snížit či ohrozit chod řešení

Implementační poznámka - Celé řešení bude implementováno na strojích a síti labového prostředí UJEP.

5.1 High level architektura

Podívejme se nejprve na high-level architekturu řešení. High-level architektura (HLA) či design (HLD) vysvětluje architekturu, která bude použita k vývoji systému. Toto schéma poskytuje přehled celého systému, identifikuje hlavní komponenty, které budou vyvinuty v rámci řešení a jejich rozhraní a komunikaci. Jedná se o klíčovou součást procesu vývoje softwaru, hardwaru či celého řešení, protože poskytuje jasnou představu o struktuře a chování systému. [86]



Obrázek 5.1: High-level architektura

Vlastní řešení je v tomto diagramu označeno jako **Service**. Na řešení bude napojeno několik zařízení skrze lab síť. Až teprve řešení bude mít přístup k internetu, tedy bude poskytovat i router.

5.2 Generování síťového provozu

V první fázi si můžeme připravit několik klientů, které budou generovat náhodný síťový provoz. Ke generování síťového provozu byly použity následující skripty.

- PyTgen (https://github.com/reissmann/PyTgen)
- spinneret (https://github.com/steder/spinneret)

Oba skripty jsou volně dostupné na githubu, chtěl bych tímto poděkovat autorům @reissmann a @steder za vytvoření volně dostupného softwaru.

5.3 Router a firewall (NGFW)

Jako router a firewall jsem si vybral OPNsense. OPNsense je open source software, toto byl hlavní důvod rozhodnutí, protože je možné zasáhnout do zdrojového kódu. OPNsense je firewall a router

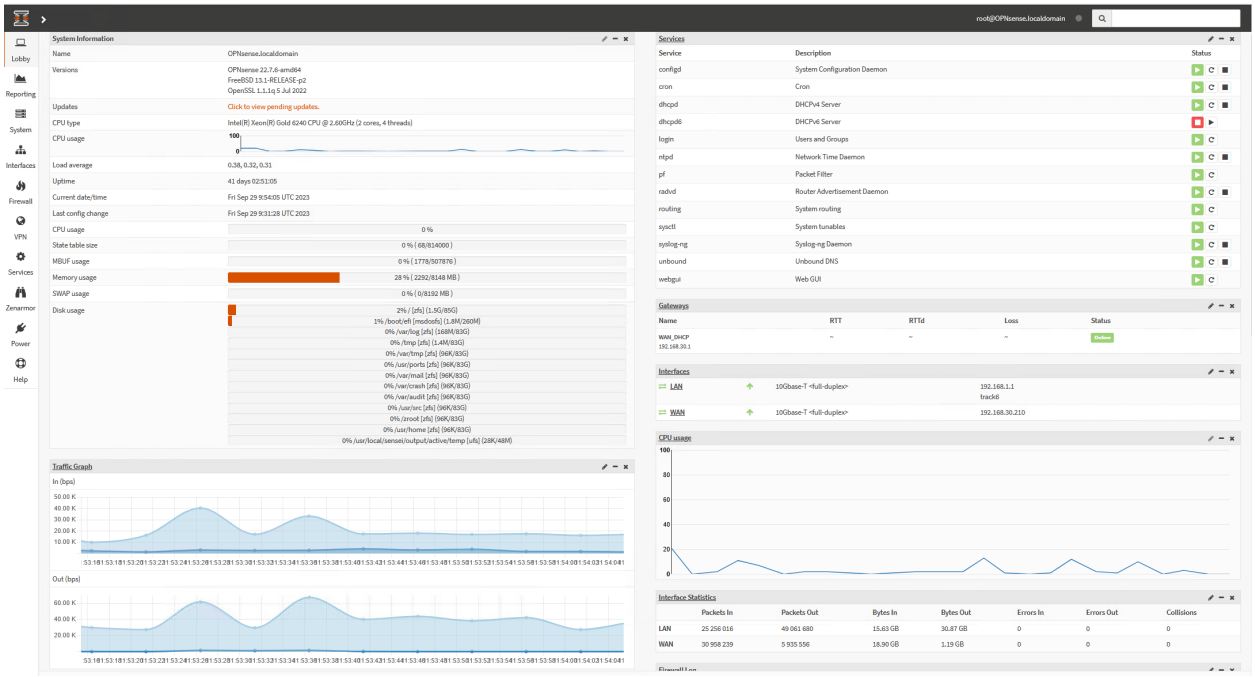
založená na FreeBSD. Obsahuje většinu funkcí dostupných v drahých komerčních firewallech a v mnoha případech i další.

OPNsense začal jako fork pfSense a m0n0wall v roce 2014, s jeho prvním oficiálním vydáním v lednu 2015. Projekt se vyvíjel velmi rychle a stále si zachovává známé aspekty m0n0wall a pfSense. Silné zaměření na bezpečnost a kvalitu kódu řídí vývoj projektu. [87]

OPNsense nabízí pravidelné aktualizace zabezpečení i s malými vylepšeními, aby bylo možné reagovat na nově vznikající hrozby v reálném čase. Pevný cyklus vydání 2 hlavních verzí každý rok nabízí příležitost plánovat upgrady dopředu. [87]

Stejně jako pfSense, OPNsense lze efektivně nasadit v domácích sítích, malých či středních podnicích, korporátních prostředích či datacentrech. Jeho flexibilita je v tomto případě obrovská, pro účely vlastního řešení více než dostačující.

Řídící panel



Obrázek 5.2: Řídící panel

Srdcem OPNsense je jeho dashboard či řídicí panel – centrální prostor, který umožňuje efektivně monitorovat, konfigurovat a ovládat sítě. Řídicí panel OPNsense poskytuje okamžitý přehled o stavu a výkonu sítě.

Jednotlivé metriky je možné přidávat pomocí widgetů. Tyto widgety mohou obsahovat např.:

- **Grafy provozu**

  Grafy provozu v reálném čase zobrazují využití šířky pásma a vzorce provozu pro každé síťové rozhraní. Tyto grafy jsou neocenitelné pro diagnostiku síťových problémů a optimalizaci alokace zdrojů.

- **Stav systému**

  OPNsense monitoruje stav hardwaru a systémových komponent a zobrazuje údaje o teplotě, rychlosti ventilátoru a stavu napájení. Sledování stavu systému pomáhá předcházet selhání hardwaru a prostojům.

- **Systémové protokoly**

  Řídicí panel poskytuje rychlý přístup k systémovým protokolům, což usnadňuje odstraňování problémů a sledování systémových aktivit. Protokoly lze filtrovat a přizpůsobovat, což umožní zaměřit se na konkrétní události nebo chyby. [88]

- **Konfigurace rozhraní**

  Síťová rozhraní lze konfigurovat přímo z řídicího panelu, což zjednodušuje úkoly, jako je nastavení VLAN, přidělování IP adres a konfigurace síťových služeb. To zjednodušuje proces přizpůsobování sítě měnícím se požadavkům.

- **Security Insights**

  OPNsense obsahuje widget Security Insight, který poskytuje informace o aktuálních hrozbách, pokusech o narušení a zablokovaných připojeních. Být informován o potenciálních bezpečnostních rizicích je zásadní pro udržení bezpečného síťového prostředí.

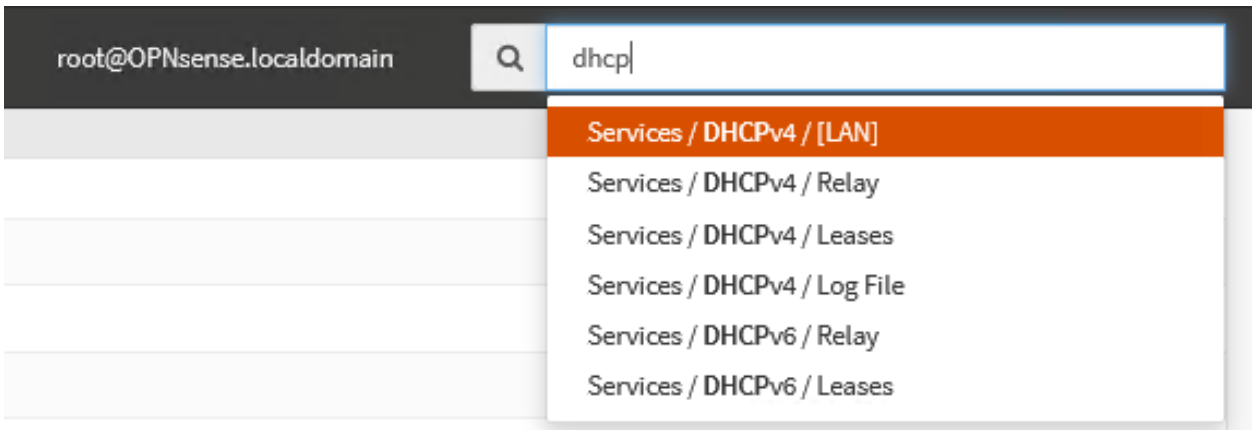
- **Balíčky a pluginy**

  OPNsense podporuje širokou škálu balíčků a pluginů, které rozšiřují jeho funkčnost. Tyto doplňky lze spravovat z řídicího panelu, což usnadňuje instalaci, aktualizaci a konfiguraci dalších funkcí, jako jsou služby VPN, systémy detekce narušení a nástroje pro tvarování provozu. [88]

Řídicí panel je možné přizpůsobit tak, aby vyhovoval konkrétním potřebám přidáním, změnou uspořádání nebo odebráním widgetů. Tím je zajištěno, že informace, které jsou nejdůležitější pro správy sítě, jsou snadno dostupné na první pohled. Ať už dáváte přednost minimalistickému řídicímu panelu nebo panelu plnému monitorovacích nástrojů, OPNsense se přizpůsobí preferencím.

Vyhledávání a dokumentace

Nepostradatelnou funkcionalitou během konfigurace je rychlé vyhledávání. Panel rychlého vyhledávání se nachází v pravém horním rohu webového rozhraní a umožňuje vyhledávat klíčová slova související s dotazem. Během psaní poskytuje OPNsense návrhy v reálném čase, což usnadňuje rychlé nalezení relevantních informací.



Obrázek 5.3: Vyhledávání

Zatímco možnost vyhledávání je vynikajícím nástrojem pro hledání rychlých odpovědí, dokumentace je zdroj pro podrobné informace. Oficiální dokumentace OPNsense je k dispozici na jejich webových stránkách a je pravidelně aktualizována, aby odrážela nejnovější funkce a změny. Pokrývá širokou škálu témat, od instalace a konfigurace až po pokročilé síťové a bezpečnostní koncepty.

OPNsense má prosperující komunitu uživatelů a vývojářů, kteří aktivně přispívají k dokumentaci. Příručky, výukové programy a návody od komunity mohou poskytnout cenné poznatky a alternativní přístupy k řešení konkrétních problémů. Prozkoumání sekce příspěvků komunity v dokumentaci může poskytnout nový pohled a užitečná řešení. [88]

Stateful firewall

Stateful Packet Inspection (SPI) zkoumá pakety v kontextu aktivních připojení, díky čemuž je vysoce efektivní při identifikaci a blokování škodlivého provozu a zároveň umožňuje bezproblémový průchod legitimních dat. Stavový firewall OPNsense nabízí silnou linii obrany pro vaši síť a kombinuje výkonné bezpečnostní funkce s uživatelskou přívětivostí. [88]

Traffic Shaper

Traffic Shaper je komplexní nástroj pro kvalitu služeb (QoS), který umožňuje efektivně řídit, stanovovat priority a přidělovat šířku pásma. Je nedílnou součástí sady funkcí OPNsense, která poskytuje prostředky pro efektivní řízení síťového provozu, zajišťuje, že kritické aplikace obdrží potřebné zdroje, a zároveň zabraňuje tomu, aby nepodstatné služby spotřebovávaly nadměrnou šířku pásma. [88]

Traffic Shaper umožňuje přidělit šířku pásma kritickým aplikacím a službám a zajistit jejich optimální výkon. Pomáhá předcházet zahlcení sítě tím, že řídí, kolik šířky pásma mohou různé aplikace spotřebovat. To zajišťuje, že žádná jednotlivá služba nebude monopolizovat celé připojení, což vede ke spravedlivému rozdělení zdrojů. [6]

Captive Portal

Captive Portal umožňuje vynutit autentizaci nebo přesměrování na proklikávací stránku pro přístup k síti. To se běžně používá v sítích hot spot, ale je také široce používáno v podnikových sítích pro další vrstvu zabezpečení bezdrátového nebo internetového přístupu. OPNsense nabízí většinu podnikových funkcí včetně podpory Radius. [88]

Vysoká dostupnost / selhání hardwaru (CARP)

OPNsense využívá protokol Common Address Redundancy Protocol nebo CARP pro hardwarové převzetí služeb při selhání. Dvě nebo více bran firewall lze nakonfigurovat jako skupinu převzetí služeb při selhání. Pokud jedno rozhraní selže na primárním nebo primární zcela přejde do režimu offline, sekundární se stane aktivním. Využitím této výkonné funkce OPNsense vytváří plně redundantní firewall s automatickým a bezproblémovým převzetím služeb při selhání. Při přepnutí na záložní síť zůstane připojení aktivní s minimálním přerušením pro uživatele. [88]

IDS a IPS

Inline IPS systém OPNsense je založen na Suricata a využívá Netmap ke zvýšení výkonu a minimalizaci využití procesoru. [88]

Integrovaná podpora pro pravidla ETOpen

ETOpen Ruleset je vynikající sada pravidel IDS/IPS proti malwaru, která uživatelům s omezenými náklady umožňuje výrazně zlepšit jejich stávající síťovou detekci malwaru. [88]

Integrovaný SSL black list (SSLBL)

Cílem je poskytnout seznam „špatných“ certifikátů SSL, které server identifikoval jako spojené s malwarem nebo aktivitami botnetu. SSLBL se spoléhá na SHA1 otisky škodlivých SSL certifikátů a nabízí různé black listiny. [88]

Hlášení a monitorování

OPNsense nabízí mnoho možností pro hlášení a monitorování systému, mezi které patří:

- **Zdraví systému** - Moderní pojetí RRD grafů s možností přiblížení a exportu dat.
- **Insight** – Integrovaný Netflow Analyzer - OPNsense také nabízí integrovaný analyzátor Netflow bez potřeby dalších pluginů nebo nástrojů, podobný tomu, co lze najít u špičkových komerčních produktů.

[88]

Firmware a pluginy

OPNsense nabízí robustní cestu upgradu firmwaru pro reakci na vznikající hrozby v současné době. OPNsense je vybaven spolehlivým a bezpečným aktualizačním mechanismem, který poskytuje týdenní aktualizace zabezpečení. K instalaci dalších balíčků a přizpůsobení lze použít mechanismus zásuvných modulů. [88]

5.4 Kolektor dat

Pro kolektor či úložiště dat jsem zvolil Elasticsearch. Elasticsearch je distribuovaný, open-source vyhledávací a analytický nástroj postavený na Apache Lucene. Začal jako škálovatelná verze otevřeného vyhledávacího frameworku Lucene. Elasticsearch umožňuje ukládat, vyhledávat a analyzovat obrovské objemy dat velice rychle. Je schopen dosáhnout rychlých vyhledávacích odpovědí, protože místo přímého prohledávání textu prohledává index. Používá strukturu založenou na dokumentech namísto tabulek a schémat. Přichází s rozsáhlými REST API pro ukládání a vyhledávání dat. [89]

Elasticsearch v zásadě organizuje data do dokumentů, což jsou jednotky informací založené na JSON představující entity. Dokumenty jsou seskupeny do indexů, podobně jako databáze, na základě jejich charakteristik. Elasticsearch využívá invertované indexy, datovou strukturu, která mapuje slova na jejich umístění v dokumentu, pro efektivní vyhledávání. [90]

Abychom lépe porozuměli tomu, jak Elasticsearch funguje, ukažme si některé základní koncepty toho, jak organizuje data a jejich backendové komponenty.

Dokumenty

Dokumenty jsou základní jednotkou informací, které lze indexovat v Elasticsearch vyjádřené v JSON. Tento dokument si můžeme představit jako řádek v relační databázi, který představuje danou entitu – věc, kterou hledáte. V Elasticsearch může být dokument víc než jen text, může to být jakákoli strukturovaná data zakódovaná v JSON. Těmito daty mohou být věci jako čísla, řetězce a data. Každý dokument má jedinečné ID a daný datový typ, který popisuje, o jakou entitu dokumentu jde. Dokument může například představovat článek encyklopedie nebo položky protokolu z webového serveru. [90]

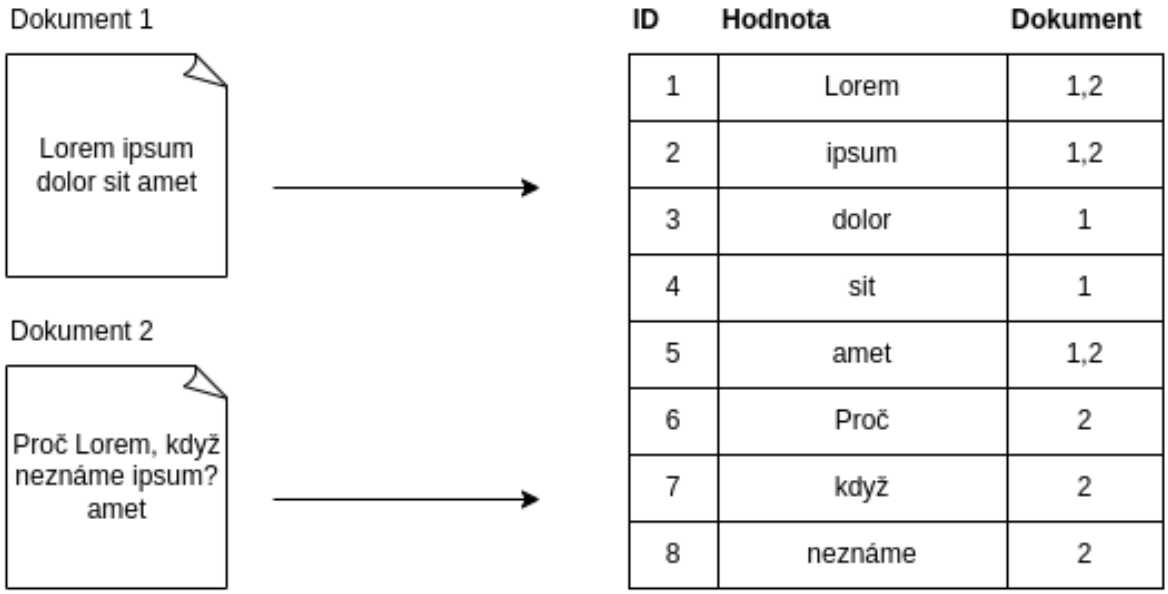
Indexy

Index je soubor dokumentů, které mají podobné vlastnosti. Index je entita nejvyšší úrovně, na kterou můžete v Elasticsearch dotazovat. Index si můžeme představit jako podobný databázi ve schématu relační databáze. Všechny dokumenty v rejstříku spolu obvykle logicky souvisí. V kontextu webové stránky elektronického obchodu můžete mít například index pro zákazníky, jeden

pro produkty, jeden pro objednávky a tak dále. Index je identifikován názvem, který se používá k odkazování na index při provádění operací indexování, vyhledávání, aktualizace a odstraňování dokumentů v něm obsažených. [90]

Invertovaný index

Index v Elasticsearch je ve skutečnosti to, co se nazývá invertovaný index, což je mechanismus, pomocí kterého fungují všechny vyhledávače. Je to datová struktura, která ukládá mapování obsahu, jako jsou slova nebo čísla, do jeho umístění v dokumentu nebo v sadě dokumentů. V podstatě se jedná o datovou strukturu podobnou hashmap, která vás nasměruje od slova k dokumentu. Invertovaný index neukládá řetězce přímo a místo toho rozděluje každý dokument na jednotlivé hledané výrazy (tj. každé slovo) a poté mapuje každý hledaný výraz na dokumenty, v nichž se tyto hledané výrazy vyskytují. Na níže uvedeném obrázku je reprezentovaný invertovaný index. Například hodnota ”Proč” se vyskytuje v dokumentu 2, takže je na tento dokument namapován. Toto slouží k rychlému vyhledání toho, kde v daném dokumentu najít hledané výrazy. Pomocí distribuovaných invertovaných indexů Elasticsearch rychle najde nejlepší shody pro fulltextové vyhledávání i z velmi rozsáhlých datových souborů. [90]



Obrázek 5.4: Invertovaný index

V naší síti budeme předpokládat středně velký provoz (např. provoz na střední škole), provedeme následující optimalizace.

- **Zrušíme refresh interval** - Operace, která spočívá ve zviditelnění změn pro vyhledávání – nazývaná obnovování – je nákladná a její časté volání během probíhající aktivity indexování může snížit rychlost indexování. Ve výchozím nastavení Elasticsearch periodicky obnovuje

indexy každou sekundu, ale pouze u indexů, které obdržely jeden nebo více požadavků na vyhledávání za posledních 30 sekund. Toto je optimální konfigurace, pokud nemáme žádný nebo velmi malý provoz z vyhledávání (např. méně než jeden požadavek na vyhledávání každých 5 minut) a chcete optimalizovat rychlost indexování. Toto chování má za cíl automaticky optimalizovat hromadné indexování ve výchozím případě, kdy nejsou prováděna žádná vyhledávání. [91]

- **Zakážeme repliky pro počáteční load** - Protože budeme zpracovávat skoro celý dataset najednou do Elasticsearch, je výhodné nastavit **index.number_of_replicas** na 0, aby se indexování urychlilo. Neexistence replik znamená, že ztráta jednoho uzlu může způsobit ztrátu dat, takže je důležité, aby data žila jinde, aby bylo možné toto počáteční načtení v případě problému zopakovat. [90]
- **Zakážeme swapping** - Většina operačních systémů se snaží využít co nejvíce paměti pro mezipaměti systému souborů a dychtivě vyměňovat nevyužitou paměť aplikací. To může vést k tomu, že části heapu (haldy) JVM nebo dokonce její spustitelné stránky budou vyměněny na disk. Swapování je velmi špatné pro výkon, stabilitu uzlů a je třeba se mu za každou cenu vyhnout. Může způsobit, že shromažďování paměti bude trvat minuty namísto milisekund a může způsobit, že uzly budou reagovat pomalu nebo se dokonce odpojí od clusteru. V odolném distribuovaném systému je efektivnější nechat operační systém zabít uzel. [92]
- **Použijeme automaticky generované ID** - Při indexování dokumentu, který má explicitní ID, musí Elasticsearch zkontrolovat, zda dokument se stejným ID již existuje ve stejném datovém fragmentu, což je nákladná operace a s rostoucím indexem je ještě dražší. Pomocí automaticky generovaných ID může Elasticsearch tuto kontrolu přeskočit, což zrychluje indexování. [91]

Pokud bychom měli k dispozici správu nad HW, mohli bychom použít rychlejší úložiště. Elasticsearch obecně vytváří jednotlivé soubory se sekvenčním zápisem. Indexování však zahrnuje zápis více souborů současně a také kombinaci náhodného a sekvenčního čtení, takže disky SSD mají tendenci fungovat lépe než klasické mechanické disky. Pokud bychom měli k dispozici pole disků, mohli bychom zvolit pole RAID 0, který rozloží index do více disků, však se tímto zvýší riziko selhání, protože selhání kteréhokoli SSD zničí index. [89]

Elasticsearch je ideální volbou pro analýzu dat protokolů a událostí. Je široce používán v IT operacích a DevOps ke sledování stavu systému, odstraňování problémů a identifikaci anomálií.

Elastick Stack

V krátkosti se ještě zmíním o Elastic Stacku. Elastic search je součástí tohoto eko-systému nazývaného „ELK“, pojmenovaného podle jeho komponent - Elastic search, Logstash a Kibana.

- **Logstash** - používá se k agregaci a zpracování dat (transformuje a připravuje data bez ohledu

na formát tím, že identifikuje pojmenovaná pole pro vytvoření struktury a transformuje je tak, aby konvergovala na společném formátu) a jejich odeslání do Elasticsearch. [93]

- **Kibana** - Nástroj pro vizualizaci a správu dat pro Elasticsearch, který poskytuje histogramy, spojnicové grafy, koláčové grafy a mapy v reálném čase.

Tyto komponenty jsou velice dobře integrovatelné mezi sebou, však pro naše účely využijeme pouze Elastic search a Kibanu. Důvodem vynechání Logstash je, že pro naše účely není zcela nutný, data budeme kolektovat pouze z jednoho zdroje, tím pádem transformace nejsou potřeba a hrála by zde pouze zbytečnou režii. Pokud bychom měli více zdrojů dat, Logstash by byl využit. [93]

5.5 Analýza dat a možnosti reportingu

V rámci generování náhodného síťového provozu byl nashromážděn dataset obsahující přes 7 mil. záznamů během dvou měsíců generování náhodného provozu. V praxi se analýza síťového provozu provádí nad určitým časovým intervalem, primárně v rámci řešení bezpečnostního problému. Výsledná implementace umožňuje i vizualizaci nad celým datasetem. Protože byl síťový provoz náhodně vygenerován, představuji zde obecné řešení možnosti analýzy síťového provozu - představení dostupných nástrojů, nikoliv řešení konkrétního případu.

Administrátorské prostředí

OPNsense je již robustní firewall, s použitím pluginu ZenArmor posouvá možnosti síťové analýzy, reportingu a zabezpečení na další úroveň NGFW. Začleněním inteligence o hrozbách v reálném čase a pokročilých mechanismů detekce a prevence narušení ZenArmor výrazně zlepšuje schopnost OPNsense detekovat a zmírňovat vznikající hrozby. Toto nám umožňuje v administrátorském prostředí OPNsense následující možnosti reportingu či analýzy: [94]

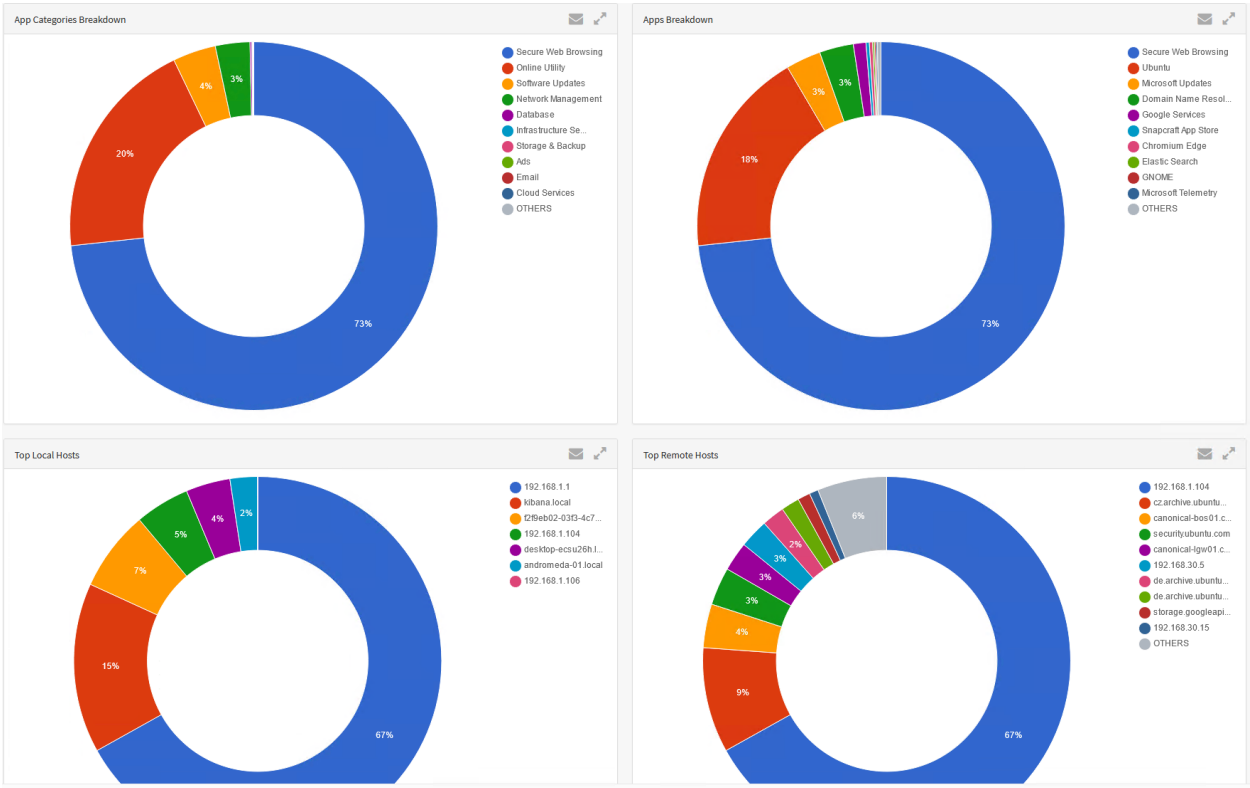
Monitorování síťového provozu v reálném čase

ZenArmor poskytuje monitorování síťového provozu v reálném čase (Live Sessions Explorer), který umožňuje zobrazit nejnovější připojení, bloky, webové relace, požadavky DNS a relace TLS.

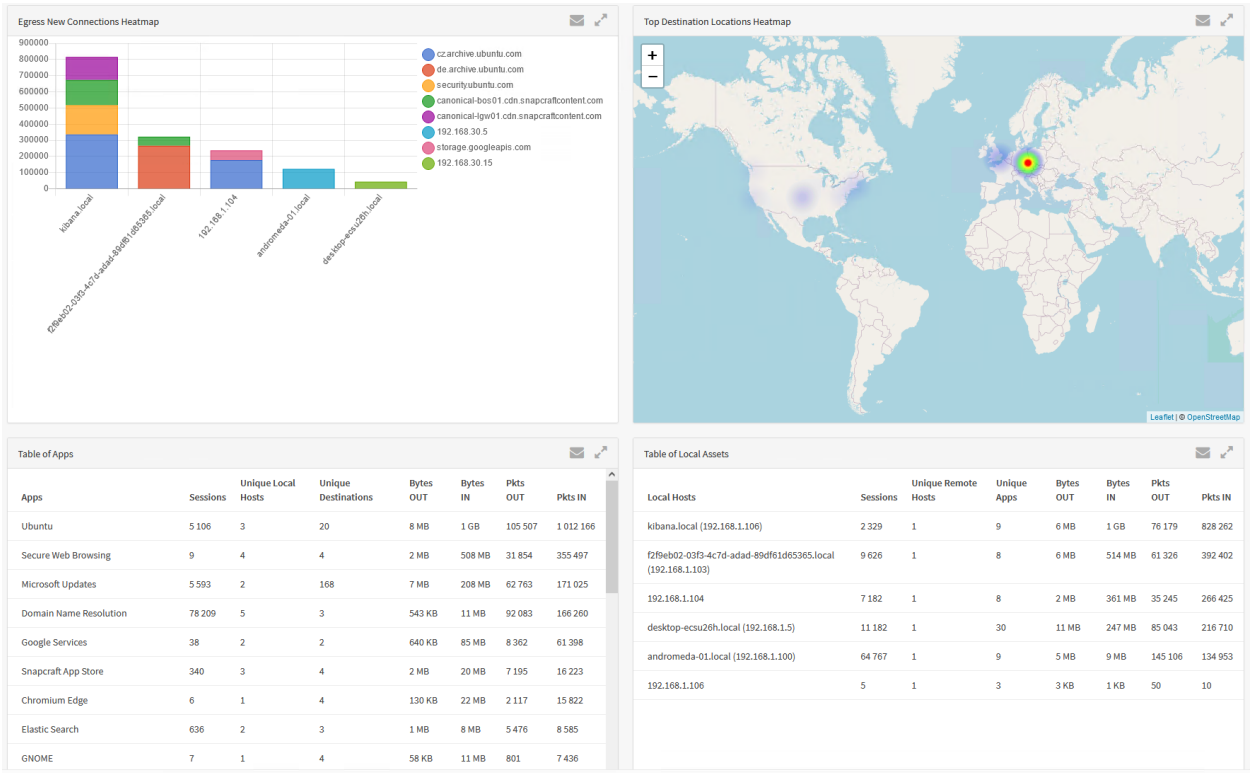
Obrázek 5.5: Monitorování síťového provozu v reálném čase

Dashboard a Reporting

Záložka „Připojení“ nám zobrazuje různé aplikace síti, které navazují interní i externí připojení. Tato připojení mohou mít jakýkoli protokol, nikoliv pouze provoz HTTP/HTTPS. Tato karta zobrazuje kategorie aplikací, protokol použitý pro připojení, dobu trvání připojení, připojené klienty, heatmapu konečných destinací a mnoho dalších.

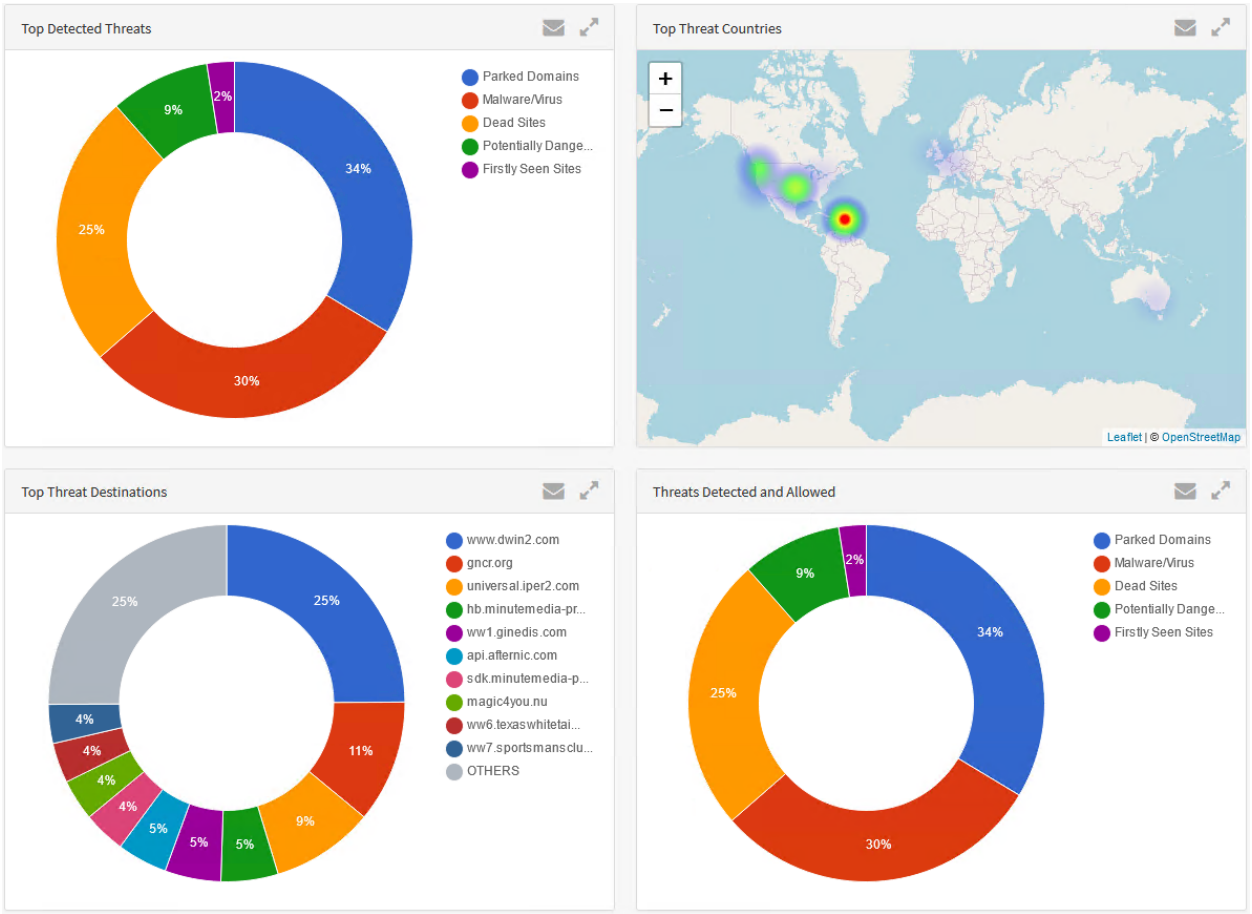


Obrázek 5.6: Připojení - přehled kategorie aplikací, aplikace využívající síť, připojení klienti a konečné destinace

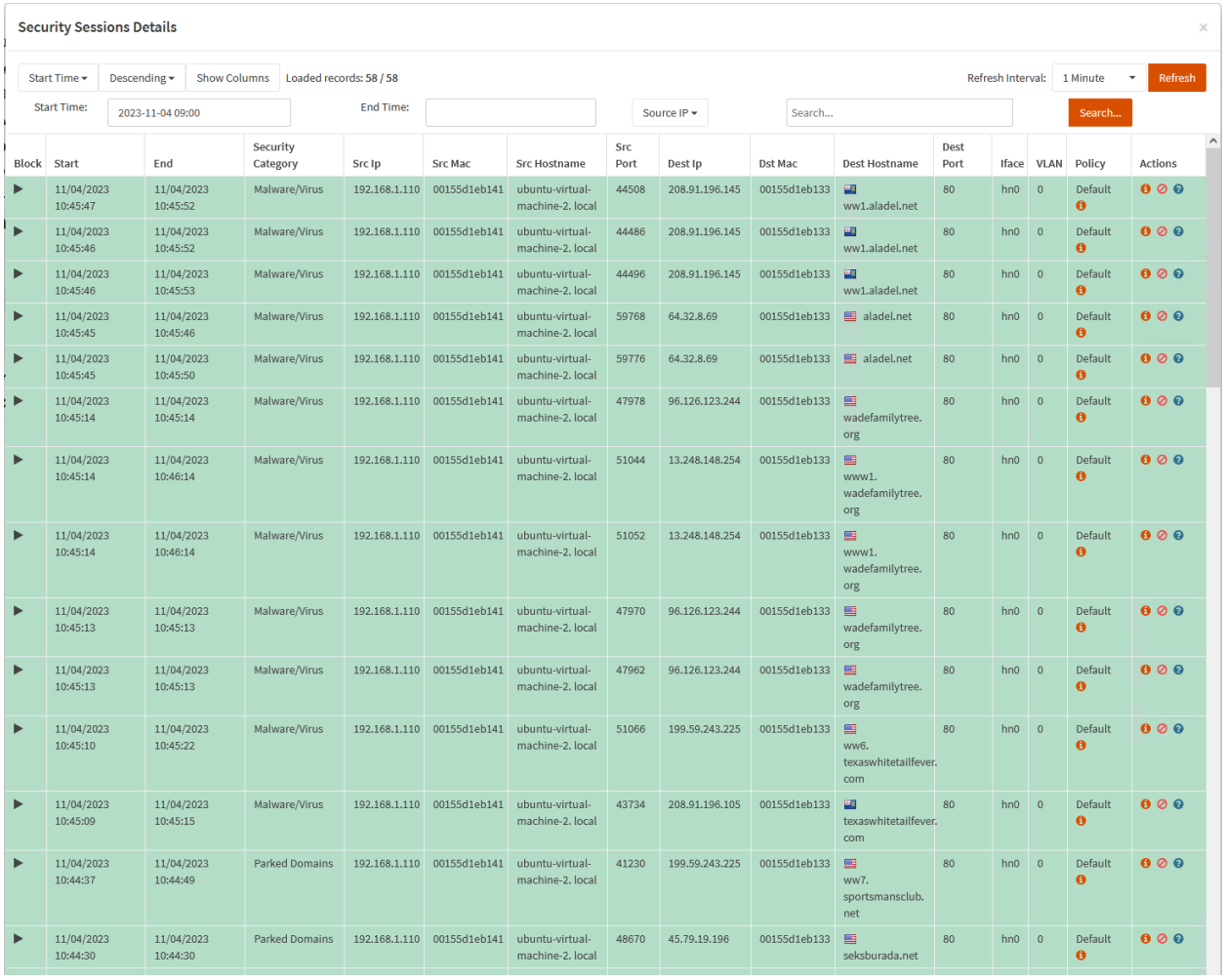


Obrázek 5.7: Připojení - mapa výstupních nových připojení, mapa cílových umístění vizualizovaných do mapových podkladů, tabulka statistik o aplikacích využívající síť a tabulka zařízení na síti a jejich statistiky

Záložka „Hrozby“ zobrazuje vše, co bylo zablokováno na základě nastavení zabezpečení definovaného zásadami. Hlášení jsou rozdělena podle různých kategorií: malware, phishing, potenciálně nebezpečné stránky a jiné. Záložka „Blocks“ zobrazuje vše, co bylo zablokováno na základě zásad ovládání webu/aplikace.



Obrázek 5.8: Hrozby - nejčastěji zjištěné hrozby, země s nejvyššími hrozbami, destinace nejčastějších hrozeb a zjištěné a povolené hrozby



Security Sessions Details

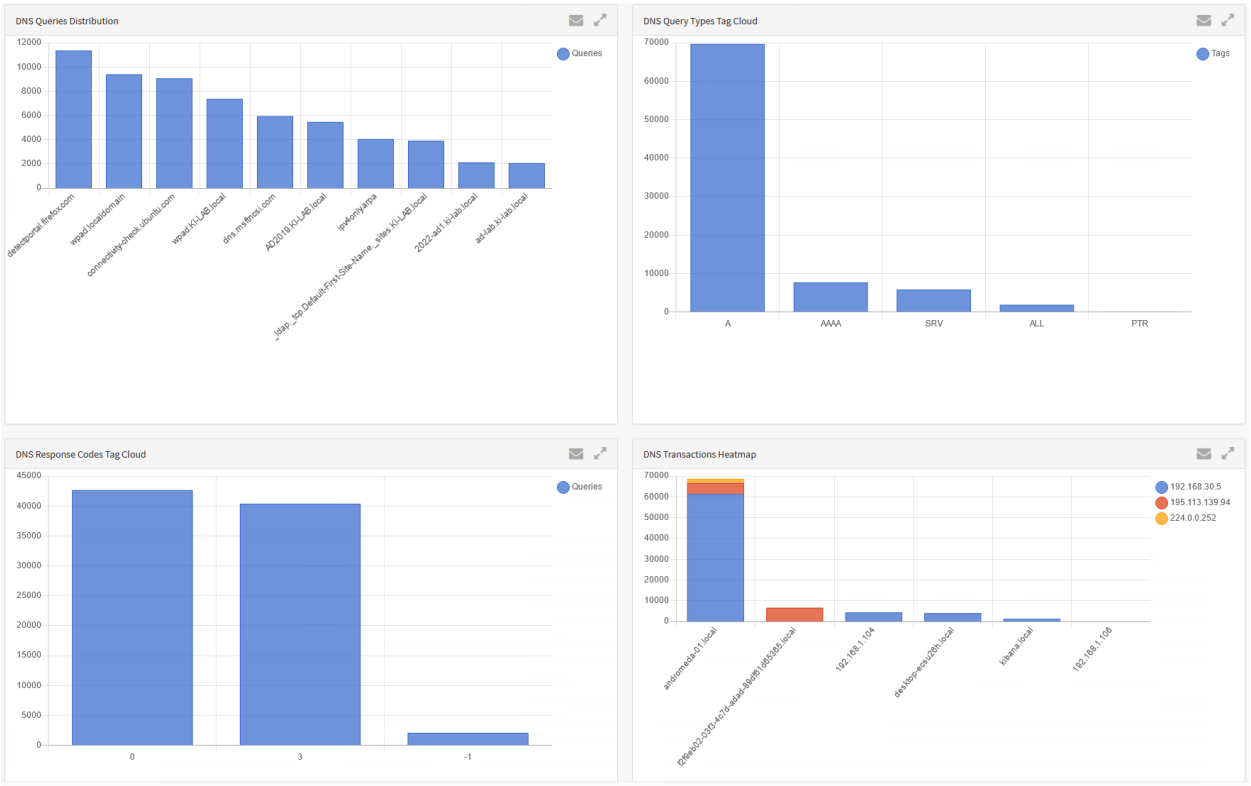
Start Time ▾ | Descending ▾ | Show Columns | Loaded records: 58 / 58 | Refresh Interval: 1 Minute | Refresh

Start Time: 2023-11-04 09:00 | End Time: | Source IP ▾ | Search... | Search...

Block	Start	End	Security Category	Src Ip	Src Mac	Src Hostname	Src Port	Dest Ip	Dst Mac	Dest Hostname	Dest Port	Iface	VLAN	Policy	Actions
▶	11/04/2023 10:45:47	11/04/2023 10:45:52	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	44508	208.91.196.145	00155d1eb133	ww1.aladel.net	80	hn0	0	Default	
▶	11/04/2023 10:45:46	11/04/2023 10:45:52	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	44486	208.91.196.145	00155d1eb133	ww1.aladel.net	80	hn0	0	Default	
▶	11/04/2023 10:45:46	11/04/2023 10:45:53	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	44496	208.91.196.145	00155d1eb133	ww1.aladel.net	80	hn0	0	Default	
▶	11/04/2023 10:45:45	11/04/2023 10:45:46	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	59768	64.32.8.69	00155d1eb133	aladel.net	80	hn0	0	Default	
▶	11/04/2023 10:45:45	11/04/2023 10:45:50	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	59776	64.32.8.69	00155d1eb133	aladel.net	80	hn0	0	Default	
▶	11/04/2023 10:45:14	11/04/2023 10:45:14	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	47978	96.126.123.244	00155d1eb133	wadefamilytree. org	80	hn0	0	Default	
▶	11/04/2023 10:45:14	11/04/2023 10:46:14	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	51044	13.248.148.254	00155d1eb133	www1. wadefamilytree. org	80	hn0	0	Default	
▶	11/04/2023 10:45:14	11/04/2023 10:46:14	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	51052	13.248.148.254	00155d1eb133	www1. wadefamilytree. org	80	hn0	0	Default	
▶	11/04/2023 10:45:13	11/04/2023 10:45:13	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	47970	96.126.123.244	00155d1eb133	wadefamilytree. org	80	hn0	0	Default	
▶	11/04/2023 10:45:13	11/04/2023 10:45:13	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	47962	96.126.123.244	00155d1eb133	wadefamilytree. org	80	hn0	0	Default	
▶	11/04/2023 10:45:10	11/04/2023 10:45:22	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	51066	199.59.243.225	00155d1eb133	ww6. texaswhitetailfever. com	80	hn0	0	Default	
▶	11/04/2023 10:45:09	11/04/2023 10:45:15	Malware/Virus	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	43734	208.91.196.105	00155d1eb133	texaswhitetailfever. com	80	hn0	0	Default	
▶	11/04/2023 10:44:37	11/04/2023 10:44:49	Parked Domains	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	41230	199.59.243.225	00155d1eb133	ww7. sportsmansclub. net	80	hn0	0	Default	
▶	11/04/2023 10:44:30	11/04/2023 10:44:30	Parked Domains	192.168.1.110	00155d1eb141	ubuntu-virtual-machine-2. local	48670	45.79.19.196	00155d1eb133	seksburada.net	80	hn0	0	Default	

Obrázek 5.9: Monitorování hrozeb v síti v reálném čase

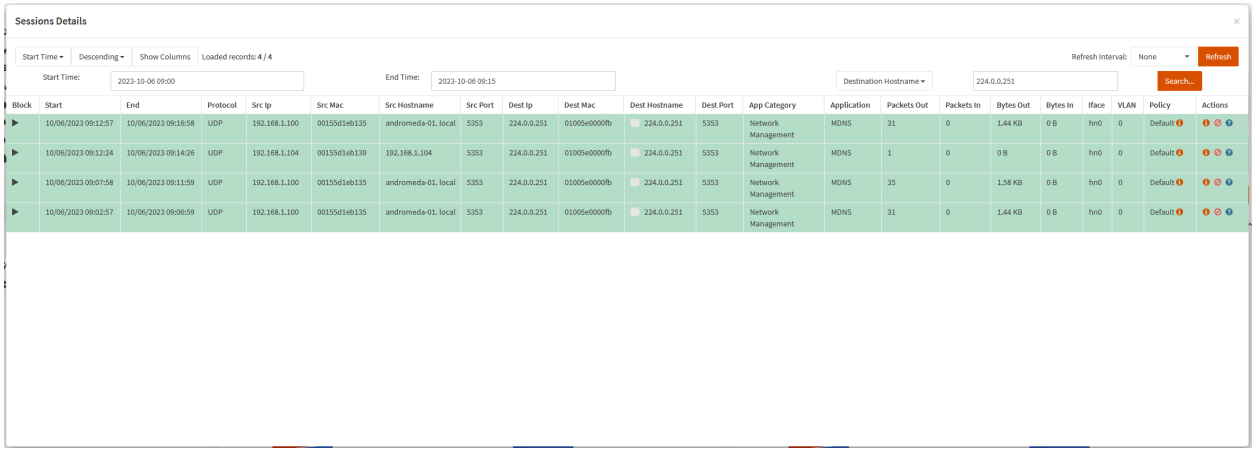
Záložka „DNS“ zobrazuje informace o nejčastěji prováděných DNS požadavcích a další informace související s DNS. Tyto přehledy jsou užitečné při hledání zařízení, která provádějí velké množství požadavků DNS.



Obrázek 5.10: DNS - distribuce dns dotazů, typy dotazů DNS Tag Cloud, DNS odpovědi Tag Cloud a mapa transakcí DNS

Záložka „TLS“ zobrazuje informace o relaci TLS, jako jsou hostitelé/IP adresy, ve kterých je vytvořeno nejvíce relací TLS, použité porty, obecné kategorie relací a další informace. Osobně nepovažuji informace TLS za tak užitečné jako ostatní přehledy, protože se zdá, že většinu těchto informací najdete v jiných přehledech. Mohou však existovat případy, kdy může být užitečné zobrazit informace specifické pro TLS.

Nad veškerými výše uvedenými možnostmi jsme schopni provést detailní filtraci dat. Na níže uvedeném příkladu si můžeme ukázat filtraci dat. Mějme situaci, kdy potřebujeme zjistit, kdo se v určitém časovém intervalu připojoval k IP adrese 224.0.0.251. Filtrů je možné aplikovat více



Obrázek 5.11: Filtrace dat

Veškeré tyto statistiky je možné exportovat do dokumentového formátu.

Politika zabezpečení

ZenArmor umožňuje povolit blokování různých typů aktivity malwaru a potenciálně nebezpečných webových stránek.

Malware filter

Povolením této možnosti se blokují weby, o kterých je známo, že obsahují malware. Webové stránky infikované malwarem mohou rychle ohrozit zařízení a sítě, což vede k narušení dat, zranitelnosti systému a neoprávněnému přístupu. Jedná se o proaktivní obranou proti těmto zákeřným hrozbám. Odříznutím přístupu ke známým webům hostujícím malware minimalizujete riziko nechtěného vystavení sítě škodlivému kódu. [95]

Phishingové servery

Povolení této možnosti umožní síti blokovat phishingové útoky. Phishingové útoky často využívají škodlivý software hostovaný na konkrétních serverech. Tato funkce je strategickou obranou proti podvodným taktikám. Narušuje infrastrukturu útočníků a brání uživatelům v interakci s doménami, které by mohly ohrozit citlivá data nebo přihlašovací údaje. [95]

Spam filter

Chraní síť před záplavou nechtěného a potenciálně škodlivého obsahu se spamem. Tato funkce brání přístupu na webové stránky před šířením spamu, což může vést k přeplněné schránce, plýtvání zdroji a potenciálním bezpečnostním rizikům. Odfiltrováním spamových stránek nejen zlepší provozní efektivitu, ale také sníží vystavení potenciálním pokusům o phishing nebo malwaru. [95]

Hackerské stránky

Toto proaktivní opatření brání přístupu na webové stránky, o kterých je známo, že šíří obsah související s hackingem. Takové stránky často poskytují zdroje a nástroje, které usnadňují kyberzločinecké aktivity. Zablokováním přístupu na tyto stránky je omezeno riziko nedopatřeného získají nástroje nebo informace, které by mohly být zneužity ke škodlivým účelům. [95]

Potenciálně nebezpečné stránky

Tato možnost blokuje potenciálně nebezpečné stránky. Dané stránky nemusí být 100% škodlivé, nicméně mohou obsahovat podezřelé aktivity či příznaky, které připomínají škodlivé stránky.

Tato funkce umožňuje preventivně zamezit přístupu na stránky s podezřelými aktivitami, které odrážejí chování škodlivých webových stránek. Jedná se o klíčový aspekt proaktivní bezpečnostní strategie ZenArmoru. Zablokováním přístupu na stránky vykazující podezřelé vlastnosti se sníží riziko ohrožení uživatelů na síti. [95]

Napadené webové stránky

Napadené webové stránky jsou webové stránky, které byly infiltrovány nebo hacknuty neoprávněnými subjekty se zlými úmysly. Napadené webové stránky mohou sloužit jako vektory pro šíření malwaru, což vede k potenciálnímu narušení dat a narušení systému. ZenArmor využívá mechanismy detekce hrozeb k identifikaci a blokování přístupu k napadeným webům. Tento proaktivní přístup zabraňuje uživatelům neúmyslně navštěvovat webové stránky, které byly infiltrovány nebo hacknuty škodlivými aktéry. [95]

Keyloggery a monitorování

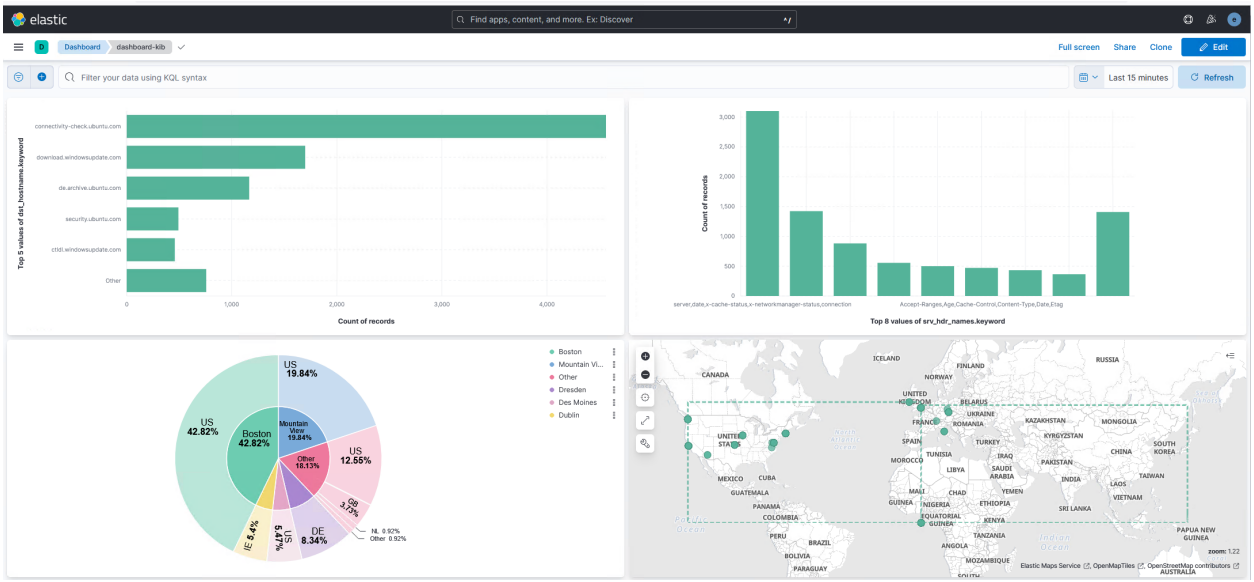
Tato ochrana se vztahuje na všechna zařízení a uživatele a poskytuje komplexní obranu proti těmto invazivním hrozbám. Keyloggery a monitorovací software jsou programy, které skrytě zaznamenávají stisknuté klávesy uživatele nebo sledují návyky při procházení webu, často bez vědomí nebo souhlasu uživatele. ZenArmor aktivně detekuje a blokuje keyloggery a monitorovací software pokoušející se proniknout do sítě. Tato proaktivní obrana zabraňuje neoprávněnému přístupu k citlivým údajům, jako jsou přihlašovací údaje a soukromé informace. [95]

Cloud Threat Intelligence

Cloud Threat Intelligence lze definovat jako proaktivní přístup k identifikaci, analýze a zmírňování bezpečnostních hrozeb, které se zaměřují na cloudovou infrastrukturu, platformy a aplikace. Zahrnuje shromažďování a analýzu obrovského množství dat z různých zdrojů, jako jsou protokoly, síťový provoz a informační kanály o hrozbách, za účelem zjišťování hrozeb a reakce na ně v reálném čase. ZenArmor, přední poskytovatel cloudového zabezpečení, integroval do svých nabídek Cloud Threat Intelligence, což umožňuje posílit jejich bezpečnostní pozici v cloudu. [95]

5.6 Kibana

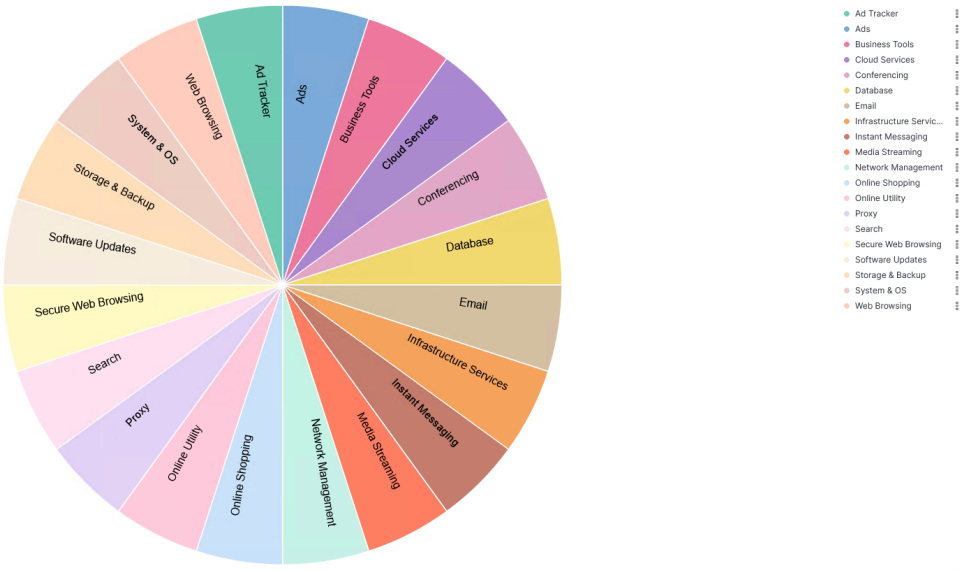
Kibana je open-source nástroj pro vizualizaci a průzkum dat. Kibana je zde implementována primárně pro síťové analytiky ke zkoumání dlouhodobých dat. Instance Kibany běží na vlastním serveru, čímž neohrozí provoz OPNsense či Elasticsearch při náročných dotazech. Kibana čerpá z Elasticsearch úložiště, tedy jsou k dispozici veškerá data. Pro vytvoření statistik můžeme použít Kibana Lens. Lens je vestavěná funkce pro vytváření grafů.



Obrázek 5.12: Dashboard v kibaně

Níže si uvedeme příklad denní práce síťového analytika. Je vyžadováno zjistit na jaké zahraniční weby, ideálně s uvedeným městem, se streamovacím obsahem a weby s reklamami se zařízení v naší síti připojily v určitý den. Zadavatel také vyžaduje grafy distribuce provozu dle cílových zemí včetně měst.

V rámci celého příkladu budeme předpokládat, že veškeré vizualizace provádíme nad datech se zafixovaný dnem. V první řade musíme nejdříve zjistit, jak se provoz klasifikuje. Tyto informace můžeme zjistit z aplikační kategorie, kterou OPNSense přidává ke každému požadavku. Vizualizujeme si unikátní hodnoty aplikačních kategorií.

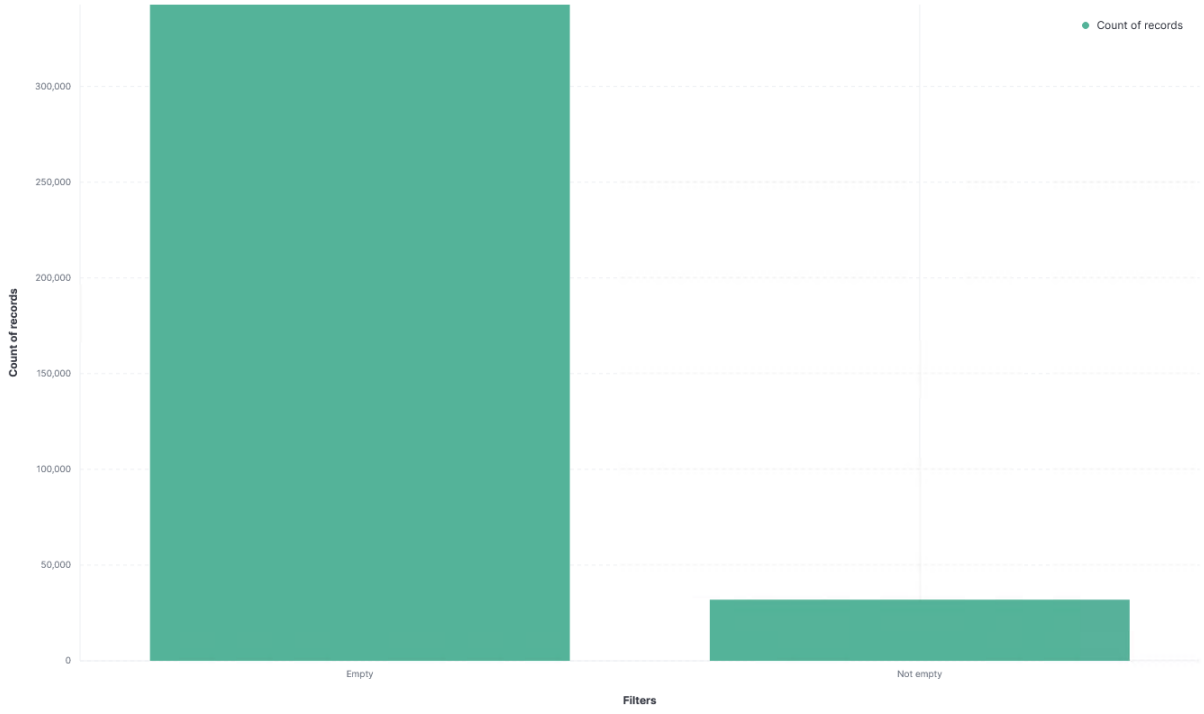


Obrázek 5.13: Aplikační kategorie (koláčový graf)

Pro lepší přehled si tento graf můžeme vizualizovat do stromové mapy.

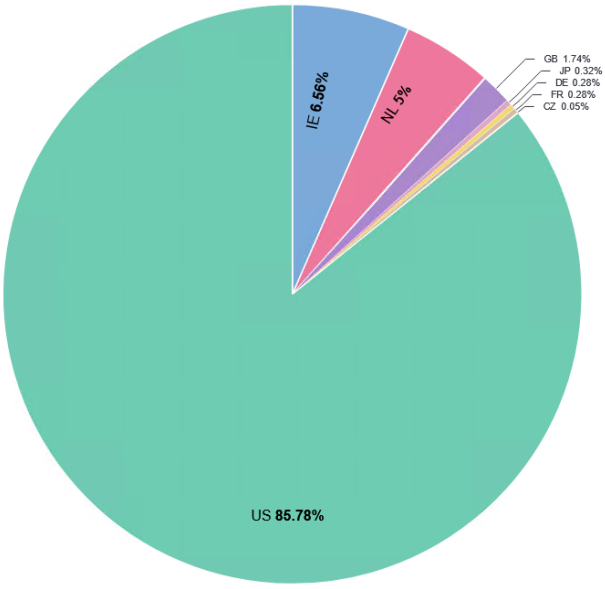
Obrázek 5.14: Aplikační kategorie (stromová mapa)

Pro náš případ jsou důležité kategorie Ads, Ad Tracker a Media Streaming. Následujeme pátráním po geolokaci. Zajímá nás pouze provoz, který má cílovou GeoIP země a města. Ještě než se pustíme do daného pátrání, podívejme se na menší porovnání provozu s prázdnou a vyplněnou cílovou GeoIP.



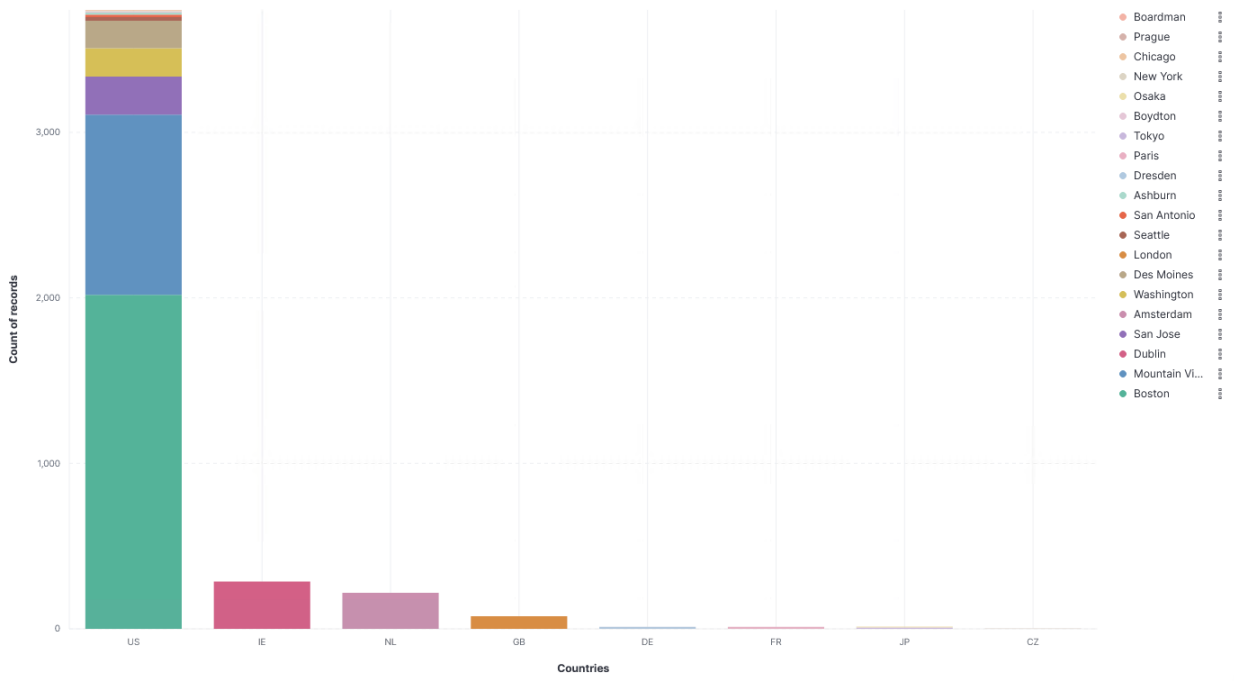
Obrázek 5.15: Záznamy s prázdnou vs vyplněnou cílovou GeoIP země

Na daném grafu vidíme, že většina provozu je bez cílové GeoIP země. Pojďmě se nyní podívat na distribuci cílových zemí celého provozu. K zafixovanému filtru na určité datum přidejme nový filtr na neprázdné cílové GeoIP zemí.



Obrázek 5.16: Distribuce záznmů dle cílové GeoIP země

Tento graf rozšíříme a přidáme do něj informace o cílové GeoIP města.

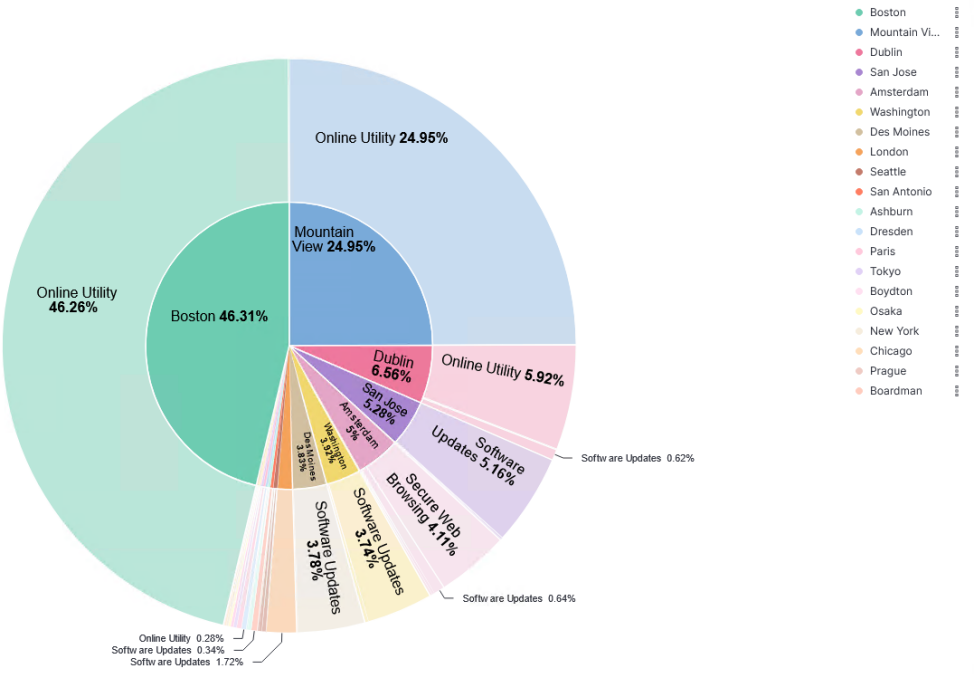


Obrázek 5.17: Distribuce záznamů dle cílové GeoIP země a města

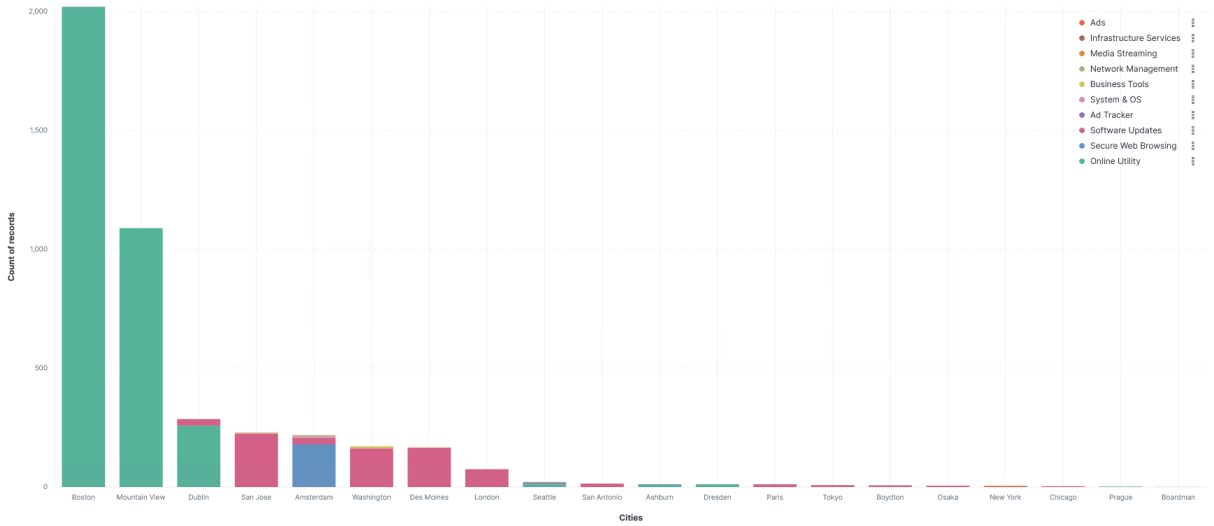
Pro větší přehled můžeme předchozí graf vizualizovat pomocí mapových podkladů. Zadavatel má nyní několik grafů vizualizujících distribuci síťového provozu dle cílové země a měst.

Obrázek 5.18: Distribuce záznamů dle cílové GeoIP země a města (mapa)

Zadavateli připravíme i grafy zobrazující distribuci provozu dle cílové GeoIP města a aplikačních kategorií v koláčovém grafu a stromové mapě.

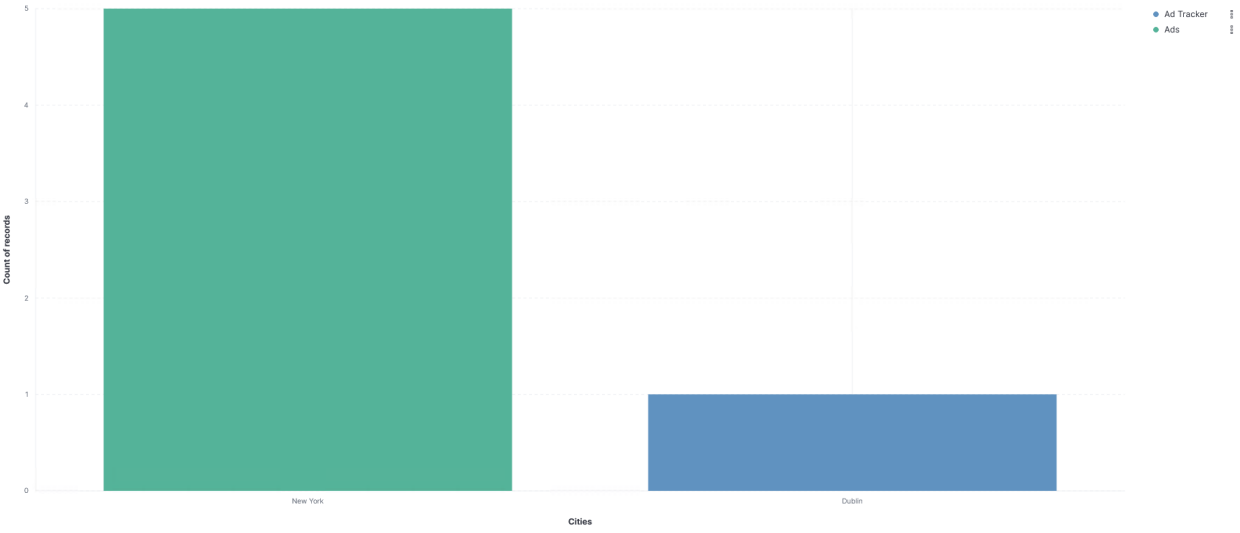


Obrázek 5.19: Distribuce záznamů dle cílové GeoIP města a aplkikační kategorie (koláčový graf)



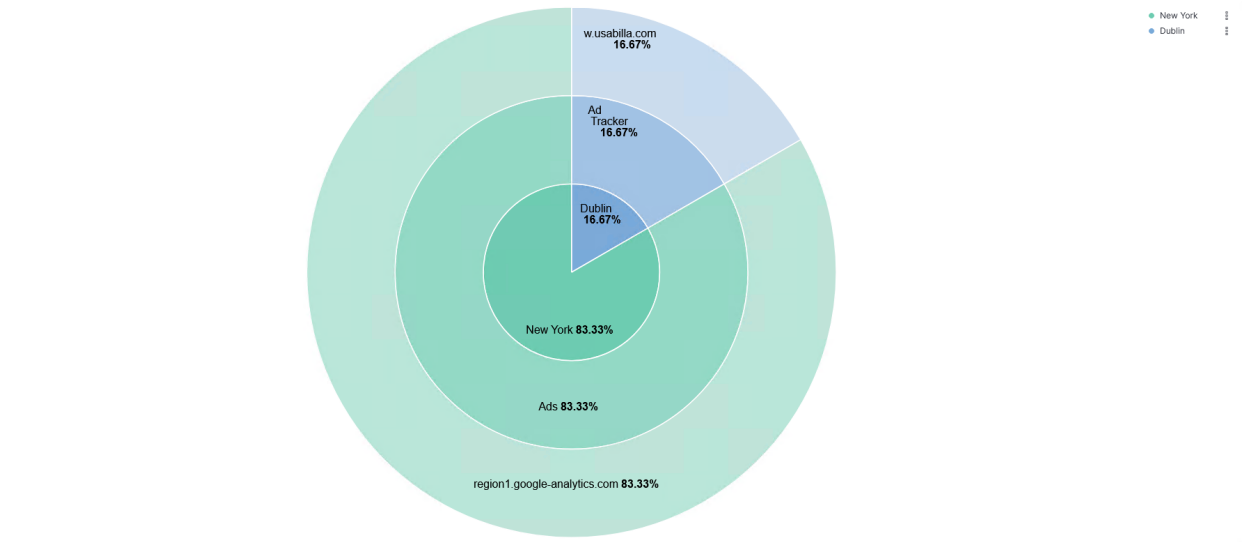
Obrázek 5.20: Distribuce záznamů dle cílové GeoIP města a aplkikační kategorie (stromová mapa)

Nyní se můžeme zaměřit na zjišťování reklamních webů a webů se streamovacím obsahem. Nejprve prověříme reklamní weby.



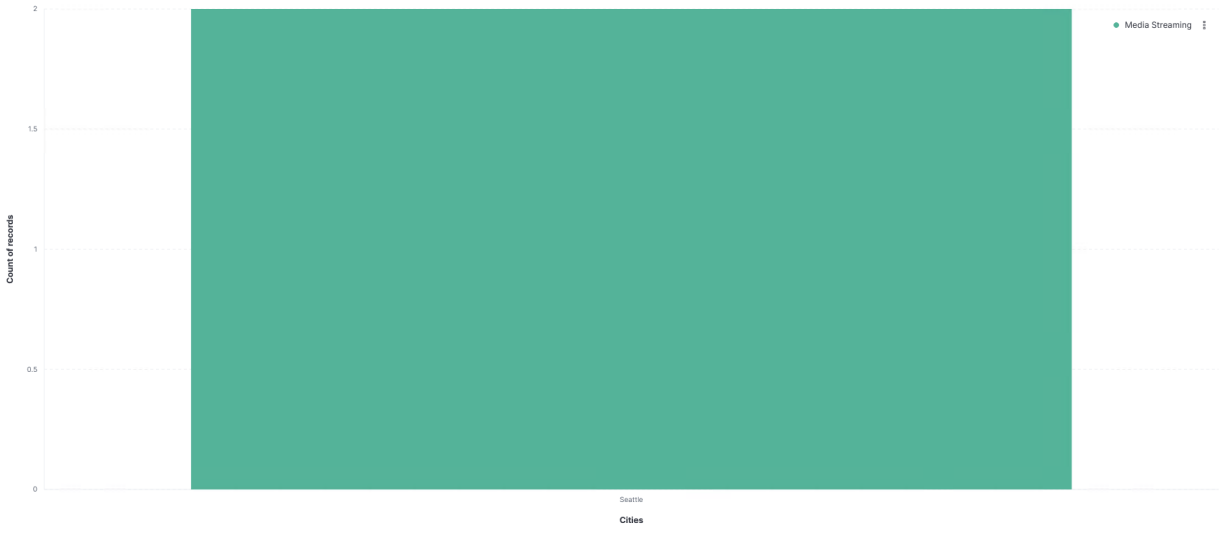
Obrázek 5.21: Cílové města s reklamním síťovým provozem

V grafu vidíme, že provoz vedl pouze do měst New York a Dublin.



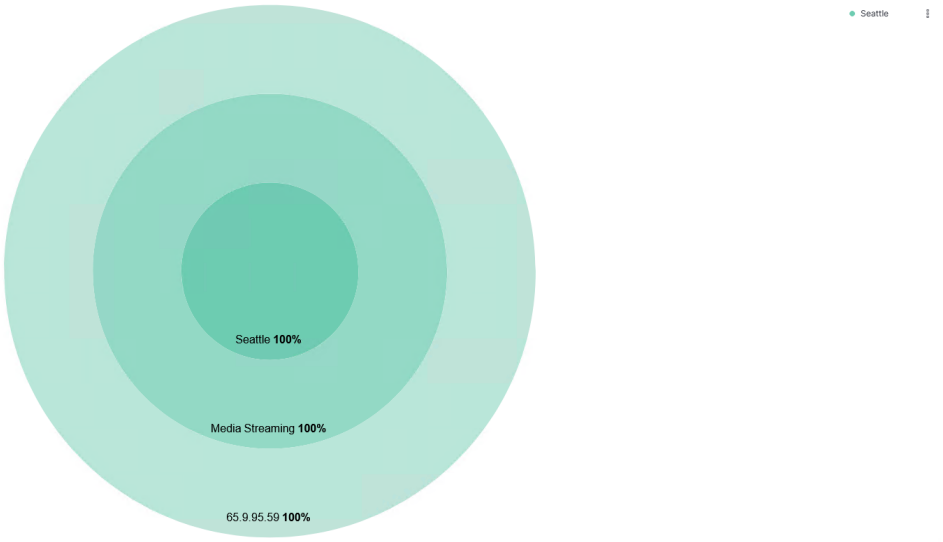
Obrázek 5.22: Cílové města s reklamním síťovým provozem včetně webů

Zbýva nám ještě zjistit adresa se streamovacím obsahem.



Obrázek 5.23: Cílové města s streamovacím síťovým provozem

V tomto případě se jedná pouze o město Seattle.

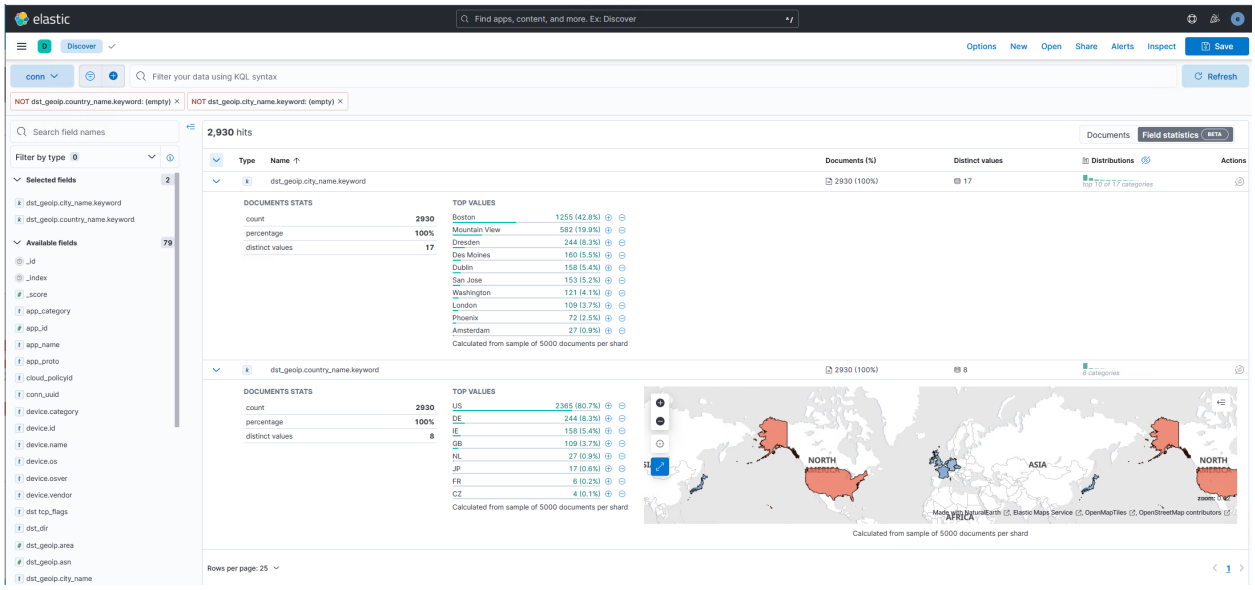


Obrázek 5.24: Cílové města s streamovacím síťovým provozem včetně webů

Výsledkem tohoto zjišťování je připravený dashboard s grafy pro zadavatele a adresy w.usabilla.com, region1.google-analytics.com a IP adresa 65.9.95.59.

Cílem tohoto zjišťování bylo ukázat sílu Kibana Lens. Od základních sloupcových grafů až po složité geoprostorové reprezentace, Lens nabízí rozmanitou škálu typů vizualizace, které uspokojí různé analytické potřeby. Tato všestrannost zajišťuje, že si uživatelé mohou vybrat nejvhodnější metodu vizualizace pro jejich konkrétní případ, což zvyšuje jasnost a dopad jejich postřehů.

Pokud bychom chtěli hlouběji zkoumat data, můžeme využít modul Discover. Kibana Discover je speciálně navržen tak, aby pomohl prozkoumat a prohledávat data uložená v Elasticsearch interaktivním a uživatelsky přívětivým způsobem nebo za pomoci vlastního dotazovacího jazyka KQL. Na níže uvedeném obrázku je zobrazen příklad zkoumání, kam se klienti připojují (stát a město).



Obrázek 5.25: Kibana Discover

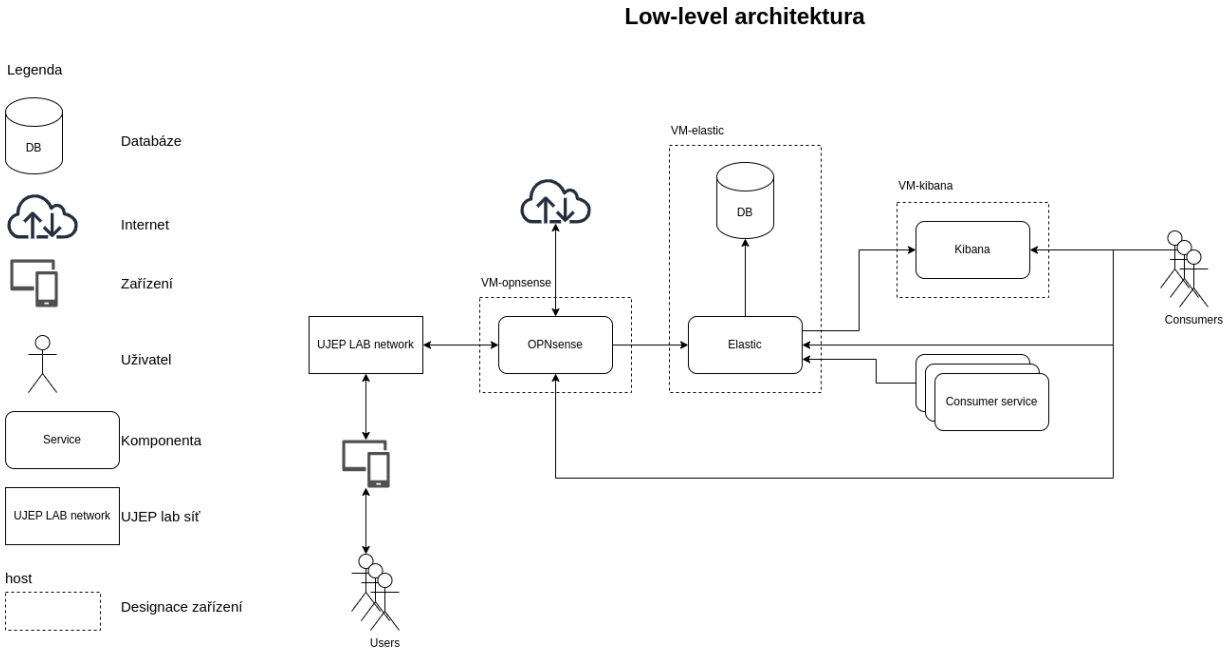
5.7 API

Elasticsearch poskytuje výkonné a flexibilní API, které umožňuje pracovat s daty uloženými v indexech Elasticsearch a manipulovat s nimi. Elasticsearch API je RESTful API, což znamená, že se řídí principy podobnými protokolu HTTP. To znamená, že k interakci se systémem používá standardní metody HTTP (GET, POST, PUT, DELETE) a adresy URL. Každý požadavek API obvykle odpovídá konkrétní akci nebo operaci. Rozhraní API poskytuje podrobné chybové reakce se stavovými kódy a chybovými zprávami, což usnadňuje diagnostiku a odstraňování problémů. [96]

Možnosti provádění analýzy, reportingu či exportu dat máme v této implementaci velké. V praxi bychom přehledy v OPNsense zpřístupnili pouze osobám s nejvyšším oprávněním jako jsou systémoví administrátoři. Pro ostatní uživatele, ať už analytiky či bežné uživatele, bychom využili Kibanu, např. pro potřeby pravidelného reportingu.

5.8 Low-level architektura

V závěru se podívejme na low-level architekturu řešení. Low-level architektura (LLA) či design (LLD) rozšiřuje high-level architekturu. V tomto pohledu již jsou zanesené skutečné komponenty výsledného řešení. [97]



Obrázek 5.26: Low-level architektura

Za zmínku stojí komponenta Consumer service, zde je koncovému uživateli nabídnuta možnost konzumovat hrubá data přímo z Elastic search. Daný uživatel, rozhraní či služba může tato data zpracovávat dle své libosti.

5.9 Shrnutí vlastního řešení a porovnání s představeným SW

V dnešním digitálním prostředí má zabezpečení sítě prvořadý význam, zejména pro podniky a organizace, které při svých každodenních operacích spoléhají na internet. Vzhledem k tomu, že se kybernetické hrozby neustále vyvíjejí, je zásadní mít k dispozici robustní a flexibilní bezpečnostní řešení. OPNsense a ZenArmor jsou taková řešení, která si získala oblibu pro svou schopnost zvýšit zabezpečení sítě prostřednictvím jednotného přístupu.

ZenArmor doplňuje OPNsense tím, že poskytuje komplexní řešení kybernetické bezpečnosti, které zvyšuje zabezpečení sítě. Je navržen tak, aby čelil moderním výzvám, jako jsou útoky zero-day, pokročilé perzistentní hrozby a zvyšující se sofistikovanost kyberzločinců. ZenArmor jde nad rámec tradičních firewallů a systémů detekce narušení a nabízí pokročilé možnosti detekce hrozeb a reakce. Srdcem ZenArmoru je jeho platforma pro informace o hrozbách, která nepřetržitě monitoruje síťový provoz a identifikuje podezřelé aktivity a anomálie. Prostřednictvím strojového učení a algoritmů umělé inteligence dokáže ZenArmor detekovat vzorce a chování svědčící o kybernetických hrozbách. Tento proaktivní přístup umožňuje organizacím reagovat na potenciální hrozby dříve, než přerostou v závažné bezpečnostní incidenty.

Pro maximalizaci potenciálu OPNsense a ZenArmor je integrace Elasticsearch jako analytického nástroje strategickou volbou. Elasticsearch je open source, distribuovaný, RESTful vyhledávací a analytický nástroj, známý svou škálovatelností a schopnostmi analýzy dat v reálném čase. Využitím Elasticsearch můžeme získat mocný přehled ze síťových dat. Elasticsearch poskytuje vylepšenou analytickou vrstvu, která umožňuje korelovat a analyzovat síťová data, detekovat neobvyklé vzorce a přesně reagovat na bezpečnostní incidenty. Jeho škálovatelnost a robustní možnosti dotazování z něj činí ideálního společníka pro tato bezpečnostní řešení.

V porovnání s představenými softwary, Wireshark a vlastní řešení jsou dva výkonné nástroje, které slouží odlišným účelům v oblasti správy a zabezpečení sítě. I když sdílejí některé společné rysy, zásadně se liší ve svých aplikacích a schopnostech, primárně v analýze protokolů ve kterém Wireshark vyniká. Na druhou stranu, vlastní řešení je bezpečnostní řešení navržené k ochraně sítí proti široké škále kybernetických hrozeb. Na rozdíl od Wireshark je OPNsense se ZenArmor zaměřen na aktivní ochranu sítě předcházením a zmírňováním bezpečnostních hrozeb v reálném čase.

Vlastní řešení je bližší ostatním představeným softwarům. Ve výsledku „nejlepší“ řešení monitorování sítě závisí na konkrétních potřebách a preferencích. SolarWinds NetFlow Traffic Analyzer je vynikající volbou pro ty, kteří oceňují snadné použití a širokou škálu podporovaných technologií. Nagios Network Analyzer je vhodnou volbou pro ty, kteří již jsou v ekosystému Nagios. ntop je ideální pro nadšence hloubkové kontroly paketů a analýzy síťového provozu. Vlastní řešení poskytuje komplexní platformu pro zabezpečení a monitorování, ideální pro ty, kteří hledají integrované a odolné řešení.

Každé z těchto řešení nabízí jedinečný přístup k monitorování sítě a nejlepší volba závisí na požadavcích uživatele.

6 Závěr

Závěrem lze říci, že cesta analýzy síťových dat v kontextu kybernetické bezpečnosti byla poučná i náročná. Tato práce se pustila do detailního teoretického poznání analýzy síťového provozu, firewallů a poslání navrhnout a implementovat vlastní infrastrukturu věnovanou analýze síťových dat. Středobodem této infrastruktury byl Next Generation Firewall, který hrál klíčovou roli při zajišťování bezpečnosti sítě a umožňoval sofistikovanou analýzu dat.

Naším primárním cílem bylo vyvinout funkční infrastrukturu schopnou komplexně analyzovat síťový provoz, identifikovat potenciální hrozby a poskytovat v reálném čase přehled o chování sítě. V tomto úsilí se integrovala řada nástrojů pro vytvoření vlastního řešení, která efektivně řeší tyto cíle.

Mezi základní prvky našeho vlastního řešení patří OPNsense, ZenArmor, Elasticsearch a Kibana. Základ naší infrastruktury tvořil OPNsense, platforma firewallu a směrovače s otevřeným zdrojovým kódem, který poskytoval robustní bezpečnostní perimetr. ZenArmor, bezpečnostní rozšíření pro OPNsense, zlepšilo naši schopnost efektivně detekovat a zmírňovat hrozby.

Elasticsearch a Kibana sloužily jako sada nástrojů pro analýzu dat a vizualizaci. Elasticsearch, výkonný vyhledávací a analytický nástroj, umožnil dotazování na obrovské množství síťových dat. Kibana se svým intuitivním řídicím panelem a možnostmi vizualizace umožnila vytvářet působivé datové sestavy.

Porovnání vlastního řešení s dostupnými komerčními alternativami bylo zásadním aspektem práce. I když komerční řešení nabízejí pohodlí a podporu, často jsou za ně vysoké náklady. Vlastní řešení poskytlo cenově výhodnou alternativu bez kompromisů ve výkonu nebo funkčnosti. Ukázalo se, že nástroje s otevřeným zdrojovým kódem a vlastní integrace mohou přinést výsledky srovnatelné s komerčními řešeními, zejména pro organizace s omezeným rozpočtem.

V rychle se vyvíjejícím prostředí kybernetické bezpečnosti je nezbytné neustálé přizpůsobování a zlepšování opatření pro zabezpečení sítě. Vlastní řešení nabízí flexibilitu, škálovatelnost a bezpečnost, což umožňuje uživatelům přizpůsobit se specifickým potřebám a vyvíjejícím se hrozbám. [3]

Na konci této práce je důležité si uvědomit, že oblast analýzy síťových dat v kontextu kybernetické bezpečnosti je dynamická. Stále se objevují nové hrozby a zranitelnosti, proto je nutné vyvíjet nástroje a techniky, aby jim bylo možné čelit. Úspěšné řešení vyžaduje spolupráci velkých týmů, komerčnních společností či komunity a využívání specializovaných nástrojů a technologických inovací. [23]

Na závěr bych rád poděkoval všem vývojářům použitých knihoven (a open source komunitě) za poskytování profesionálních nástrojů v daném okruhu a vynikající použité literatuře [6] [7] [23] [3] [2] ke splnění cílů, které jsem si v této diplomové práci vytyčil.

Seznam použitých zdrojů

1. FORCEPOINT. *What is the OSI Model?* [online]. The OSI Model Defined, Explained, and Explored. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.forcepoint.com/cyber-edu/osi-model.
2. RADZISZEWSKI, Joshua. *Fundamentals of Computer Network Analysis and Engineering: Basic approaches for solving problems in the networked computing environment: Paperback – November 23, 2005.* iUniverse, 2005. ISBN 0595376703.
3. STANLEY, William D. *Network Analysis with Applications (4th Edition): 4th Edition.* Pearson, 2002. ISBN 0130602469.
4. HOPE, Network. *ISO OSI Basic reference model* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://networkhope.in/iso-osi-basic-reference-model/.
5. RUSSELL, Andrew L. *OSI: The Internet That Wasn't* [online]. How TCP/IP eclipsed the Open Systems Interconnection standards to become the global protocol for computer networking. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://spectrum.ieee.org/osi-the-internet-that-wasnt.
6. HAUGDAHL, J. Scott. *Network Analysis and Troubleshooting: PAP/CDR Edition.* Addison--Wesley, 1999. ISBN 0201433192.
7. ZHANI, Mohamed Faten; ELBIAZE, Halima. *Analysis and Prediction of Real Network Traffic* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.researchgate.net/publication/42804292_Analysis_and_Prediction_of_Real_Network_Traffic.
8. BLUECATNETWORKS. *Glossary: What is IPv4?* [online]. What is IPv4? It routes most of today's internet traffic. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://bluecatnetworks.com/glossary/what-is-ipv4/.
9. PRAMATAROV, Martin. *What is IPv4? Everything you need to know* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.cloudns.net/blog/what-is-ipv4-everything-you-need-to-know/.
10. SIDDIQUI, Aftab. *RFC 8200 – IPv6 has been standardized* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.internetsociety.org/blog/2017/07/rfc-8200-ipv6-has-been-standardized/.
11. PUJOL, Enric. *What stops IPv6 traffic in a dual-stack ISP?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://blog.apnic.net/2017/06/13/stops-ipv6-traffic-dual-stack-isp/.

12. VAUGHAN-NICHOLS, Steven. *Five ways for IPv6 and IPv4 to peacefully co-exist* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.zdnet.com/home-and-office/networking/five-ways-for-ipv6-and-ipv4-to-peacefully-co-exist/.

13. CLOUDFLARE. *What is the Internet Control Message Protocol (ICMP)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.cloudflare.com/en-gb/learning/ddos/glossary/internet-control-message-protocol-icmp/.

14. GEEKSFORGEEKS. *Internet Control Message Protocol (ICMP)* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.geeksforgeeks.org/internet-control-message-protocol-icmp/.

15. POSTEL, J. *Internet Control Message Protocol* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://datatracker.ietf.org/doc/html/rfc792.

16. KHANACADEMY. *Transmission Control Protocol (TCP)* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.khanacademy.org/computing/computers-and-internet/xcae6f4a7ff015e7d:the-internet/xcae6f4a7ff015e7d:transporting-packets/a/transmission-control-protocol--tcp.

17. GEEKSFORGEEKS. *What is Transmission Control Protocol (TCP)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.geeksforgeeks.org/what-is-transmission-control-protocol-tcp/.

18. AHMED, Aftab. *TCP Handshakes | 3-Way and 4-Way TCP handshake* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://csinfos.com/tcp-handshakes-3-way-and-4-way-handshake/.

19. CHAPPELL, Laura. *Wireshark Network Analysis (Second Edition): The Official Wireshark Certified Network Analyst Study Guide*. Laura Chappell University, 2012. ISBN 1893939944.

20. CLOUDFLARE. *What is UDP?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.cloudflare.com/en-gb/learning/ddos/glossary/user-datagram-protocol-udp/.

21. GEEKSFORGEEKS. *User Datagram Protocol (UDP)* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.geeksforgeeks.org/user-datagram-protocol-udp/.

22. SIU, Kai-Yeung; JAIN, Raj. *A Brief Overview of ATM: Protocol Layers, LAN Emulation, and Traffic Management* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.cse.wustl.edu/~jain/papers/ftp/atm_tut/index.html.

23. COLLINS, Michael. *Network Security Through Data Analysis: From Data to Action: 2nd Edition*. O'Reilly Media, 2017. ISBN 1491962844.

24. VIJAYAKUMAR, M.; PARVATHI; R.M.S. *Concept mining of high volume data streams in network traffic using hierarchical clustering* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.researchgate.net/publication/287008148_Concept_mining_of_high_volume_data_streams_in_network_traffic_using_hierarchical_clustering.

25. JEFFREY, Erman; MARTIN, Arlitt; ANIRBAN, Mahanti. *Traffic Classification Using Clustering Algorithm* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z DOI: 10.1145/1162678.1162679.

26. SHAILENDRA, Shrivastava; PREETI, Jain. *Effective Anomaly based Intrusion Detection using Rough Set Theory and Support Vector Machine* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z DOI: 10.5120/2261-2906.

27. GHANSHYAM, Dubey; NEETESH, Gupta; DR, Bhujade. *A Novel Approach to Intrusion Detection System using Rough Set Theory and Incremental SVM* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.researchgate.net/publication/344363153_A_Novel_Approach_to_Intrusion_Detection_System_using_Rough_Set_Theory_and_Incremental_SVM.

28. AL-NAFJAN, Khaled; AL-HUSSEIN, Musaed A.; ALGHAMDI, Abdullah S.; HAQUE, Mohammad Amanul; AHMAD, Iftikhar. *Intrusion Detection Using PCA Based Modular Neural Network* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: http://www.ijmlc.org/papers/194-C01280-004.pdf.

29. SINI, Lakhina a; BHUPENDRA, Joseph a; VERMA. *Feature Reduction using Principal Component Analysis for Effective Anomaly–Based Intrusion Detection on NSL-KDD* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.researchgate.net/publication/50281791_Feature_Reduction_using_Principal_Component_Analysis_for_Effective_Anomaly-Based_Intrusion_Detection_on_NSL-KDD.

30. ZHANG, Jiong; ZULKERNINE, Mohammad. *Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www2.cs.uh.edu/~acl/cs6397/Presentation/2006-IEEE-Anomaly%20Based%20Network%20Intrusion%20Detection%20with%20unsupervised%20outlier%20detection.pdf.

31. LYON, Gordon. *TCP SYN (Stealth) Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/synscan.html.

32. LYON, Gordon. *TCP Connect Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/scan-methods-connect-scan.html.

33. LYON, Gordon. *TCP FIN, NULL, and Xmas Scans* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/scan-methods-null-fin-xmas-scan.html.

34. LYON, Gordon. *TCP ACK Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/scan-methods-ack-scan.html.

35. LYON, Gordon. *TCP Maimon Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/scan-methods-maimon-scan.html.

36. LYON, Gordon. *TCP Idle Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/idlescan.html.

37. LYON, Gordon. *UDP Scan* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://nmap.org/book/scan-methods-udp-scan.html.

38. FORTINET. *What Is a Firewall?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/firewall.

39. SAHAY, Manish. *Who Invented the Firewall? History, Types, and Generations of Firewall.* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.thepcinsider.com/who-invented-firewall-history-evolution-types-generations/.

40. GREENE, Tim; BUTLER, Brandon. *Types of firewalls: What they do and what they're used for* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.networkworld.com/article/3230457/what-is-a-firewall-perimeter-stateful-inspection-next-generation.html.

41. INFOSECADEMY. *Stateful VS Stateless Firewalls And Why It Matters Which You Choose* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.infosecademy.com/stateful-vs-stateless-firewall/.

42. FORTINET. *Stateful Stateless Firewall Differences* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/stateful-vs-stateless-firewall.

43. MAKEITSECURE. *Personal Firewall* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://makeitsecure.org/personal-firewall/.

44. ICSA. *ICSA Firewall Policy Guide V2.00* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: http://www.icsa.net/services/consortia/firewalls/fwpg.shtml.

45. ZOLA, Andrew. *Mail Bomb* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.techtarget.com/searchsecurity/definition/mail-bomb.

46. IMPERVA. *Ping flood (ICMP flood)* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.imperva.com/learn/ddos/ping-icmp-flood/.

47. REDHAT. *What is a CVE?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.redhat.com/en/topics/security/what-is-cve.

48. PHIPPS, J. L. *Hackers: Can You Stop Them?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: http://www.mediainfo.com:81/ephome/news/newshtm/minfocom/1198a.htm.

49. DALTON, G. *Acceptable Risks* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: http://www.informationweek.com/698/98iursk.htm.

50. 1996, Warroom Study: *Security in Cyberspace Hearings before the Permanent Subcommittee on Investigations of the Committee on Governmental Affairs* [online]. 1996. [cit. 2023-10-15]. ISBN 0160539137. Dostupné z: http://www.warroomresearch.com/researchcollabor/infosecuritysurvey.htm.

51. BELLOVIN, Steven M. *Distributed Firewalls* [online]. 1999. [cit. 2023-10-15]. Dostupné z: https://www.cs.columbia.edu/~smb/papers/distfw.pdf.

52. TECHNOLOGY, Phonex Adaptive Firewall. *Multi-Layer Stateful Inspection White Paper* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.progressive-systems.com/.

53. FORTINET. *What is an Intrusion Detection System (IDS)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/intrusion-detection-system.

54. ANSAM KHRAISAT, Iqbal Gondal; VAMPLEW, Peter; KAMRUZZAMAN, Joarder. *Survey of intrusion detection systems: techniques, datasets and challenges* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://cybersecurity.springeropen.com/articles/10.1186/s42400-019-0038-7.

55. FORTINET. *What Is Intrusion Prevention System? Definition and Types* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/what-is-an-ips.

56. MOHANAKRISHNAN, Ramya. *What Is Intrusion Detection and Prevention System? Definition, Examples, Techniques, and Best Practices* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.spiceworks.com/it-security/vulnerability-management/articles/what-is-idps/.

57. REDHAT. *What is an intrusion detection and prevention system (IDPS)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.redhat.com/en/topics/security/what-is-an-IDPS.

58. KOKKO, Kalle. *Next-generation firewall case study* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.theseus.fi/bitstream/handle/10024/139127/kokko_kalle.pdf;jsessionid=DED7749462527E2FB2866400A41B333A?sequence=1.

59. GEIER, Eric. *Intro to Next Generation Firewalls* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.esecurityplanet.com/products/intro-to-next-generation-firewalls/.

60. CLOUDFLARE. *What is a next-generation firewall (NGFW)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.cloudflare.com/en-gb/learning/security/what-is-next-generation-firewall-ngfw/.

61. MILLER, Lawrence C.; CISSP. *Next-Generation Firewalls For Dummies* [online]. [B.r.]. [cit. 2023-10-15]. ISBN 978-0-470-93955-0. Dostupné z: https://www.cloudflare.com/en-gb/learning/security/what-is-next-generation-firewall-ngfw/.

62. FORTINET. *What Is Deep Packet Inspection (DPI)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/dpi-deep-packet-inspection.

63. NETWORKS, Palo Alto. *Next Generation Firewall* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.paloaltonetworks.com/network-security/next-generation-firewall.

64. LEWIS, Robert. *Cisco Systems* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.britannica.com/topic/Cisco-Systems-Inc.

65. CISCO. *Cisco Firepower Next-Generation Firewall (NGFW)* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://community.cisco.com/kxiwq67737/attachments/kxiwq67737/discussions-network-security/1039914/1/Cisco%20Firepower%20Next-Generation%20Firewall%20(NGFW).pdf.

66. ZIPPIA. *Fortinet History* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.zippia.com/fortinet-careers-4655/history/.

67. FORTINET. *What Is Unified Threat Management (UTM)?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/unified-threat-management.

68. FORTINET. *Fortinet Security Fabric* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.fortinet.com/solutions/enterprise-midsize-business/security-fabric.

69. ZIPPIA. *Sonicwall History* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.zippia.com/sonicwall-careers-21190/history/.

70. SONICWALL. *SonicWall Capture Advanced Threat Protection Service* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.sonicwall.com/medialibrary/en/datasheet/datasheet-sonicwall-capture-advanced-threat-protection-service.pdf.

71. SONICWALL. *How to Configure Botnet Filtering with Firewall Access Rules* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.sonicwall.com/support/knowledge-base/how-to-configure-botnet-filtering-with-firewall-access-rules/170503936467975/.

72. WIRESHARK. *About Wireshark* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.wireshark.org/about.html.

73. PROTOCOG. *QA; with the founder of Wireshark and Ethereal* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://web.archive.org/web/20160307232509/http://www.protocog.com/gerald_combs_interview.html.

74. VULCANSPHERE. *Wireshark GUI* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://en.wikipedia.org/wiki/Wireshark#/media/File:Wireshark_3.6_screenshot.png.

75. SHARPE, Richard; WARNICKE, Ed; LAMPING, Ulf. *Wireshark User's Guide* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.wireshark.org/docs/wsug_html_chunked/.

76. FLOWMON. *What is NetFlow?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.flowmon.com/en/solutions/network-and-cloud-operations/netflow-ipfix#NetFlow-Description.

77. MANAGEENGINE. *Flow-based network traffic monitoring for in-depth traffic analysis* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.manageengine.com/products/netflow/.

78. ITMANAGEWORKS. *SolarWinds Netflow Traffic Analyzer* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.itmanageworks.com/NetFlow-Traffic-Analyzer.asp.

79. SOLARWINDS. *NetFlow Traffic Analyzer* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.solarwinds.com/netflow-traffic-analyzer?CMP=ORG-BLG-TEK-X_WW_X_NP_X_X_EN_X_X-NTA-20200406_TopNetworkAnaly_X_X_VidNo_X-X.

80. COMPTIA. *What Is Wireshark and How Is It Used?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.comptia.org/content/articles/what-is-wireshark-and-how-to-use-it.

81. BARTH, Wolfgang. *Nagios: System and Network Monitoring* [online]. 2006. [cit. 2023-10-15]. ISBN 1593270704. Dostupné z: https://books.google.cz/books/about/Nagios.html?id=F3ealpHDklUC&redir_esc=y.

82. NAGIOS. *network_report*$_0$2 [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://labs.nagios.com/wp-content/uploads/2014/06/network_report_02.png.

83. NTOP. *ntop: Home* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://github.com/ntop/ntopng/wiki#using-ntopng-as-flow-collector.

84. DIGIEX. *screen-shot-2017-03-23-at-14-51-52-png* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://digiex.net/attachments/screen-shot-2017-03-23-at-14-51-52-png.15639/.

85. NTOP. *pfSense: Third Party Integrations* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.ntop.org/guides/ntopng/third_party_integrations/pfsense.html.

86. HUGH, Johnson Chadwick. *High Level Design: Distributed Network Traffic Controller* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://cmps-people.ok.ubc.ca/rlawrenc/research/Students/CJ_05_Design.pdf.

87. OPNSENSE. *About OPNsense* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://opnsense.org/about/about-opnsense/.

88. OPNSENSE. *OPNsense Freatures* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://opnsense.org/about/features/.

89. AMAZON, AWS. *What is Elasticsearch?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://aws.amazon.com/what-is/elasticsearch/.

90. GOPALAKRISHNAN, Jay. *Elasticsearch: What It Is, How It Works, And What It's Used For* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.knowi.com/blog/what-is-elastic-search/.

91. ELASTIC. *Tune for indexing speed* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.elastic.co/guide/en/elasticsearch/reference/current/tune-for-indexing-speed.html.

92. CHANG, Emily. *How to monitor Elasticsearch performance* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.datadoghq.com/blog/monitor-elasticsearch-performance-metrics/.

93. AMAZON, AWS. *What is the ELK Stack?* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://aws.amazon.com/what-is/elk-stack/.

94. ZENARMOR. *Time to empower your open source firewall!* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.zenarmor.com/zenarmor-next-generation-firewall.

95. ZENARMOR. *Security Rules* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.zenarmor.com/docs/policies/security-rules.

96. ELASTIC. *Search API* [online]. [B.r.]. [cit. 2023-10-15]. Dostupné z: https://www.elastic.co/guide/en/elasticsearch/reference/current/search-search.html.

97. BELL, Douglas; BELL, Doug; MORREY, Ian; PUGH, John R. *The Essence of Program Design*. Prentice Hall, 1997. ISBN 0133678067.

Seznam obrázků