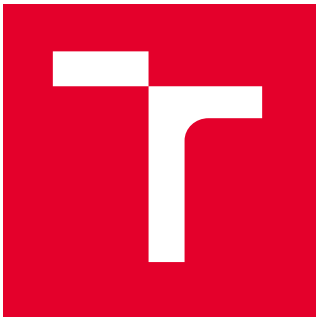


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

WEBOVÁ APLIKACE INTEGRUJÍCÍ TECHNIKY UMĚLÉ INTELIGENCE DO PROCESU TVORBY KORELAČNÍCH PRAVIDEL

WEB APPLICATION INTEGRATING ARTIFICIAL INTELLIGENCE TECHNIQUES INTO THE CORRELATION RULE CREATION PROCESS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE — **Martin Šibor**

AUTHOR

VEDOUCÍ PRÁCE — **Ing. Yehor Safonov**

SUPERVISOR

BRNO 2024

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Martin Šibor ***ID:*** 240441

Ročník: 3 ***Akademický rok:*** 2023/24

NÁZEV TÉMATU:

Webová aplikace integrující techniky umělé inteligence do procesu tvorby korelačních pravidel

POKYNY PRO VYPRACOVÁNÍ:

Hlavním cílem bakalářské práce je návrh a implementace webové aplikace sloužící k zefektivnění procesu vývoje a správy korelačních pravidel pro SIEM řešení pomocí integrace moderních platforem realizujících techniky umělé inteligence. Grafické uživatelské rozhraní bude integrovat moderní způsoby reprezentace dat a mít formu interaktivní webové aplikace. Serverová část aplikace bude napojena na vhodné externí AI platformy. Aplikace bude schopna zpracovávat korelační pravidla v různých formátech a umožňovat bezpečnostnímu správci využít potenciál moderních neuronových sítí. V teoretické části proveďte analýzu existujících architektur neuronových sítí (se zaměřením na sítě typu Transformer) a AI platforem (např. ChatGPT), nastudujte problematiku SIEM korelačních pravidel, popište techniky tvorby a testování korelačních pravidel při realizaci bezpečnostního monitoringu. Srovnejte moderní způsoby zobrazení dat při integraci na AI. Navrhněte způsoby zefektivnění procesu vývoje SIEM pravidel integrací technik umělé inteligence. Argumentujte finální výběr použitých knihoven a AI nástrojů, přičemž důraz bude kladen na jejich efektivní využití pro zadané účely, nikoliv na jejich vylepšení či rozšíření. Teoreticky a programově ošetřete rizika napojení na externí API (např. únik citlivých dat). Všechny identifikované poznatky integrujte do funkcionality aplikace.

DOPORUČENÁ LITERATURA:

[1] PEASE, Andrew. Threat Hunting with Elastic Stack: Solve complex security challenges with integrated prevention, detection, and response. Packt Publishing Ltd, 2021. ISBN 1801079803.

[2] MARTINEZ, Roberto Incident Response with Threat Intelligence. Packt Publishing Ltd, 2022. ISBN 1801070997.

Termín zadání: 5.2.2024 ***Termín odevzdání:*** 28.5.2024

Vedoucí práce: Ing. Yehor Safonov

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

ABSTRAKT

V současné době, kdy se digitalizace stává neodmyslitelnou součástí všech oblastní našich životů, se neustále zvyšuje komplexnost a sofistikovanost kybernetických hrozeb. Klíčovým prvkem v boji proti těmto kybernetickým hrozbám je bezpečnostní monitoring. Důležitým nástrojem bezpečnostního monitoringu jsou systémy SIEM, které umožňují včasnou detekci a reakci na potenciální útoky na základě korelačních pravidel. Hlavním přínosem této práce je návrh a implementace webové aplikace, která integruje techniky umělé inteligence do procesu tvorby a správy korelačních pravidel pro systémy bezpečnostních monitoringů s cílem zefektivnit proces tvorby, úprav a pochopení korelačních pravidel. Práce se nejdříve věnuje teoretickému úvodu do oblasti zpracování přirozeného jazyka a moderních neuronových sítí, zejména architektury transformers, která je základem generativních modelů umělé inteligence (např. ChatGPT, Gemini). Dále jsou představeny principy bezpečnostního monitoringu, systémů pro zpracování záznamů událostí, koncept generalizace korelačních pravidel a v neposlední řadě výzvy spojené se správou a udržováním korelačních pravidel, které integrace umělé inteligence do těchto procesů výrazně odbourává. Praktická část práce popisuje návrh a implementaci webové aplikace, která využívá modely gpt-4 a gpt-3.5-turbo od společnosti OpenAI a model Gemini Ultra 1.0 od společnosti Google pro tvorbu nových korelačních pravidel, úpravu existujících pravidel a jejich vysvětlením a interpretací pro snazší pochopení a rychlejší nasazení. Aplikace je navržena s ohledem na uživatelskou přívětivost a efektivitu. Výsledky práce ukazují, že integrace umělé inteligence do procesu tvorby korelačních pravidel přináší významné zlepšení efektivity. Webová aplikace umožňuje uživatelům snadno vytvářet a upravovat korelační pravidla. Aplikace také umožňuje uživatelům lépe porozumět korelačním pravidlům a umožňuje jim takto rychleji reagovat na potenciální hrozby.

KLÍČOVÁ SLOVA

Anonymizace, architektura mikroservis, architektura transformer, Bard, bezpečnostní monitoring, ChatGPT, Docker, Flask, Gemini, GPT-4, korelační pravidla, neuronové sítě, Sigma, SIEM, umělá inteligence, Vue, webová aplikace, zpracování přirozeného jazyka.

ABSTRACT

Currently, as digitalization becomes an integral part of all areas of our lives, the complexity and sophistication of cyber threats are constantly increasing. A key element in the fight against these cyber threats is security monitoring. An important tool for security monitoring are SIEM systems, which allow for early detection and response to potential attacks based on correlation rules. The main contribution of this work is the design and implementation of a web application that integrates artificial intelligence techniques into the process of creating and managing correlation rules for security monitoring systems, with the aim of streamlining the process of creating, modifying, and understanding correlation rules. The work first provides a theoretical introduction to the field of natural language processing and modern neural networks, particularly the transformer architecture, which is the basis of generative artificial intelligence models (e.g., ChatGPT, Gemini). It then introduces the principles of security monitoring, log management systems, the concept of correlation rule generalization, and, last but not least, the challenges associated with managing and maintaining correlation rules, which the integration of artificial intelligence into these processes significantly reduces. The practical part of the work describes the design and implementation of a web application that utilizes the gpt-4 and gpt-3.5-turbo models from OpenAI and the Gemini Ultra 1.0 model from Google for creating new correlation rules, modifying existing rules, and explaining and interpreting them for easier understanding and faster deployment. The application is designed with user-friendliness and efficiency in mind. The results of the work show that the integration of artificial intelligence into the correlation rule creation process brings significant efficiency improvements. The web application allows users to easily create and modify correlation rules. The application also allows users to better understand correlation rules, enabling them to respond to potential threats more quickly.

KEYWORDS

Anonymization, microservice architecture, transformer architecture, Bard, security monitoring, ChatGPT, Docker, Flask, Gemini, GPT-4, correlation rules, neural networks, Sigma, SIEM, artificial intelligence, Vue, web application, natural language processing.

ŠIBOR, Martin. *Webová aplikace integrující techniky umělé inteligence do procesu tvorby korelačních pravidel*. Bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2024. Vedoucí práce: Ing. Yehor Safonov

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Martin Šibor
VUT ID autora:	240441
Typ práce:	Bakalářská práce
Akademický rok:	2023/24
Téma závěrečné práce:	Webová aplikace integrující techniky umělé inteligence do procesu tvorby korelačních pravidel

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Zde bych rád poděkoval vedoucímu mé bakalářské práce, panu Ing. Yehoru Safonovi, za jeho odborné vedení, časté konzultace, cenné rady a neustálou podporu. Jeho podnětné rady a návrhy pro mě byly neocenitelnou inspirací a motivovaly mě dosáhnout co nejlepšího výsledku. Dále moje díky patří také bezpečnostním specialistům ze společnosti Aricoma, za jejich odborné konzultace a zkušenosti, které mi předali. Nakonec bych chtěl poděkovat také své rodině, která mě vždy ve všem podporovala a stála při mně.

Obsah

Seznam obrázků

Úvod

V dnešní době, kdy se svět čím dál víc přesouvá do online prostředí, čelíme neustále novým hrozbám v kyberprostoru. Útočníci mají k dispozici čím dál sofistikovanější nástroje a metody. Proto je čím dál důležitější mít kvalitní bezpečnostní monitoring kyberprostoru, a v tom hrají klíčovou roli systémy SIEM (angl. *Security Information and Event Management*), které dokážou zpracovávat obrovské množství dat a upozornit na případné bezpečnostní hrozby (2). Tyto systémy fungují na principu korelačních pravidel, která ale vyžadují odborníky a neustálé úpravy, aby byly aktuální a dokázaly čelit i nejnovějším hrozbám. V posledních letech se umělá inteligence, hlavně modely zpracování přirozeného jazyka založené na architektuře transformers, staly klíčovými nástroji pro zefektivnění spousty procesů.

Tato práce se tedy zaměřuje na využití generativní umělé inteligence, konktérně využití modelů gpt-4 a gpt-3.5-turbo od společnosti OpenAI a modelu Gemini (před únorem 2024 pod označením Bard) od společnosti Google, pro tvorbu, úpravu a vysvětlení korelačních pravidel v rámci bezpečnostního monitoringu. Hlavním přínosem práce je vytvoření webové aplikace, která bezpečnostním analytikům usnadní práci s korelačními pravidly a pomůže jim rychleji reagovat na nové hrozby.

Práce je rozdělena do šesti kapitol. Teoretická část práce se nejprve věnuje zpracování přirozeného jazyka a moderních neuronových sítí (1), zejména architektury transformers (1.3). Druhá kapitola popisuje principy bezpečnostního monitoringu kyberprostoru (2), systémy SIEM (2.1) a korelační pravidla (2.2). Třetí kapitola se zabývá možnostmi, jak využít umělou inteligenci při tvorbě korelačních pravidel, ve které byla provedena podrobná analýza možných případů užití (3). Ve čtvrté kapitole je představen návrh webové aplikace včetně jejích cílů, požadavků a použitých technologií (4). Pátá kapitola popisuje samotnou implementaci aplikace (5), a v šesté kapitole jsou prezentovány dosažené výsledky(6.1), jejich zhodnocení (6.2) a taky nápady na budoucí vývoj (6.3).

1 Zpracování přirozeného jazyka a moderní neuronové sítě

V rámci této kapitoly bude probrán současný vývoj a integrace dvou zásadních oblastí výzkumu ve sféře umělé inteligence. Těmi jsou:

- Zpracování přirozeného jazyka, neboli NLP (*Natural Language Processing*), které se zabývá pochopením, interpretací a generováním lidského jazyka pomocí počítačů a stalo se klíčovým prvkem v mnoha aplikacích, od automatického překladu po virtuální asistenty založené na umělé inteligenci (detailně rozebráno v kapitole číslo 1.1). [1]
- Moderní neuronové sítě, zejména ty, které využívají architekturu transformer. Takové neuronové sítě nám přinášejí nové možnosti v oblasti strojového učení a umělé inteligence (detailně rozebráno v kapitole číslo 1.2). [2]

V následujících několika podkapitolách bude probíráno, jak se tyto dvě technologie vzájemně prolínají a doplňují. Zahrnuty budou základní principy a techniky používané v NLP, včetně mechanismů porozumění a generování jazyka (viz podkapitola číslo 1.1.1) a klasifikace problémů řešených pomocí NLP (viz podkapitola číslo 1.1.2). Dále budou probrány různé typy architektur neuronových sítí, jejich základní principy (viz podkapitola číslo 1.2.1) a aplikace, zejména v kontextu generativních sítí a architektury transformer (viz podkapitola číslo 1.3). Nakonec bude probrán význam generativních modelů pro společnost a výzvy spojené s jejich vývojem a využitím (detailně rozebráno v kapitole číslo 1.4).

1.1 Základní aspekty zpracování přirozeného jazyka

Zpracování přirozeného jazyka je odvětvím informatiky, přesněji řečeno podmnožinou umělé inteligence, která se zaměřuje na umožnění počítačům porozumět textu a mluvenému slovu. Tato oblast zahrnuje vývoj algoritmů a modelů, které počítačům umožňují rozumět, interpretovat a generovat lidský jazyk, a to jak ve psané, tak mluvené formě. [1]

Zpracování přirozeného jazyka v počítačích se dělí do dvou hlavních částí, a to porozumění přirozenému jazyku a generování přirozeného jazyka. Tyto dvě části zpracování přirozeného jazyka budou probrány níže v podkapitole číslo 1.1.1. Kategorizace problémů, které dokáže zpracování přirozeného jazyka řešit potom bude probráno v rámci podkapitoly číslo 1.1.2.

1.1.1 Mechanismy porozumění a generování jazyka

Zpracování přirozeného jazyka lze rozdělit na dvě části – porozumění přirozenému jazyku (*NLU*) a generování přirozeného jazyka (*NLG*).

Porozumění přirozenému jazyku je technika, která se zaměřuje na syntaktickou strukturu jazyka a také na odvozování sémantického významu[1] z něj. Patří sem například rozpoznávání řeči, rozpoznávání pojmenovaných entit nebo klasifikace textu. [3, 4]

U generování přirozeného jazyka jsou potom vědomosti, které jsou odvozeny z porozumění přirozenému jazyku, dále rozvíjeny prostřednictvím generování jazyka. Příklady zahrnují odpovídání na otázky, generování textu nebo generování řeči. [3, 4]

Aplikace zpracování přirozeného jazyka, jako je překlad jazyků nebo automatické doplňování vyhledávání (více o kategoriích řešených problémů pomocí NLP bude probráno v kapitole 1.1.2), se mohou zdát jednoduché, ale jsou vyvíjeny pomocí řady základních a jednoduchých technik NLP. Proto je vhodné zmínit i dva hlavní typy algoritmů NLP, které se běžně používají. [1, 5]

- **Systémy založené na pravidlech**: Tyto algoritmy používají předdefinovaná pravidla a vzory pro zpracování a porozumění jazyku.
- **Systémy založené na strojovém učení**: Tyto algoritmy využívají statistické a techniky strojového učení k učení se z dat a k provádění predikcí nebo klasifikací na základě vzorů v textu.

1.1.2 Klasifikace problémů řešených pomocí NLP

Zpracování přirozeného jazyka zahrnuje řadu technik a postupů, které umožňují počítačům porozumět, interpretovat a generovat lidský jazyk. Níže budou probrány ty nejběžnější a nejvíce populární techniky používané v NLP pro práci s textem.[2]

- **Tokenizace (*tokenization*)** – Proces rozkládání textu na menší jednotky, jako jsou slova, věty, nebo dokonce jednotlivé znaky. Tato technika je základním krokem pro většinu úloh zpracování přirozeného jazyka, neboť rozděluje text na manipulovatelné části pro další analýzu. [1, 5]
- **Redukce na kořen slova a lemmatizace (*stemming and lemmatization*)** – Tyto dvě metody se zabývají redukcí slov na jejich základní tvar. Redukce na kořen slova odstraňuje koncovky slov, aby našel tzv. kořen slova,

[1] Sémentika je nauka o významu výrazů z různých strukturních úrovní jazyka – slov, slovních spojení a vět. Více informací o sémantice lze dočíst na odkaze <https://cs.wikipedia.org/wiki/S%C3%A9mantika>.

[2] Detailnější popis jednotlivých technik používaných v NLP lze nalézt na odkaze <https://www.projectpro.io/article/10-nlp-techniques-every-data-scientist-should-know/415#toc-1>.

zatímco lemmatizace konvertuje slovo na jeho lexikální základní tvar (*lexém*[3]), beroucí v úvahu jeho gramatický kontext. [1, 5]

- **Odstranění stop slov (*stop words removal*)** – Tato technika zahrnuje identifikaci a odstranění běžných slov (*tzv. stop slov*), která jsou ve většině kontextů považována za nevýznamná pro analýzu. Příklady zahrnují předložky, spojky, a další často se vyskytující slova. [1, 5]
- **TF-IDF (*term frequency-inverse document frequency*)** – Statistická metoda používaná k určení důležitosti slova v dokumentu vzhledem k celému souboru dokumentů. TF-IDF hodnotí významnost slova tím, že porovnává jeho frekvenci v daném dokumentu s frekvencí v jiných dokumentech. [1, 5]
- **Extrakce klíčových slov (*keyword extraction*)** – Proces identifikace nejdůležitějších a nejvýznamnějších slov nebo frází v textu. Tato technika je klíčová pro různé aplikace, jako je sumarizace textu, vyhledávání informací nebo textová analýza. [1, 5]
- **Vektorové reprezentace slov (*word embeddings*)** – Metoda převedení slov do vektorového prostoru, kde každé slovo je reprezentováno vektorem, který zachycuje jeho sémantický význam a vztahy s ostatními slovy. [1, 5]
- **Analýza sentimentu (*sentiment Analysis*)** – Technika určená k detekci a klasifikaci emocionálního tónu nebo sentimentu vyjádřeného v textu. Často se používá k analýze názorů a postojů v uživatelských recenzích, sociálních médiích a dalších typech textu. [1, 5]
- **Modelování témat (*topic modelling*)** – Metoda určená k identifikaci, pozorování a extrahování hlavních témat z velkého objemu textových dat. Tato technika pomáhá pochopit hlavní myšlenky a koncepty obsažené v textech. [1, 5]
- **Sumarizace textu (*text summarization*)** – Proces vytváření stručného shrnutí delšího textového obsahu. Cílem je zachovat klíčové informace a hlavní myšlenky originálního textu v kompaktnější formě. [1, 5]
- **Rozpoznání pojmenovaných entit (*named entity recognition*)** – Úloha spočívající v identifikaci a klasifikaci specifických entit, jako jsou jména osob, organizací, geografických názvů a dalších, které se vyskytují v textu. [1, 5]

1.2 Architektura neuronových sítí

Neuronové sítě představují základní pilíř v současném výzkumu v oblasti umělé inteligence. Jsou to modely strojového učení inspirované lidským mozkem. Neuronové

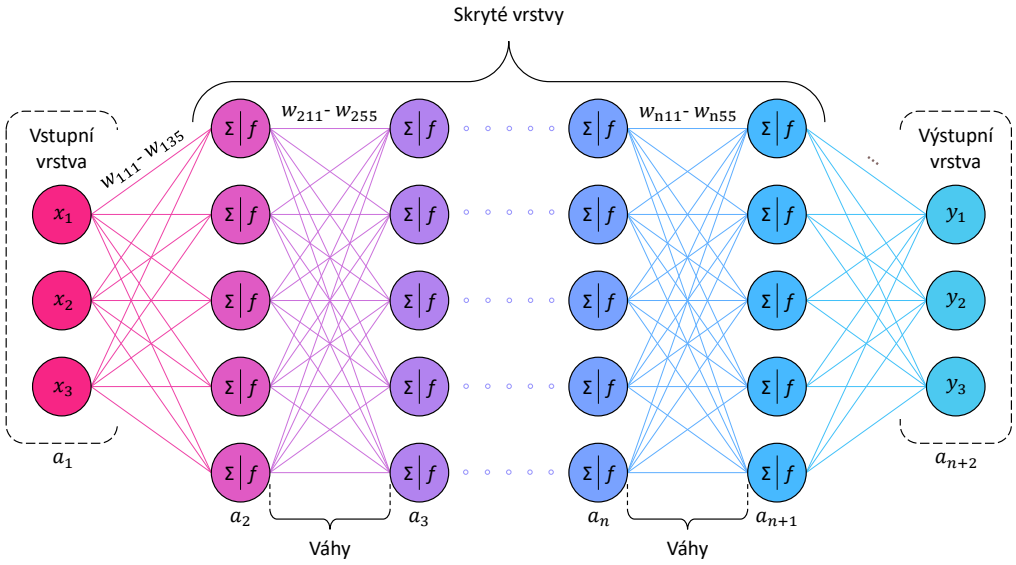
[3]Lexém je souhrnné označení všech tvarů téhož slova nebo slovního spojení. Více o něm lze dočíst na odkaze <https://cs.wikipedia.org/wiki/Lex%C3%A9m>.

sítě jsou tvořeny vrstvami vzájemně propojených uzlů zvaných neurony, které zpracovávají vstupní data a předávají výsledky dalším vrstvám. Každý neuron aplikuje na svůj vstup matematickou funkci, čímž produkuje výstup, který je následně předán do další vrstvy. Architektura neuronové sítě potom určuje její složitost, kapacitu a funkčnost. [6]

Tato část se zaměří na vysvětlení, jak neuronové sítě fungují, a jak jsou využívány v kontextu NLP. Podrobně se podíváme na různé typy neuronových sítí a jejich aplikace.

1.2.1 Základní principy moderních neuronových sítí

Princip fungování architektury neuronové sítě spočívá v sérii propojených vrstev, které transformují vstupní data na smysluplné reprezentace. Vstupní vrstva přijímá surová data, která jsou následně procházena jednou nebo více skrytými vrstvami provádějícími matematické výpočty. Výstupní vrstva produkuje konečné výsledky, jako jsou předpovědi nebo klasifikace. [7, 8]



Obr. 1.1: Architektura neuronové sítě.

Jak lze vidět na obrázku 1.1, spojení mezi neurony jsou reprezentována váhami w_{nxy}, kde n je číslo sloupce, x je pořadí vstupního neuronu ve sloupci a y pořadí výstupního neuronu v následujícím sloupci. Každá váha určuje sílu a vliv vstupů na výstup neuronu. Během fáze trénování neuronová síť upravuje váhy na základě poskytnutých dat a požadovaných výsledků, čímž postupně zlepšuje svou schopnost

přesně předpovídat nebo klasifikovat. Specifikuje, jak jsou vrstvy, uzly a spojení sítě organizovány za účelem zpracování a analýzy dat. [6, 9]

1.2.2 Generativní sítě a jejich aplikace

Generativní sítě lze popsat jako odvětví strojového učení, jehož cílem je trénování sítě (*modelu*) na produkci nových dat, která se podobají existující datové sadě. [11]

Například si lze představit, že máme sadu fotografií koček. Potom lze trénovat generativní model na této sadě, aby zachytil pravidla, která určují složité vztahy mezi pixely na obrázcích koček. Poté je možno vzorkovat z tohoto modelu, abychom vytvořili nové realistické obrázky koček, které v původní datové sadě neexistovaly.

Pro výstavbu generativního modelu je potřeba sada dat, která obsahuje mnoho příkladů dané entity, kterou se snažíme generovat. To se nazývá trénovací sada a každý datový bod se nazývá vzorek.

Každý vzorek se skládá z mnoha vlastností. U problému generování obrázků jsou vlastnostmi obvykle hodnoty jednotlivých pixelů. U generování textu by to mohla být jednotlivá slova nebo skupiny písmen. Cílem je vytvořit model, který může generovat nové sady vlastností, které vypadají, jako by byly vytvořeny podle stejných pravidel jako původní data. [11, 12]

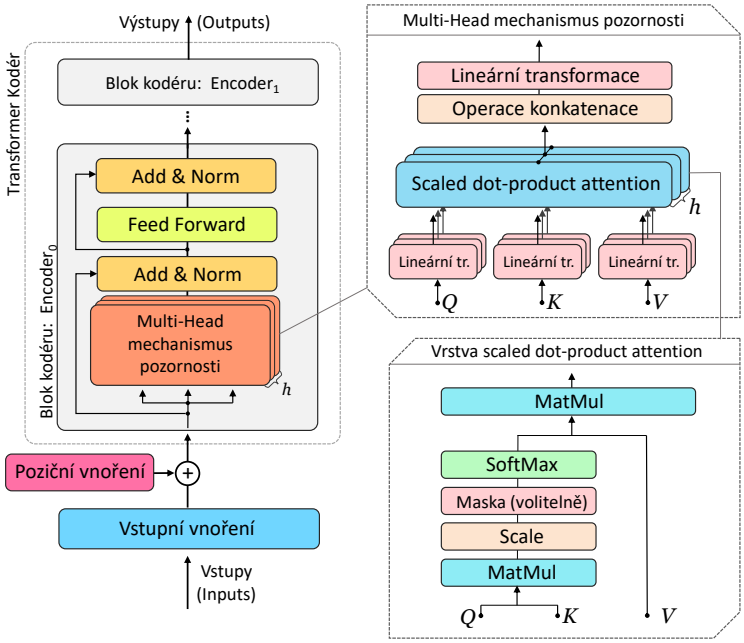
Konceptuálně je pro generování obrázků tento úkol nesmírně složitý, vzhledem k obrovskému množství způsobů, jakými lze přiřadit hodnoty jednotlivých pixelů, a relativně malému počtu takových uspořádání, které tvoří obrázek entity, kterou se snažíme generovat. [11, 12]

Generativní model musí být také pravděpodobnostní, nikoli deterministický, protože chceme být schopni vzorkovat mnoho různých variant výstupu, namísto toho, abychom vždy získali stejný výstup. Pokud by byl model pouze pevným výpočtem, jako je například průměrná hodnota každého pixelu v tréninkové sadě dat, nebyl by generativní. Generativní model musí zahrnovat náhodnou složku, která ovlivňuje jednotlivé vzorky generované modelem. [13, 17]

Jinými slovy, lze si představit, že existuje nějaké neznámé pravděpodobnostní rozdělení, které vysvětluje, proč jsou některé obrázky pravděpodobně nalezeny v tréninkové sadě a jiné ne. [11]

1.3 Architektura *transformers*

Transformer je architektura hlubokého učení, která byla původně navržena v roce 2017. Spoléhá se na paralelní mechanismus *multi-head attention* (*vícehlavého pozornostního mechanismu*) [10, 14]. Lišila se tím, že vyžaduje méně času na trénování než předchozí rekurentní neuronové architektury, jako jsou například LSTM (*Long*



Obr. 1.2: Architektura *transformer* modelu [16].

Short-Term Memory) [15]. Její pozdější variace byla rozšířeně přijata pro trénování velkých jazykových modelů na rozsáhlých jazykových datech, jako jsou korpusy Wikipedie a Common Crawl, díky paralelizovanému zpracování vstupní sekvence.

Transformer model, zobrazený na obrázku 1.2, se skládá z několika identických vrstev (*typicky 6*). Zpracování vstupní sekvence je prováděno kodérem, který generuje reprezentaci pro předání dekodéru. Následně je tato informace využívána dekodérem k generování cílové sekvence. V následujících podkapitolách budou popsány jednotlivé funkce architektury transformer. [2, 10]

Kodér a dekodér

Model se skládá ze dvou hlavních částí, a to kodéru (*encoder*) a dekodéru (*decoder*). Kodér je zodpovědný za zpracování vstupní sekvence a vytvoření reprezentace, která zachycuje význam a vztahy mezi jednotlivými prvky sekvence. Dekodér pak využívá reprezentaci k vytvoření výstupní sekvence, tedy například k překladu věty do jiného jazyka. [2, 10]

Mechanismus pozornosti

Klíčovou součástí transformer modelu je mechanismus pozornosti (*self-attention*). Tento mechanismus umožňuje každému prvku ve vstupní sekvenci zvážit všechny ostatní prvky při vytváření své reprezentace. To je dosaženo pomocí trojice matic

(Query – Q, Key – K, Value – V), které jsou pro každý prvek vypočítány lineární transformací jeho původní reprezentace. Výsledná reprezentace prvku je pak váženým součtem všech vektorů V, kde váhy jsou určeny mírou podobnosti mezi vektory Q a K a je dána vztahem:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\text{T}}}{\sqrt{d_k}}\right)V, \tag{1.1}$$

Multi-head mechanismus pozornosti

Transformer model využívá vícehlavový mechanismus pozornosti (*multi-head attention*), což je rozšíření mechanismu self-attention popsaného v kapitole výše. Místo jedné sady matic (Query – Q, Key – K, Value – V) se zde však používá několik sad, které pracují paralelně. To umožňuje modelu zachytit různé typy vztahů mezi prvky sekvence. [2, 10]

Poziční vnoření

Protože transformer model neobsahuje žádné rekurentní spojení, je nutné do něj explicitně zakódovat informaci o pořadí prvků ve vstupní sekvenci. To je dosaženo pomocí pozičního vnoření (*positional encoding*), což je vektor, který je přičten k původní reprezentaci každého prvku. [2, 10]

Dopředné neuronové sítě

Kromě mechanismu pozornosti obsahuje transformer model také několik vrstev plně propojených neuronových sítí (*feed-forward networks*). Tyto sítě slouží k dalšímu zpracování reprezentací prvků a k zachycení nelineárních vztahů mezi nimi. [2, 10]

Reziduální spojení a normalizace vrstev

Pro zlepšení stability a urychlení trénování modelu se používají takzvané reziduální spojení (*residual connections*) a normalizace vrstev (*layer normalization*). Reziduální spojení umožňují hlubší vnoření do sítě, zatímco normalizace vrstev pomáhá stabilizovat hodnoty aktivačních funkcí. [2, 10]

1.4 Sociální a technologický vliv generativní AI

V této kapitole bude zaměřena pozornost na sociální, etické a technologické dopady generativních modelů, včetně problémů spojených s jejich vývojem a využitím. Bude rozebrán, jaký vliv tyto technologie mají na společnost, jejich potenciální výzvy

a rizika. Vliv generativní umělé inteligence, jako jsou ChatGPT, Midjourney, Copilot a další, se projevuje a bude se projevovat v různých odvětvích a organizacích.

Mezi bezpečnostními manažery se rozrůstají obavy. Mnoha CISO (*chief information security officer*) je vyvíjen tlak na široké uplatnění generativní umělé inteligence, přičemž jsou si vědomi, že její používání může přinést potenciální vážná rizika. [18]

Podobně jako při zrodu internetu, cloudu, chytrých telefonů a sociálních médií, nacházíme se na pomyslné hranici přijímání technologie a čelíme příležitosti toho využít. S tímto je však spojeno mnoho výhod, ale i nevýhod, z kterých mnoho bude rozebráno v rámci této kapitoly. [19]

Výhody generativní umělé inteligence:

- **Kreativita a Inovace** – Generativní umělá inteligence je schopná vytvářet nový, kreativní a originální obsah v mnoha oblastech od umění a hudby, až po psaní poezie nebo vytváření softwaru.
- **Automatizace a efektivita** – Generativní AI umožňuje automatizaci úkolů generování obsahu, čímž dochází k úspoře času a zdrojů. Lze ji použít například pro návrh emailů, generování kódu nebo vytváření marketingových materiálů.
- **Porozumění přirozenému jazyku** – Generativní modely vynikají v porozumění a generování lidského jazyka, což umožňuje přirozenější a kontextově relevantní interakce, jak již bylo probráno v kapitole 1.1.
- **Personalizace obsahu** – Tyto modely lze do jisté míry nastavit tak, aby odpovídaly konkrétním preferencím uživatelů, což umožňuje generování personalizovaného obsahu.
- **Překlad jazyků a vícejazyčná komunikace** – Generativní modely prokázaly schopnost v přesném překladu mezi jazyky, což dělá globální komunikaci přístupnější a efektivnější.

Nevýhody generativní umělé inteligence:

- **Nekorektní a zkreslené výstupy** – Generativní modely mohou neúmyslně generovat zkreslený nebo politicky nekorektní obsah, což odráží zkreslení v trénovací sadě. Zajištění etického používání a minimalizace zkreslení je výzvou pro každý model.
- **Nedostatek odpovědnosti** – U obsahu autonomně generovaného modelem je obtížné přiřadit odpovědnost, což může být problémové v situacích, kde je odpovědnost klíčová.
- **Dobrý učitel ale zlý pán** – Nadměrná závislost na generativní umělé inteligenci může vést ke snížení lidského kritického myšlení a kreativity.

- **Náročnost na zdroje** – Trénink a provoz velkých generativních modelů je výpočetně velmi náročný, což vyžaduje značné výpočetní zdroje a energii.
- **Bezpečnostní a obavy o soukromí** – Existují potenciální rizika spojená s používáním generativní umělé inteligence pro nekalé účely, jako je generování falešných zpráv nebo deepfakes[4].

Generativní umělá inteligence tedy nabízí obrovský potenciál pro inovaci a efektivitu v různých oblastech. Její nasazení však vyžaduje pečlivý přístup, řešení etických obav a potenciálních zkreslení. Je zásadní dosáhnout harmonické rovnováhy mezi lidským faktorem a umělou inteligencí. [19]

1.4.1 Význam *SaaS* v souvislosti s GenAI

V souvislosti s generativní umělou inteligencí představuje model SaaS (*Software as a Service*) způsob, jakým jsou softwarové aplikace poskytovány a spravovány prostřednictvím internetu. Tento model se vyznačuje tím, že umožňuje snadný přístup k pokročilým nástrojům generativní umělé inteligence pro široké spektrum uživatelů. Díky SaaS může kdokoliv využívat umělou inteligenci bez toho, aby musel investovat do drahého hardware.

V rámci SaaS je zajištěno, že software je pravidelně aktualizován a udržován poskytovatelem služby, což zbavuje zákazníky od nutnosti tomuto se věnovat. SaaS také umožňuje flexibilní škálování služeb podle aktuálních potřeb uživatelů, což znamená, že mohou snadno upravovat úroveň a rozsah využívaných služeb. Tímto způsobem SaaS zjednodušuje a zpřístupňuje používání generativní umělé inteligence pro širokou škálu aplikací a uživatelů. [20, 21]

1.4.2 Srovnání moderních generativních modelů

Tato kapitola se zaměří na nejpopulárnější generativní modely, které jsou dostupné veřejnosti. Budou probrány rozdíly mezi ChatGPT a Bardem (od února 2024 již Gemini), generativními modely umělé inteligence v podobě chatbotů, které využívají strojové učení a zpracování přirozeného jazyka pro pochopení a generování lidského jazyku. [22]

ChatGPT, vyvinutý společností OpenAI, je založen na modelu GPT (*generativního předtrénovaného transformeru*). Tento model byl trénován na datovém souboru textů a kódu z internetu, a to před zářím 2021. ChatGPT je využíván pro generování krátkých odstavců, souhrnů a textových odpovědí.

[4] Jako deepfakes se obecně označují manipulované fotografie, videa a zvuky, které se tváří jako opravdové. Více informací o nich lze nalézt na odkaze <`https://cs.wikipedia.org/wiki/Deepfake`>

Na druhé straně Bard, vytvořený společností Google, je poháněn modelem PaLM 2 (*Pathways Language Model*). Byl navržen pro interaktivní konverzace a má přístup k internetu, aby poskytoval aktuální odpovědi.

Rozdíl mezi ChatGPT a Bardem spočívá v jejich datech a designové filozofii. ChatGPT je více orientovaný na text, zatímco Bard je zaměřen na konverzace s přístupem k reálným datům, což mu umožňuje poskytovat aktuálnější odpovědi. V tabulce 1.1 jsou vypsány nejpodstatnější rozdíly těchto dvou modelů.

	ChatGPT	Bard
Tréninková data	Natrénován na textu z internetu do září 2021.	Natrénován na specifické konverzační datové sadě
Jazykový model	**Zdarma:** GPT-3.5 **Placený:** GPT-4	Pathways Language Model (PaLM 2)
Přístup k internetu	ChatGPT Plus má ve výchozím stavu přístup k Bingu. Prostřednictvím pluginů potom i ke Googlu.	Bard má ve výchozím stavu přístup k internetu Google.
Podpora více jazyků	Rozumí a umí generovat text ve více jak 20 jazycích	Podporuje americkou angličtinu, japonštinu a korejštinu a umí překládat další jazyky.
Nejlepší využití	Generování obsahu, generování kódu a ladění. ChatGPT plus potom umí i analýzu PDF a generování obrázků.	Učení o tématech, výzkum a analýza obrazů.
Cena	**GPT-3.5**: Zdarma **GPT-4**: 20 $ měsíčně	Zdarma

Tab. 1.1: Tabulka rozdílů mezi ChatGPT a Google Bard.

2 Principy bezpečnostního monitoringu kyberprostoru

Kyberprostor je nedílnou součástí dnešního moderního digitálního světa, ve kterém se každý z nás nějakým způsobem pohybuje a je do něj zapojen. V tomto světě je právě bezpečnostní monitoring jedním ze základních pilířů kybernetické obrany, neboť bez něho by se internet mohl stát daleko nebezpečnějším místem. Bezpečnostní monitoring v kyberprostoru lze přirovnat k právnímu řádu, který chrání a udržuje pořádek ve společnosti. K tomu mohou být využity moderní systémy pro sledování aktivit a předcházení nežádoucímu chování. Způsoby a techniky jak tohoto docílit budou probrány dále v textu.

Bezpečnost na internetu nebyla po dlouhou dobu vnímána jako problém, neboť digitální svět nebyl tak rozšířený jako tomu je dnes. Mnoho lidí považuje za samozřejmost, že jejich data sdílená online jsou chráněna a zůstávají soukromá. Nicméně realita je často jiná. V rámci rychle se rozvíjejícího digitálního světa lidé teprve v posledních pár letech začínají chápat, že data, která online poskytují, mohou přímo či nepřímo ovlivnit je samotné či ostatní, je-li proces jakým je s daty nakládáno nesprávný.

Bezpečnostní monitoring je proto zásadní pro ochranu v kyberprostoru. Je klíčový zejména pro organizace, které intenzivně využívají počítačové systémy. Umožňuje dohled nad celou infrastrukturou a s adekvátním nasazením je možné monitorovat aktivitu v celé síti. Pro větší organizace je však nereálné sledovat provoz sítě v reálném čase kvůli obrovskému množství generovaných událostí. Proto bylo nutné vyvinout systémy, které by za nás analyzovaly provoz a poskytovaly pouze relevantní informace, efektivně oddělující důležité události od nepodstatných.

V následujících podkapitolách budou probrány klíčové aspekty a technologie kybernetické bezpečnosti, které zahrnují systémy pro zpracování záznamů událostí jako SIEM a SOAR a jejich role a význam v bezpečnostním monitoringu. Důraz bude kladen i na procesy a výzvy spojené s tvorbou, správou a udržováním korelačních pravidel, která jsou zásadní pro efektivní identifikaci a reakci na potenciální bezpečnostní hrozby.

2.1 Systémy pro zpracování záznamů událostí

Zpracování bezpečnostních informací ve formě záznamů událostí (logů) je klíčovou součástí bezpečnostního monitoringu. K tomu slouží technologie jako jsou SIEM (*angl. Security Information and Event Management*) – z původních SIM (*Security Information Management*) a SEM (*Security Event Management*), nebo SOAR (*Security Orchestration, Automation, and Response*).

Takové systémy pro zpracování záznamů událostí jako je SIEM nebo SOAR a další systémy pro správu událostí (logů) jsou nezbytné pro efektivní zpracování a analýzu dat v kybernetické bezpečnosti. Tyto systémy sbírají a analyzují velké množství dat z různých zdrojů, což umožňuje organizacím odhalovat potenciální bezpečnostní hrozby a včas na ně reagovat. [24]

Podrobněji budou zmíněné technologie probrány níže v textu.

Security Information Management

SIM (*Security Information Management*), neboli správa informací o bezpečnosti, je technologie zaměřená na správu záznamů událostí, známých jako logy. SIM slouží jako centrální úložiště pro tyto logy, umožňující jejich pozdější analýzu. Hlavním cílem tohoto systému je shromažďování a ukládání záznamů událostí na jednom místě pro budoucí využití. Tento proces je však efektivní pouze v případě, že logy obsahují relevantní a smysluplné informace. [25, 26]

Security Event Management

SEM (*Security Event Management*), neboli správa bezpečnostních událostí, se zaměřuje na procesování dat v reálném čase, monitorování, korelaci a hlášení bezpečnostních událostí. Na rozdíl od SIM, SEM nejen shromažďuje události, ale také je aktivně analyzuje, čímž dává logům smysl. SEM pomáhá identifikovat a řešit bezpečnostní události hned v případě, že se objeví, což je zásadní pro včasnou reakci na potenciální bezpečnostní rizika. [25, 26]

Security Information and Event Management

SIEM (*Security Information and Event Management*) představuje spojení technologií SIM a SEM, což přináší komplexní řešení pro správu bezpečnostních informací a událostí. Tato technologie kombinuje shromažďování dat, jejich normalizaci a analýzu na základě korelačních pravidel. SIEM prezentuje data ve formátu přívětivém a srozumitelném pro uživatele, což je klíčové pro efektivní sledování a zabezpečení infrastruktury. SIEM adresuje limitace lidských kapacit v monitorování a analýze

událostí, což umožňuje sledovat všechny události z celé infrastruktury v reálném čase a generovat potenciální bezpečnostní incidenty. [24, 27]

Security Orchestration, Automation, and Response

SOAR (*Security Orchestration, Automation, and Response*) je technologie, která dále rozšiřuje možnosti SIEM o automatizaci a orchestraci bezpečnostních procesů. Tato technologie umožňuje automatické reagování na detekované potenciální bezpečnostní incidenty, což zvyšuje efektivitu a snižuje reakční dobu potřebnou pro zásah. SOAR a SIEM společně dokáží vytvořit robustní systém pro komplexní zpracování a reakci na potenciální bezpečnostní hrozby. [28]

2.2 Korelační pravidla pro detekci incidentu v kyberprostoru

Korelace je proces identifikace a vytváření vztahů mezi více událostmi, napříč jedním či více zařízeními. Korelace událostí je velmi důležitá, protože jediný log nemusí mít pro bezpečnostního analytika žádnou vypovídající hodnotu či se nemusí jevit sám o sobě jako podezřelý. [29]

Ku příkladu pokud se uživatel úspěšně přihlásí ke svému účtu uvnitř organizace a ve stejnou chvíli se stejný uživatel přihlásí z jiného místa ve stejné zemi, pokud se na to bude pohlížet jako na jednotlivé události, nejeví to žádné známky podezření. Avšak pokud budeme nahlížet na tyto události jako vzájemně související, toto přihlášení ve stejný čas z různých míst pod stejným účtem nám indikuje možnou kompromitaci účtu. Proto je korelace velmi důležitá ve smyslu pohlížení na záznamy o událostech.

Korelační pravidla lze brát jako pomyslné filtry logů. Nastane-li určitý sled událostí za sebou a existuje na to v SIEM pravidlo, vytvoří to potom bezpečnostní incident, protože události splnili požadavky na to, aby došlo k sepnutí pravidla. Takto lze efektivně identifikovat bezpečnostní incidenty v reálném čase. Správně nastavená korelační pravidla pomáhají odlišit podstatné události od běžného provozu, což výrazně zvyšuje efektivitu bezpečnostních týmů. [30]

Jak je znázorněno na obrázku 2.1, proces filtrování logů je velmi důležitý k efektivnímu využití zdrojů a eliminující nepodstatná data. Tím poskytuje cenné informace pro administrátory a bezpečnostní analytiky.

Typická událost (log) obsahuje záznamy o činnostech systému nebo sítě. Korelační pravidla pak slouží k identifikaci vzorců, které mohou naznačovat bezpečnostní incident. Například, sériové neúspěšné pokusy o přihlášení mohou být indikátorem

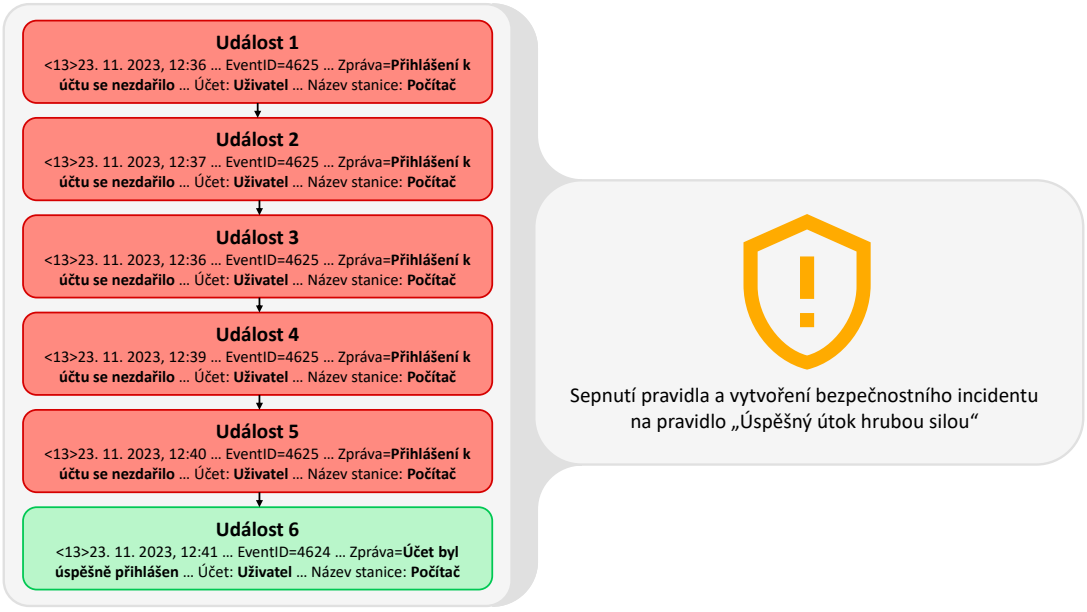
Obr. 2.1: Filtrování logů v bezpečnostním monitoringu.

pokusu o neautorizovaný přístup (viz obrázek 2.2). Vizuální nástroje pro analýzu korelačních pravidel pomáhají bezpečnostním týmům rychle identifikovat a řešit potenciální hrozby.

Proces tvorby korelačních pravidel začíná analýzou bezpečnostních požadavků a identifikací relevantních datových zdrojů. Následně se definují logická kritéria, která specifikují, co má být detekováno. Tato pravidla jsou pak testována a laděna pro zajištění přesnosti a minimalizace falešných poplachů. Iterativní vývoj a průběžná údržba jsou nezbytné pro zajištění, že pravidla zůstávají efektivní v proměnlivém bezpečnostním prostředí. [30]

2.2.1 Principy a proces tvorby pravidel

Proces tvorby korelačních pravidel je klíčovým prvkem pro zajištění bezpečnosti informačních systémů a sítí. Tento proces je komplexní a vyžaduje hluboké porozumění hrozbám, kterým organizace mohou čelit, a schopnost identifikovat a monitorovat klíčové oblasti. V procesu tvorby korelačních pravidel je několik důležitých kroků a metod, které je třeba dodržet, počínaje analýzou bezpečnostních požadavků až



Obr. 2.2: Příklad identifikace vzorce korelačním pravidlem.

po iterativní vývoj a neustálé zlepšování pravidel. Níže je vypsáno několik hlavních bodů na které je třeba dbát při tvorbě korelačních pravidel. [30]

1. **Analýza bezpečnostních požadavků** – Základem procesu je porozumění hrozbám a rizikům, kterým organizace mohou čelit. Toto zahrnuje určení klíčových oblastí monitorování a identifikaci specifických bezpečnostních potřeb.
2. **Identifikace relevantních zdrojů logů** – Důležitým krokem je určení, které logy a události (například ze serverů, síťových zařízení, aplikací) jsou relevantní pro vytváření pravidel. Tento výběr je zásadní pro efektivitu detekce.
3. **Definice logických kritérií** – Na základě identifikovaných potřeb a zdrojů logů jsou definována logická kritéria pro korelační pravidla. Toto zahrnuje specifikaci podmínek, které musí událost splnit, aby byla považována za podezřelou nebo škodlivou.
4. **Testování a ladění** – Po vytvoření pravidel následuje fáze testování a ladění. V této fázi se ověřuje, zda pravidla správně identifikují hrozby, a zároveň se ladí pro minimalizaci falešných pozitiv a negativ.
5. **Iterativní vývoj** – Proces tvorby pravidel je iterativní. Korelační pravidla je zapotřebí průběžně aktualizovat a zlepšovat tak, aby byly schopny reagovat na nové hrozby a změny v infrastruktuře. Toto zajišťuje, že pravidla zůstávají relevantní a efektivní.

Z takových kroků lze vidět, že tvorba korelačních pravidel není jednorázovým úkolem, ale neustálým cyklem analýzy, vývoje, testování a aktualizací, který je nezbytný pro udržení bezpečnosti v rychle se měnícím kybernetickém světě. [30]

2.2.2 Koncept generalizace

Generalizace v korelačních pravidlech umožňuje jejich širší uplatnění. Sigma pravidla [31] jsou příkladem generalizovaného formátu pro popis pravidel, která lze uplatnit v různých prostředích. Jsou navržena tak, aby byla snadno čitelná a přenositelná mezi různými systémy. Tento formát umožňuje bezpečnostním analytikům sdílet a implementovat detekční pravidla bez nutnosti znalosti konkrétní syntaxe dotazovacího jazyka používaného v konkrétním SIEM systému. [31]

Hlavním účelem Sigma pravidel je usnadnit sdílení, revizi a implementaci bezpečnostních detekčních pravidel. Díky své standardizované a snadno přenositelné formě mohou být tato pravidla rychle rozšířena mezi různé organizace a bezpečnostní komunity, což zvyšuje celkovou ochranu proti kybernetickým hrozbám.

Sigma pravidla jsou definována pomocí YAML syntaxe, což je snadno čitelný datový serializační formát. Struktura pravidla obvykle zahrnuje identifikaci, popis, detekční logiku a další relevantní metadata. Tato pravidla jsou navržena tak, aby mohla popsat i složité detekční scénáře, jako jsou pokročilé trvalé hrozby (APT[1]) nebo sofistikované kybernetické útoky. [31]

Sigma pravidla mohou být použita k detekci řady běžných kybernetických hrozeb, jako je ransomware, phishing, neautorizovaný přístup, anomální síťová aktivita a další. Příklady pravidel zahrnují specifikace pro identifikaci podezřelého chování, jako jsou neobvyklé přihlašovací pokusy nebo nečekané změny v konfiguraci systému. [31]

Pro práci se Sigma pravidly existuje řada nástrojů. Mezi nejpopulárnější patří GitHub repozitáře[2], kde komunita sdílí a aktualizuje Sigma pravidla. Další nástroje zahrnují konverzní nástroje[3], které umožňují převod Sigma pravidel do formátu specifického pro konkrétní SIEM systémy. [31]

2.3 Výzvy správy a udržování korelačních pravidel

Jak bylo již naznačeno v podkapitole 2.2.1, korelační pravidla a jejich správa a udržování není jednoduchým krokem. V současném dynamickém kybernetickém prostředí je správa a udržování korelačních pravidel stále složitějším úkolem. Vývoj nových technologií a metod kybernetických útoků neustále posouvá hranice toho, co je třeba zabezpečit. Aby bylo možné udržet krok s těmito změnami, je klíčové, aby byla korelační pravidla průběžně aktualizována a přizpůsobena novým hrozbám a proměnlivému prostředí. Integrace těchto pravidel do širší bezpečnostní infrastruktury

[1]Více o APT lze dočíst na odkaze <https://www.cisa.gov/topics/cyber-threats-and-advisories/advanced-persistent-threats-and-nation-state-actors>.

[2]Sigma GitHub repozitář lze nalézt na odkaze <https://github.com/SigmaHQ/sigma>.

[3]Sigma konverzní nástroj lze nalézt na odkaze <https://github.com/SigmaHQ/pySigma>.

a jejich koordinace s ostatními bezpečnostními systémy je zásadní pro vytvoření komplexní a efektivní strategie kybernetické obrany.

Výzvy, které se vyskytují při správě a udržování korelačních pravidel zahrnují nejen technické aspekty, ale i výzvy spojené s řízením falešně pozitivních a negativních výsledků, jakožto i nutnost udržovat pravidla aktuální v neustále se měnícím bezpečnostním prostředí. Níže je vypsáno několik hlavních bodů na které je třeba dbát při správě a udržování korelačních pravidel. [32]

- **Složitost a dynamika bezpečnostních hrozeb** – Bezpečnostní hrozby se neustále vyvíjí, což vyžaduje pravidelné aktualizace korelačních pravidel, aby byly schopny čelit novým hrozbám. Tyto neustálé změny pravidel mohou být složité a časově náročné.
- **Vysoký počet falešně pozitivních výsledků** – Korelační pravidla mohou generovat velké množství falešně pozitivních výsledků, což znamená, že běžné nebo neškodné aktivity jsou nesprávně identifikovány jako bezpečnostní hrozby. To může vést k velkému vytížení bezpečnostních analytiků a ztrátě důvěry v efektivitu pravidel.
- **Zajištění kompatibility a integrace** – Korelační pravidla musí být kompatibilní s různými formáty logů a SIEM systémy. Udržování kompatibility a zajištění správné integrace s různými zdroji dat a systémy může být občas výzvou.
- **Správa a optimalizace výkonu** – Výkon SIEM systému může být negativně ovlivněn přílišným počtem nebo příliš složitými korelačními pravidly. Optimalizace výkonu při zachování účinnosti detekce je klíčová.
- **Nedostatek kvalifikovaných odborníků** – Správa korelačních pravidel vyžaduje kvalifikované bezpečnostní analytiky, kteří rozumí aktuálním hrozbám a mají znalosti specifické pro dané prostředí. Nedostatek takových odborníků může být významnou překážkou.
- **Dodržování právních a regulačních požadavků** – Korelační pravidla musí být v souladu s právními a regulačními požadavky, jako jsou GDPR, HIPAA atd. Udržování tohoto souladu při zajištění efektivní detekce hrozeb je není zcela jednoduché.
- **Udržitelnost a škálovatelnost** – Jak organizace roste, její potřeby se mění, což znamená, že korelační pravidla musí být pravidelně revidována a upravována tak, aby odpovídala rostoucímu množství dat a měnícím se bezpečnostním požadavkům. [32]

3 Integrace umělé inteligence do tvorby korelačních pravidel

V této kapitole je probráno, jaká jsou možná nejvhodnější místa pro integraci umělé inteligence do procesu tvorby, úprav a optimalizací korelačních pravidel v oblasti bezpečnostního monitoringu. Umělá inteligence totiž nabízí řadu možností, jak tyto procesy zefektivnit, zvýšit jejich kvalitu a rychleji reagovat na případné bezpečnostní hrozby.

Cílem této kapitoly je ukázat, jak lze umělou inteligenci zapojit do tvorby korelačních pravidel a jak to může pomoci zvýšit efektivitu, přesnost a kvalitu bezpečnostního monitoringu.

V podkapitolách je podrobněji probráno, jak lze umělou inteligenci využít při tvorbě nových korelačních pravidel (podkapitola 3.1), jak s její pomocí upravovat stávající pravidla (podkapitola 3.2) a jak může pomoci lépe porozumět těmto pravidlům v roli zkušenějšího analytika (podkapitola 3.3). Informace a poznatky popsány níže byly pečlivě sestaveny na základě konzultací s bezpečnostními analytiky společnosti Aricoma.

3.1 Tvorba nových korelačních pravidel

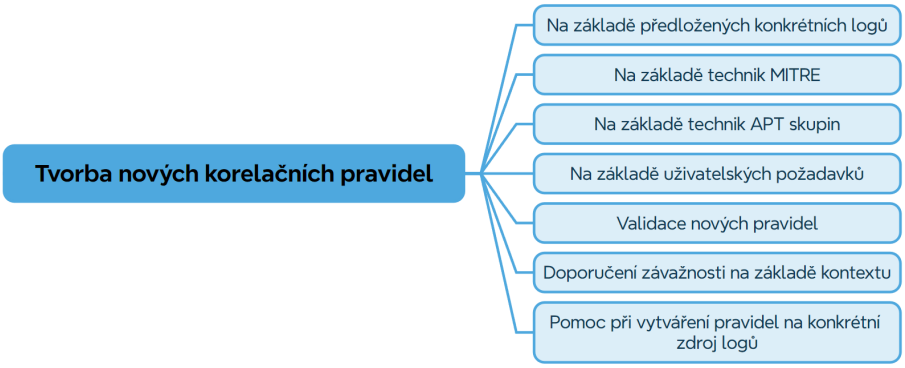
Umělá inteligence dokáže zpracovat obrovské množství dat z různých zdrojů, jako jsou záznamy událostí nebo informace o hrozbách. Díky tomu je schopna odhalit nové vzorce chování, neobvyklé aktivity a potenciální hrozby, které by člověk mohl snadno přehlédnout. Na základě těchto poznatků dokáže navrhnout nová pravidla, která se zaměří právě na tyto hrozby.

Nejde jen o analýzu dat. Umělá inteligence umí využít různé techniky a postupy pro tvorbu těchto pravidel. Může například pracovat s MITRE ATT&CK frameworkem [33], což je komplexní přehled taktik, technik a postupů používaných kybernetickými útočníky. Umělá inteligence dokáže analyzovat tyto postupy a navrhnout pravidla, která je odhalí. Stejně tak může využít informace o známých skupinách pokročilých hrozeb (APT) a jejich typickém chování, aby vytvořila pravidla zaměřená na jejich aktivity. [33, 34]

Dokonce může reagovat na konkrétní požadavky uživatelů a vytvořit pravidla na míru, která odpovídají jejich potřebám a prioritám. Po vytvoření nových pravidel umělá inteligence pomůže s jejich ověřením a určením jejich závažnosti. Může simulovat různé scénáře a testovat, jak pravidla reagují na různé typy událostí. Tím se odhalí případné chyby nebo nedostatky, které se opraví ještě předtím, než se pravidla začnou používat. Dále je vhodným místem uplatnění určení závažnosti nových

pravidel. Na základě analýzy potenciálních dopadů hrozeb a rizik může doporučit, jakou prioritu pravidlu dát a jak ho zařadit do celkového bezpečnostního systému. Tím pomáhá efektivněji řídit bezpečnostní rizika a efektivně využívat zdroje.

Využití umělé inteligence pro tvorbu nových korelačních pravidel by mohlo být tedy slibným krokem vpřed v kybernetické bezpečnosti. Popsané způsoby možné integrace do tvorby nových korelačních pravidel lze také vidět na myšlenkové mapě na obrázku 3.1. [35, 36]



Obr. 3.1: Využití AI při tvorbě korelačních pravidel.

3.2 Úprava a optimalizace existujících korelačních pravidel

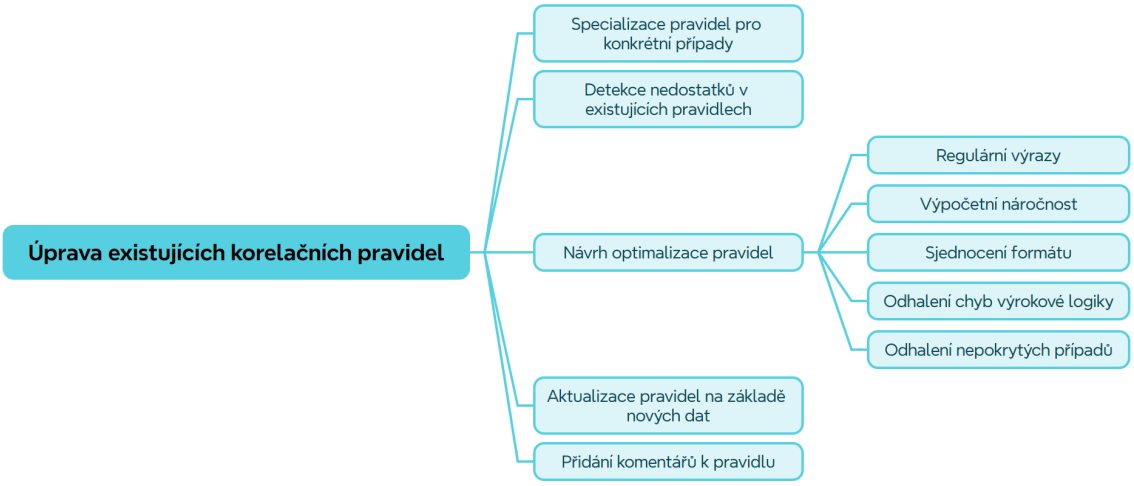
Dalším vhodnou oblastí pro integraci umělé inteligence je analýza a vylepšování stávajících korelačních pravidel. S její pomocí lze tato pravidla přizpůsobit konkrétním potřebám organizace a jejím specifickým hrozbám. Může například pomoci vytvořit pravidla zaměřená na útoky na určité systémy, aplikace, nebo sektory ve kterých se společnosti nachází.

Dokáže také analyzovat stávající pravidla a najít jejich slabá místa, jako jsou například mezery v pokrytí určitých typů útoků, neefektivní použití výrazů nebo vysoká náročnost na výpočetní výkon a na základě této analýzy navrhnout konkrétní vylepšení, která zvýší výkon a efektivitu pravidel. Může jít například o úpravu výrazů pro přesnější detekci, snížení výpočetní náročnosti pro rychlejší zpracování nebo sjednocení formátu pravidel pro lepší správu. Důležitou roli hraje i v průběžné aktualizaci pravidel na základě nových dat a informací o hrozbách. Díky tomu lze během chvíle aktualizovat stávající pravidla a přizpůsobit je novým datům ta, aby byla aktuální a relevantní.

Umělá inteligence také může být vhodná pro přidávání komentářů a vysvětlení k pravidlům, což usnadní jejich pochopení a údržbu. Komentáře mohou poskytnout

více kontextu a vysvětlení, proč bylo pravidlo vytvořeno a jak funguje, což je užitečné zejména pro nové členy týmu nebo při předávání znalostí. Další výhodou je možnost sjednocení formátu korelačních pravidel a zajištění jejich konzistence. To usnadňuje správu a údržbu, zejména ve větších společnostech s mnoha pravidly. Sjednocení formátu také umožňuje snadnější automatizaci a propojení s dalšími nástroji a systémy. Dalším místem integrace by mohlo být při odhalování chyb v logice pravidel, které by mohly vést k falešným poplachům nebo přehlédnutí skutečných hrozeb. Dokáže analyzovat logiku pravidel a najít potenciální konflikty nebo nejasnosti, což umožňuje opravit chyby a zlepšit přesnost detekce.

Využití umělé inteligence pro úpravu a optimalizaci stávajících korelačních pravidel je tedy dalším vhodným bodem pro realizaci k posílení kybernetické bezpečnosti. Shrnutí bodů popsaných výše lze vidět na myšlenkové mapě na obrázku 3.2. [35, 36]



Obr. 3.2: Využití AI při úpravách existujících korelačních pravidel.

3.3 Vysvětlení a interpretace korelačních pravidel

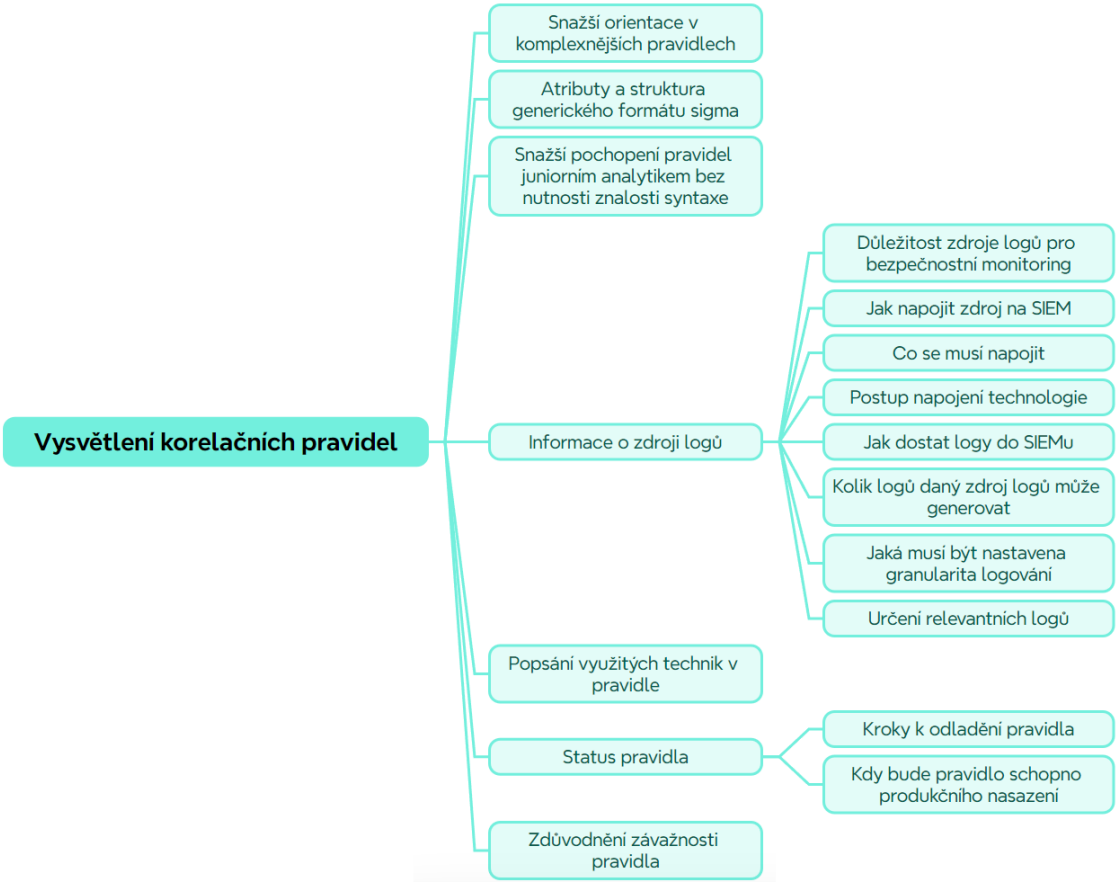
Umělá inteligence by mohla být vhodná pro zjednodušení pochopení pravidel, protože dokáže analyzovat i složitá korelační pravidla, což analytikům pomůže lépe porozumět jejich struktuře a logice. Snadno tak identifikují klíčové vlastnosti a podmínky, které vedou k odhalení bezpečnostních událostí. Tento přístup by mohl být obzvláště užitečný při práci s rozsáhlými a složitými pravidly, kde manuální analýza může být zdlouhavá a neúspěšná.

Jednou z hlavních výhod umělé inteligence je možnost interpretace korelačních pravidel bez nutnosti hluboké znalosti specifického jazyka bezpečnostním analytikem. Dokáže interpretovat pravidla do běžné řeči, takže analytici rychle pochopí

jejich význam a dopad. Díky tomu se znalost korelačních pravidel zpřístupní širšímu okruhu i tolik nezkušených analytiků.

Umělá inteligence dokáže nejen vysvětlit korelační pravidla, ale také poskytuje důležité informace o zdrojích dat, které jsou pro dané pravidlo důležité. Může například určit, jak jsou jednotlivé zdroje logů důležité pro bezpečnostní sledování, popsat jak je připojit k systému SIEM, a vysvětlit jak detailní záznamy jsou potřeba. Tyto informace jsou totiž klíčové pro efektivní správu a údržbu korelačních pravidel. Dále by bylo možné popsat a vysvětlit konkrétní techniky použité v korelačním pravidle. Může například určit typ útoku, který pravidlo odhaluje, nebo popsat specifické indikátory kompromitace (IoC).

Nakonec by mohla být využita i k určení, jak je korelační pravidlo závažné. Na základě analýzy možných dopadů odhalených hrozeb a rizik může doporučit, jakou prioritu pravidlu dát a jak ho hierarchicky zařadit do celkového bezpečnostního systému. Shrnutí výše popsaných bodů lze vidět na myšlenkové mapě na obrázku 3.3. [35, 36]



Obr. 3.3: Využití AI pro usnadnění pochopení korelačních pravidel.

4 Návrh webové aplikace

V rámci této kapitoly bude navržena webová aplikace, jejímž cílem je integrovat do sebe umělou inteligenci za účelem zefektivnění vývoje korelačních pravidel pro SIEM platformy. Kapitola je strukturována do několika částí, zahrnujících cíle a požadavky, volbu technologií, moderní architekturu mikroservis a integraci s externími API. Každá sekce je zaměřena na specifické aspekty vývoje a implementace aplikace.

4.1 Cíle a požadavky kladené na aplikaci

Hlavním cílem aplikace je poskytnout bezpečnostním analytikům intuitivní nástroj pro tvorbu a správu korelačních pravidel, minimalizující potřebu dalších pomocných nástrojů. Jak bylo zmíněno v kapitole 2.2.1, proces tvorby korelačních pravidel může být pro nezkušené analytiky obtížný. Aplikace by měla sloužit k usnadnění a správy a vývoje korelačních pravidel prostřednictvím schopností umělé inteligence. Výsledný nástroj by měl být schopný anonymizovat data posílaná na serverovou část aplikace, aby nedošlo k úniku citlivých dat a tím se tedy zajistila důvěra v systém. Uživatel by měl být rovněž schopný vidět z klientské části aplikace, jaká data budou anonymizovaná předtím, než odešle požadavek ke zpracování umělé inteligenci. V případě, že by uživatel zapomněl anonymizovat svoje data, měla by být naimplementovaná ochrana, která ho na danou skutečnost upozorní. Dále je krokem který nelze opomenout dostupnost, která by měla být zajištěna implementací a napojením více generativních modelů. Při úpravách korelačních pravidel by potom mělo být jasné a lehce pochopitelné, k jakým úpravám došlo. V aplikaci by mělo být také možno specifikovat vlastní sadu otázek, které budou nápomocné uživateli při vyhledávání a dotazování se na nová pravidla, které si bude vybírat z veřejné databáze pravidel. Za cíl se klade stav, kdy uživatel klikne jen na jedno tlačítko a budou mu k pravidlu zodpovězeny veškeré otázky, které ho zajímají, které budou obohaceny případně o doplňující relevantní otázky, které by ho nadále mohly zajímat.

4.2 Architektura moderních webových aplikací

Architektura mikroservis je přístup k organizaci softwarového vývoje, kde je software poskládán z malých nezávislých služeb, které komunikují prostřednictvím definovaných API. To umožňuje správu služeb malými samostatnými týmy.

Architektura mikroslužeb usnadňuje škálování aplikací a urychluje vývoj, což podporuje inovace a zkracuje dobu uvedení nových funkcí na trh.

Srovnání mikroservisní a monolitické architektury

V monolitických architekturách jsou všechny procesy úzce propojeny a běží jako jedna služba. To znamená, že pokud jeden proces aplikace zažívá nárůst poptávky, musí být celá architektura škálována. Přidávání nebo zlepšování funkcí monolitické aplikace se stává složitější, jak roste obsah kódu. Tato složitost omezuje experimentování a ztěžuje implementaci nových funkcí. Monolitické architektury zvyšují riziko pro nedostupnost aplikace, protože mnoho závislých a úzce propojených procesů zvyšuje dopad selhání jednoho procesu.

U architektury mikroslužeb je aplikace postavena jako nezávislé komponenty, které spouštějí každý proces aplikace jako službu. Tyto služby komunikují prostřednictvím dobře definovaného rozhraní pomocí API. Každá služba plní jednu funkci. Protože jsou provozovány nezávisle, každá služba může být aktualizována, nasazena a škálována dle požadavků konkrétních funkcí aplikace. [37]

4.3 Volba technologií a funkcionalit

V rámci této kapitoly bude představena serverová část aplikace (podkapitola 4.3.1), klientská část aplikace (podkapitola 4.3.2) a nakonec komunikace těchto dvou částí (podkapitola 4.3.2)

4.3.1 Serverová část

Pro vývoj backendové části aplikace je zvolen programovací jazyk Python spolu s technologiemi Docker a Flask. Python, jakožto výkonný a flexibilní programovací jazyk, je vybrán pro svou snadnou čitelnost, rozsáhlou knihovnu standardních modulů a schopnost efektivně zvládat různé backendové úlohy.

Docker (*kontejnerizační platforma*) slouží pro zajištění konzistentního a izolovaného prostředí pro aplikace. Využití Dockeru umožňuje snadné nasazení a škálování aplikace, nezávisle na prostředí, ve kterém běží. Kontejnery Dockeru zajišťují, že aplikace bude mít stejné chování na různých systémech, což předchází častému problému zvanému „funguje na mém počítači“.

Flask je mikroframework pro webové aplikace napsaný v Pythonu, je implementován kvůli své jednoduchosti a schopnosti rychle vytvářet prototypy. Flask poskytuje základní nástroje a knihovny potřebné pro vytváření webových aplikací, jako jsou routování URL, požadavky a odpovědi, a integrační vrstvy pro databáze. Jeho modulární povaha umožňuje snadnou integraci s řadou rozšíření, která mohou rozšířit jeho funkčnost podle potřeb projektu.

Pro vývoj backendu je tedy využit Python, Docker a Flask. Tyto technologie jsou charakterizovány pro svou vysokou efektivitu, škálovatelnost a flexibilitu, což

přispívá k rychlému vývoji, snadnému nasazení a udržitelnosti aplikace v různých prostředích. [38, 39, 40]

4.3.2 Klientská část

Pro vývoj frontendové části aplikace je zvolena technologie Vue 3, což je moderní JavaScriptový framework. Vue 3 je preferován pro jeho reaktivitu a kompoziční API, které umožňuje vytvářet dynamické uživatelské rozhraní s vysokou mírou přizpůsobivosti. Komponenty jsou v tomto frameworku definovány s využitím jednotného souborového formátu, který integruje HTML, CSS a JavaScript, čímž se zvyšuje efektivita vývoje a udržitelnost kódu.

V rámci Vue 3 je také intenzivně využíván systém reaktivního datového vazby, který zajišťuje, že změny stavu aplikace jsou automaticky reflektovány v uživatelském rozhraní. Tento mechanismus je implementován s využitím proxy objektů v JavaScriptu, což vede k efektivnímu a intuitivnímu způsobu správy stavu aplikace.

Vcelku je tato technologie charakterizována vysokou úrovní modularity, reaktivity a efektivní správou stavů, což přispívá k rychlému vývoji, vysoké udržitelnosti kódu a přívětivé uživatelské zkušenosti. [41]

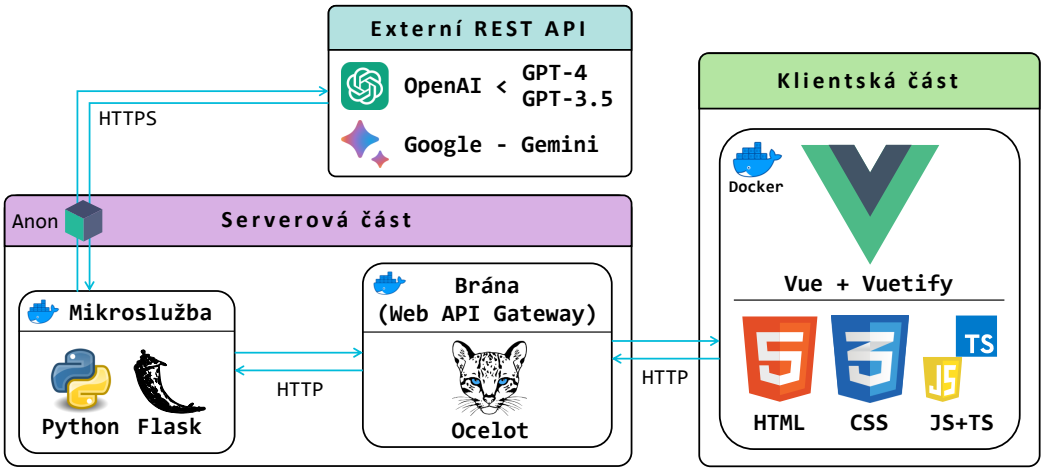
Komunikace frontend-backend

Pro komunikaci mezi frontendem a backendem aplikace bylo zvoleno řešení Ocelot Web API Gateway. Ocelot, jakožto snadná API brána, je zvolen pro jeho schopnost poskytovat jednotný vstupní bod pro všechny požadavky směřující do aplikace, čímž se zjednodušuje routování a zabezpečení.

API Gateway Ocelot je využíván pro integraci více mikroslužeb do jednoho rozhraní, což umožňuje frontendu komunikovat s různými backendovými službami prostřednictvím jediného konzistentního rozhraní. Tato centralizace vede k lepší správě a monitorování provozu, a také umožňuje efektivnější zabezpečení komunikace pomocí jednotných pravidel a politik. V rámci Ocelotu jsou implementovány funkce jako jsou load balancing, který rozděluje zátěž mezi různé instance služeb, a routing, který zajišťuje přesměrování požadavků na správné služby. Tyto vlastnosti jsou zásadní pro udržení vysoké dostupnosti a spolehlivosti aplikace, zvláště v prostředí s vysokým objemem uživatelských požadavků. Dále je Ocelot konfigurován tak, aby poskytoval podporu pro autentizaci a autorizaci. Tím je zajištěno, že všechny požadavky procházejí nezbytnými bezpečnostními kontrolami předtím, než dosáhnou backendových služeb. Toto přispívá k celkovému zabezpečení aplikace tím, že se zabrání neoprávněnému přístupu k citlivým datům a funkcím.

Ocelot Web API Gateway tedy přináší značné výhody v podobě zjednodušené a bezpečné komunikace mezi frontendem a backendem, lepší správy a monitorování služeb, a poskytuje robustní základ pro škálování a rozšiřování aplikace v budoucnu. [42]

Architektura a technologie, které byly popsány a zvoleny výše lze shrnout do schématu architektury zobrazené na obrázku 4.1.



Obr. 4.1: Architektura webové aplikace.

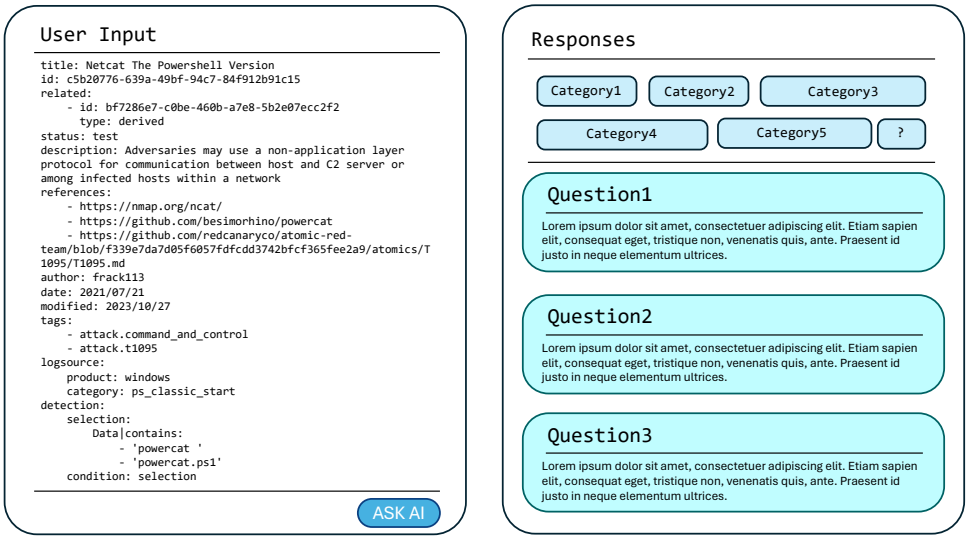
4.3.3 Návrh uživatelského rozhraní

Koncept uživatelského rozhraní, znázorněný na obrázku 4.2, byl navržen s důrazem na intuitivnost a snadnou ovladatelnost. Při tvorbě rozhraní byly zohledněny zavedené principy designu, jako je vizuální hierarchie, konzistence a jasná zpětná vazba.

Barevná paleta byla zvolena tak, aby byla příjemná pro oko a zároveň podporovala čitelnost textu a ikon. Dominantní barvy jsou neutrální, s akcenty pro zvýraznění důležitých prvků. Typografie byla pečlivě vybrána s ohledem na čitelnost a estetiku a rozvržení prvků bylo navrženo tak, aby bylo logické a předvídatelné.

4.3.4 Případy užití aplikace

K zajištění maximální dostupnosti je navrženo integrovat do aplikace více generativních modelů umělé inteligence. Konkrétně je v plánu integrovat modely od společnosti OpenAI a Google a umožnit uživateli volbu modelu, který bude nejvíce vyhovovat jeho potřebám a preferencím. S ohledem na citlivost dat zpracovávaných v rámci korelačních pravidel bude implementován anonymizační modul. Tento



Obr. 4.2: Návrh rozložení webové aplikace.

modul bude uživateli umožňovat anonymizovat citlivé údaje před jejich odesláním ke zpracování generativním modelům, čímž bude zajištěna ochrana před únikem citlivých údajů. Pro uživatelský komfort a kontrolu procesu anonymizace bude implementováno uživatelské rozhraní, které bude uživateli umožňovat zapnout nebo vypnout anonymizační modul podle potřeby. Toto rozhraní bude také vizuálně zvýrazňovat anonymizovaná data, což přispěje k lepší přehlednosti a ujištění uživatele o anonymizaci dat.

Dle rozvrhu vhodných míst implementace v kapitole 3 bylo navrženo uživatelské rozhraní ve třech částech.

První část se bude specializovat na tvorbu nových korelačních pravidel a bude uživateli umožňovat zadat požadavek na vytvoření nového korelačního pravidla, včetně výběru preferovaného modelu, formátu, jazyka a dalších parametrů. Na základě těchto vstupů generativní model vytvoří nové pravidlo, které splňuje zadané požadavky.

Návrh druhé části spočívá ve vysvětlení korelačních pravidel, což bude uživateli umožňovat klást dotazy týkající se vybraného korelačního pravidla. Uživatel bude moci využít předdefinovanou sadu otázek nebo klást vlastní dotazy. Generativní model poskytne srozumitelné a stručné odpovědi, které pomohou uživateli lépe porozumět pravidlu a jeho aplikaci.

Poslední část se bude specializovat na úpravu existujících korelačních pravidel a bude uživateli umožňovat upravovat stávající korelační pravidla podle svých potřeb. Uživatel bude moci specifikovat požadované změny, které následně budou pro-

vedeny generativním modelem. Výsledné upravené pravidlo bude vizuálně zvýrazněno, aby uživatel mohl jasně a snadno identifikovat provedené změny. S tím souvisí i návrh pokročilého tlačítka pro kopírování. Pro usnadnění práce s upravenými korelačními pravidly bylo navrženo pokročilé tlačítko pro kopírování, které bude umožňovat zkopírovat upravené pravidlo vyjma odstraněných řádků. Tato funkce bude zvyšovat efektivitu a uživatelský komfort při práci s pravidly.

Pro zajištění kvality a přesnosti výstupů generovaných umělou inteligencí je v plánu implementovat kontrolní mechanismy. Tyto mechanismy by měly být schopny lépe vyhodnocovat a filtrovat výstupy, což povede ke zvýšení počtu relevantních a spolehlivých výsledků.

5 Implementace webové aplikace

Realizace implementace webové aplikace byla provedena s důrazem na využití technologií, které byly detailně rozebrány v kapitole o návrhu aplikace v kapitole 4.

Proces tvorby zahrnoval detailní práci s vybranými programovacími jazyky (*python, typescript, javascript*), frameworky a nástroji, jejichž kombinace byla navržena tak, aby odpovídala specifickým požadavkům projektu.

Frontend byl vytvořen s využitím moderních technologií pro dynamické uživatelské rozhraní, zatímco backend byl implementován s ohledem na robustnost, bezpečnost a škálovatelnost.

Důležitým aspektem implementace bylo také zajištění plynulé komunikace mezi různými komponentami systému. Tato komunikace byla usnadněna prostřednictvím efektivně navrženého API, což umožňovalo spolehlivý a bezpečný přenos dat mezi frontendem a backendem.

Celkově byla implementace webové aplikace realizována s dodržením návrhových technologií, což zajistilo, že konečný produkt nejen splňuje všechny požadované funkce, ale je také technicky odolný a připravený na budoucí rozšíření a úpravy.

5.1 První fáze implementace

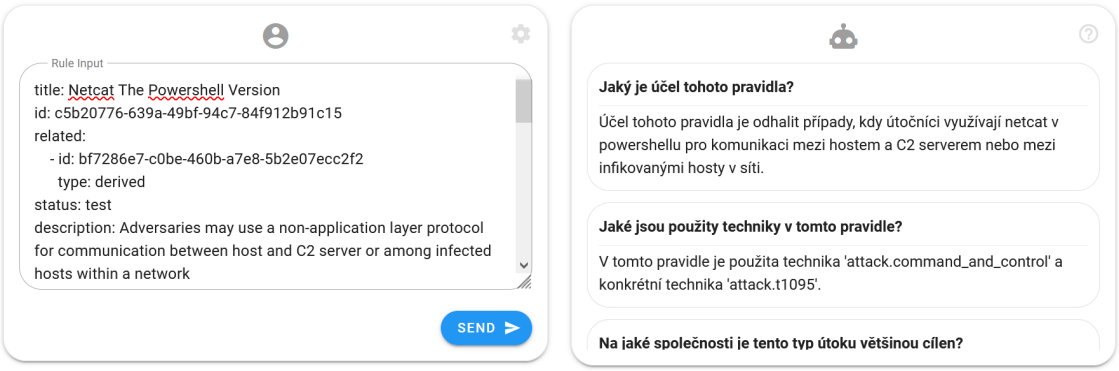
V první fázi implementace bylo naimplementováno několik následujících funkcionalit dle návrhu, který byl proveden v kapitole 4.3, které jsou navrženy tak, aby byly snadno použitelné a intuitivní, umožňující uživatelům aplikovat pokročilé techniky a analýzu umělé inteligence na korelační pravidla bez potřeby rozsáhlých technických znalostí. Výstup umělé inteligence je poté prezentován způsobem, který je informativní, což uživatelům umožňuje efektivně využít aplikaci pro získání požadovaných informací. Níže je vypsáno několik funkcionalit, které uživatelské rozhraní obsahuje. Vizuálně je lze vidět na obrázku 5.3.

- **Zadávání pravidel** – Sekce `Rule Input` poskytuje prostor pro zadání konkrétních pravidel pro následné zpracování a analýzu pravidel umělou inteligencí.
- **Výběr generativního modelu** – Uživateli je umožněno vybrat mezi různými verzemi umělé inteligence. Tato volba ovlivňuje, jaký typ generativního předtrénovaného modelu bude použit pro zpracování dotazů a vstupů.
- **Stanovení formátu pravidla** – Uživatel má možnost vybrat formát pravidla, který bude vkládat. Přičemž jsou k dispozici možnosti `Sigma` a `Netwitness EPL`. Výběr formátu určuje syntaxi a strukturu pravidel, která budou použita pro analýzu umělou inteligencí.

- **Nastavení jazyka** – Uživatel má možnost zvolit preferovaný jazyk, ve kterém bude komunikovat s aplikací, s volbami `Cesky` nebo `Anglicky`. Toto nastavení přizpůsobuje výstup umělé inteligence preferovanému jazyku uživatele.
- **Předdefinované sady otázek** – Uživatelské nastavení obsahuje přepínač, který umožňuje uživateli rozhodnout, zda chce používat předdefinovanou sadu otázek pro interakci s umělou inteligencí (viz obrázek 5.1), nebo zda chce klást otázky vlastní (viz obrázek 5.2).
- **Vkládání uživatelských dotazů** – V poli pro otázky má uživatel možnost vložit vlastní dotaz k pravidlu, na který umělá inteligence poskytne odpověď.

Důležitou funkcionalitou aplikace byl však ještě anonymizační modul[1], který je důležitou částí k zajištění důvěry v systém a zabránění úniku citlivých dat. Zpracováváme-li citlivé údaje zákazníků, je naší povinností, aby nebyla data odcizena nebo se k nim nedostal někdo neoprávněný, proto dle obecného nařízení o ochraně osobních údajů GDPR máme povinnost citlivá data anonymizovat takovým způsobem, že již nebude možno identifikovat komu patří a nebudou tak považovány za citlivé údaje.

Modul byl však v první fázi implementace naimplementovaný natvrdo na serverové části aplikace a nebylo možné ho vypnout. Rovněž byl jen jednocestný, tudíž postrádal transparentnost z pohledu uživatele. Rozšíření anonymizačního modulu k zajištění obousměrnosti a tím i transparentnosti pro uživatele bylo plánováno v druhé části implementace této práce.

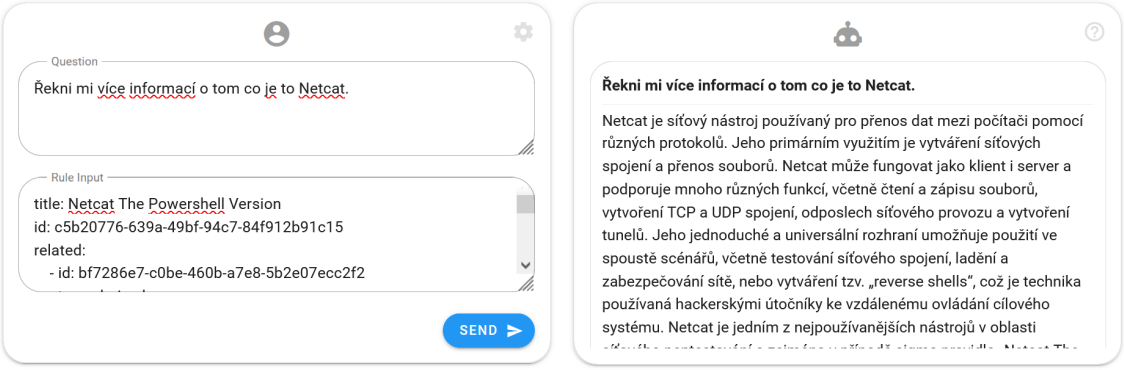


Obr. 5.1: Výchozí stav aplikace využívající předdefinovanou sadu otázek.

5.2 Druhá fáze implementace

V druhé fázi implementace bylo rozhodnuto, že se ponechá rozložení grafického rozhraní ve stejném duchu, jako bylo v první fázi implementace. Tedy jedna polovina

[1] Anonymizační modul byl převzat jako část z bakalářské práce dostupné na odkaze <`https://www.vut.cz/studenti/zav-prace/detail/151229`>.



Obr. 5.2: Využití možnosti pokládat vlastní dotazy k pravidlům.

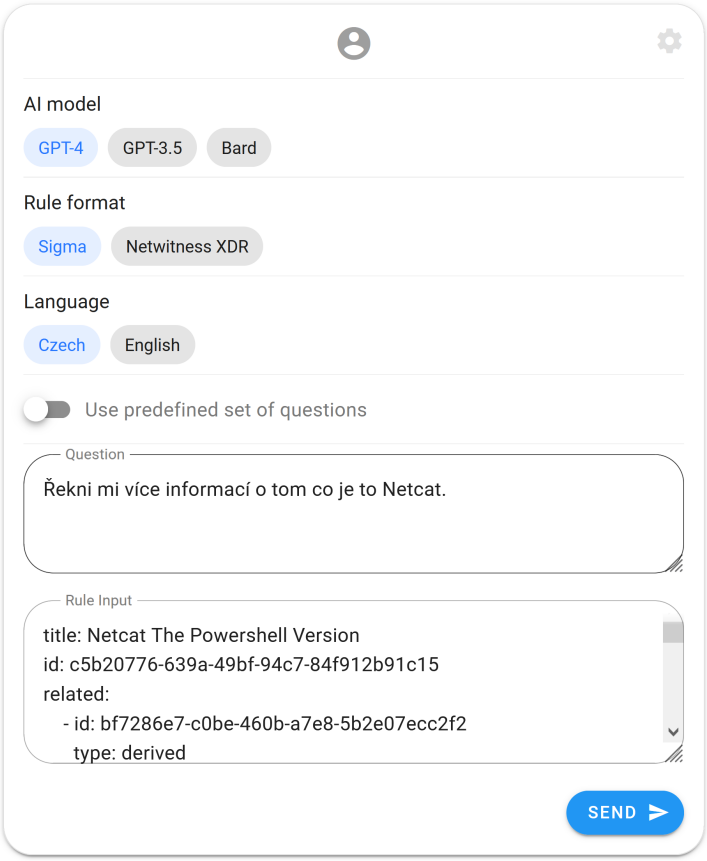
rozhraní bude sloužit pro uživatelské vstupy a druhá pro uživatelské výstupy. V první fázi implementace byl testován generativní prediktivní model Bard od společnosti Google. Ten se v první fázi implementace neosvědčil, kvůli jeho omezeným znalostem z kyberbezpečnosti. V únoru roku 2024 však společnost Google zpřístupnila veřejnosti svůj nový model Gemini, což bylo vhodnou příležitostí ho otestovat v rámci této práce. Později tohoto roku oznámila společnost OpenAI svůj nový generativní předtrénovaný model GPT-4o.

V druhé fázi implementace byly převzaty znalosti a uživatelské rozhraní z první části implementace, a bylo to překlopeno do modálního okna ve formě kompaktního tlačítka v rohu stránky, které se dynamicky zobrazí uživateli podle stránky na které se zrovna nachází, o čemž rozhoduje relevance stránky shodující se s pohledem v kapitole 3. Nový návrh rozložení lze vidět na obrázku 5.4 při zavřeném modálním okně a 5.5 při otevřeném modálním okně.

Nově má tedy uživatel na výběr ze čtyř generativních předtrénovaných modelů od dvou společností, což zajišťuje dostupnost i v případě nefunkčnosti některého z nich a odbourává to kompletní závislost na některém z nich.

V následujících podkapitolách bude probrána implementace na základě bodů vyplývajících z kapitoly 3, která bude rozdělena dle předem vymezených hranicí na 3 podkapitoly probírající tvorbu nových korelačních pravidel, vysvětlení korelačních pravidel a úprava existujících korelačních pravidel.

Dále bylo provedeno rozšíření anonymizačního modulu tak, aby zajišťoval obousměrnost a tím se stal tedy i transparentním pro uživatele. Anonymizační modul je v tomto pojetí klíčovým pro zajištění důvěry v systém a zabránění úniku citlivých dat. Anonymizace je realizována na základě regulárních výrazů a obousměrnost anonymizace a transparentnost pro uživatele byla zajištěna tím, že byl modul rozšířen o zpětný překlad na základě vyhledání pomocí stejných regulárních výrazů a ukládaným datům, které již byly anonymizovýny. Tedy v případě, že citlivý údaj byl

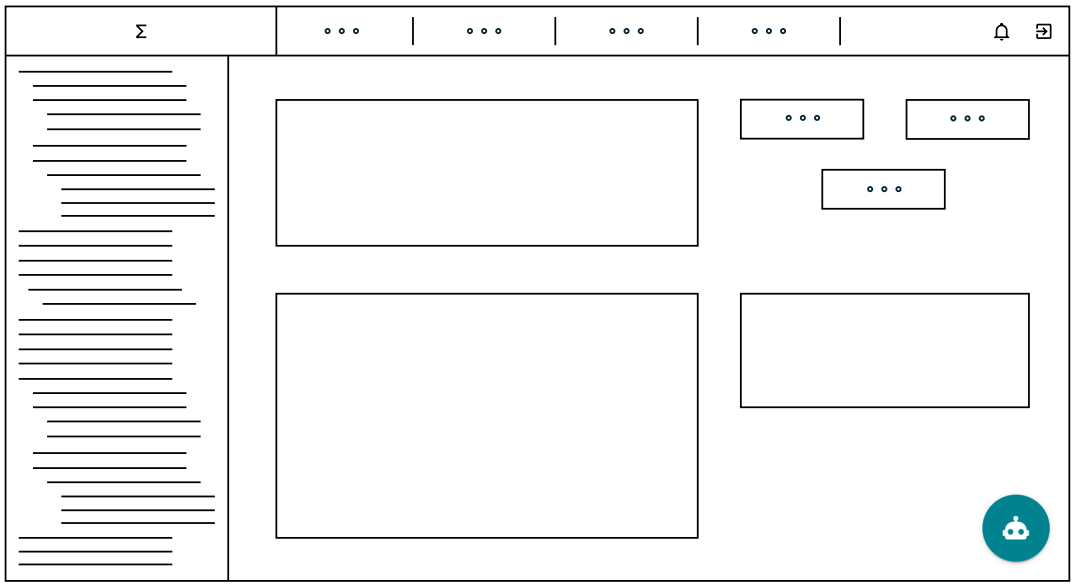


Obr. 5.3: Možné funkcionality z první fáze implementace.

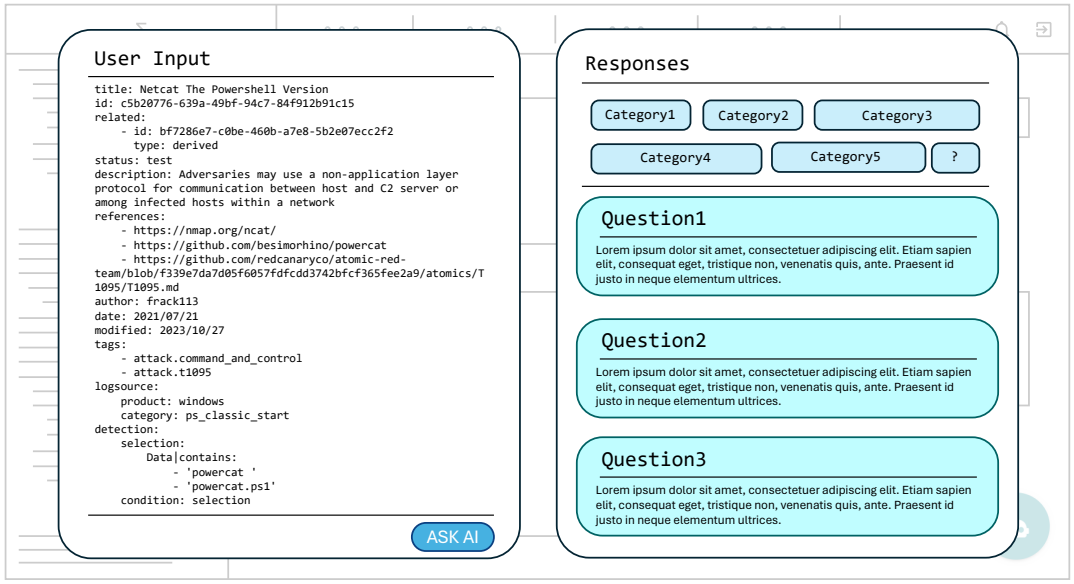
nalezen pomocí regulárních výrazů a byl tedy anonymizován a poslán na zpracování generativnímu modelu, byl rovněž uložen do listu obsahující anonymizované údaje ve své kategorii. V případě deanonymizace se provede vyhledávání údajů opět na základě regulárních výrazů a kontroluje se, zda se nejedná o údaj, který byl vytvořen v procesu anonymizace a pokud ano, udělá se zpětný překlad a data jsou úspěšně deanonymizovaná. Úspěšný příklad deanonymizace dat lze vidět na výpisech z příkazové řádky, kde lze vidět anonymizovanou odpověď ve výpisu B.2 a následně deanonymizovanou odpověď ve výpisu B.3.

Z uživatelského rozhraní však nebylo možno v rámci první části implementace volit si, zda bude anonymizace použita a zda ne. Rovněž z první části implementace nešlo rozeznat, která a zda vůbec nějaká data byla anonymizovaná. Vzhledem k výše popsaným skutečnostem bylo rozhodnuto naimplementovat na uživatelskou část aplikace přepínač, který uživateli umožní vypnout anonymizační modul na serverové části.

K zajištění důvěry v systém a pohodlí uživatele bylo rovněž naimplementováno vizuální zobrazení anonymizovaných dat v případě, že se je uživatel rozhodnul anonymizovat. Dále bylo v druhé části implementace rozhodnuto využít neoficiální API



Obr. 5.4: Návrh pohledu při zavřeném modálním okně.



Obr. 5.5: Návrh pohledu při otevřeném modálním okně.

pro propojení modelu Gemini od společnosti Google. Tím byla snížena závislost na jediném modelu a nabídnuta širší škála voleb uživateli. Důležitou komponentou při úpravách korelačních pravidel bylo tlačítko pro kopírování upravených pravidel, které zkopírovalo upravené pravidlo bez částí, které byly odebrány. Části druhé fáze implementace které byly výše popsané potom vedly k sestavení HTTP požadavku na klientské části, který lze vidět ve výpisu B.1, který byl následně poslán na serverovou část aplikace.

6 Výsledky a zhodnocení aplikace

Výsledná aplikace je předvedena níže v kapitole 6.1, ve které budou zobrazeny jednotlivé komponenty aplikace a její funkčnost. Dále v kapitole 6.2 budou shrnuty výsledky vyplývající z testování generativních modelů a využití umělé inteligence v procesu tvorby korelačních pravidel a nakonec budou rozebrány další možné kroky v rámci budoucího vývoje v kapitole 6.3.

6.1 Přehled dosažených výsledků

Přehled dosažených výsledků bylo rozhodnuto rozdělit do tří podkapitol obdobně jak tomu bylo provedeno v rámci představení oblastí vhodných pro integraci umělé inteligence do tvorby korelačních pravidel v kapitole 3. Níže lze tedy vidět rozdělení podkapitol do tvorby nových korelačních kapitol (podkapitola 6.1.1), vysvětlení korelačních pravidel při výběru z veřejné databáze (podkapitola 6.1.2) a nakonec úprava existujících korelačních pravidel (podkapitola 6.1.3). Celkový vzhled aplikace dle provedeného návrhu lze vidět na obrázcích 6.1 a 6.2.

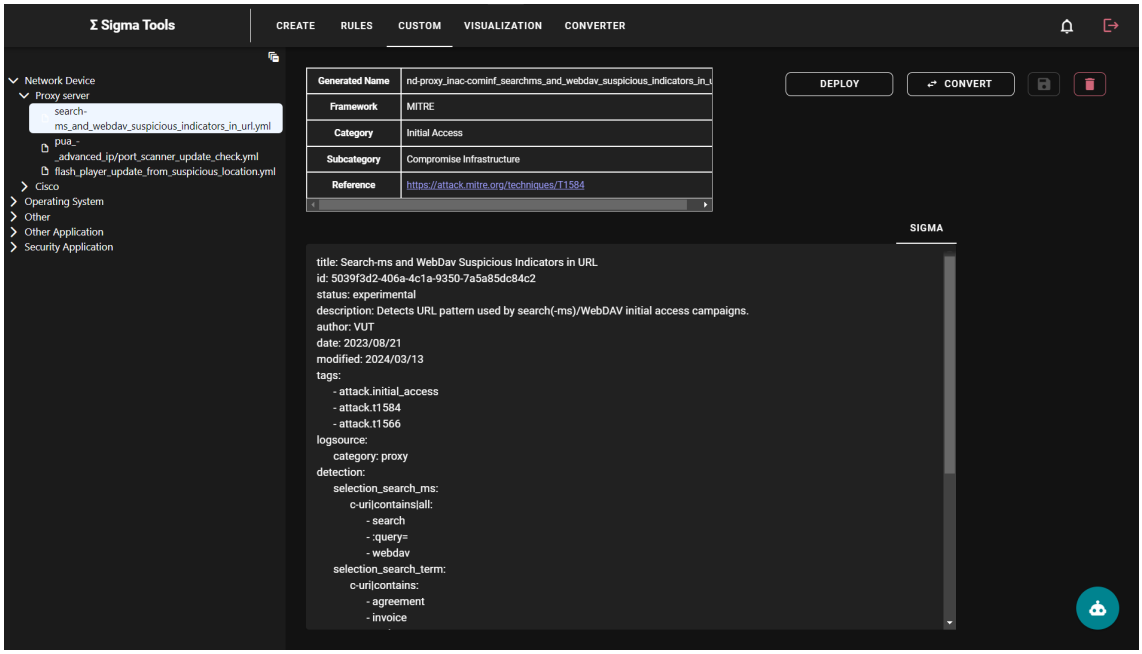
6.1.1 Tvorba nových korelačních pravidel

Při vytváření nových korelačních pravidel má uživatel možnost si přizpůsobit preference podle sebe. Při výběru generativního modelu má aktuálně na výběr celkem ze čtyř modelů, z nichž jsou 3 od společnosti OpenAI (gpt-4, gpt-4o, gpt-3.5-turbo) a jeden od společnosti Google (gemini). Dále má na výběr v jakém formátu si přeje uživatel vytvořit korelační pravidlo, preferovaný jazyk, který bude pravidlo reflektovat, možnost anonymizovat citlivá data posílaná v žádosti, žádost ve které uživatel specifikuje jaké korelační pravidlo by si přál vytvořit a nakonec samotné tlačítko pro odeslání žádosti. Výše popsané komponenty se týkají uživatelského vstupu při vytváření korelačních pravidel a lze je vidět na obrázku 6.3.

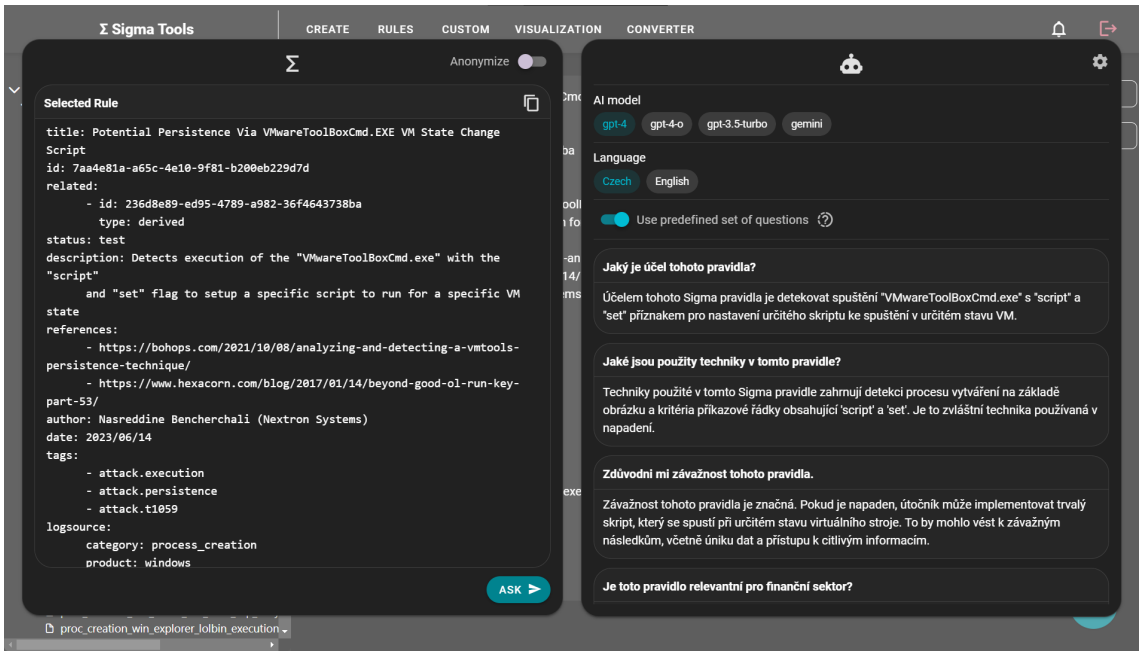
V odpovědi zobrazené na obrázku 6.4 lze potom vidět vygenerované pravidlo umělou inteligencí, které obsahuje všechny náležité atributy sigma pravidla, které by obsahovat mělo. Je ve správném formátu (YAML), a obsahuje reference na techniky MITRE, které odpovídají skutečnostem. Dané pravidlo lze tedy považovat za kompletní a dále s ním pracovat.

6.1.2 Vysvětlení korelačních pravidel

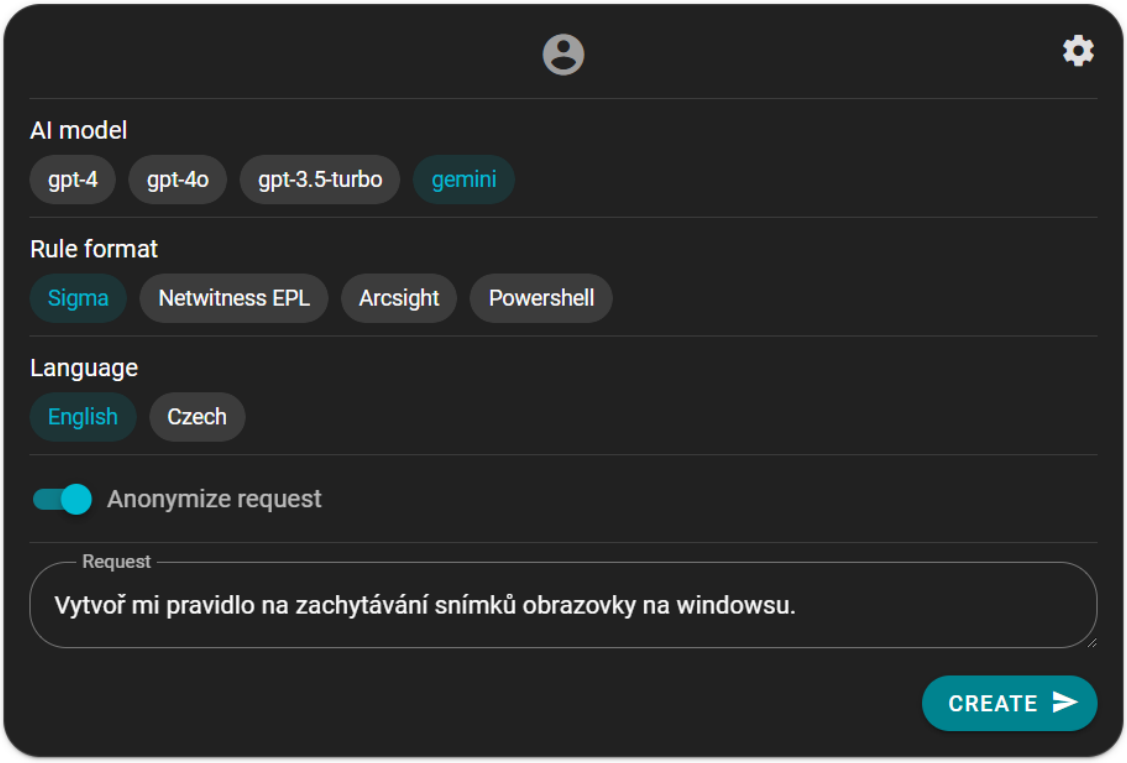
Tato část je koncipována tak, že se uživatel nachází v prostoru, kde si vybírá jaká korelační pravidla naimplementuje do svého bezpečnostního monitoringu společnosti.



Obr. 6.1: Pohled při zavřeném modálním okně.



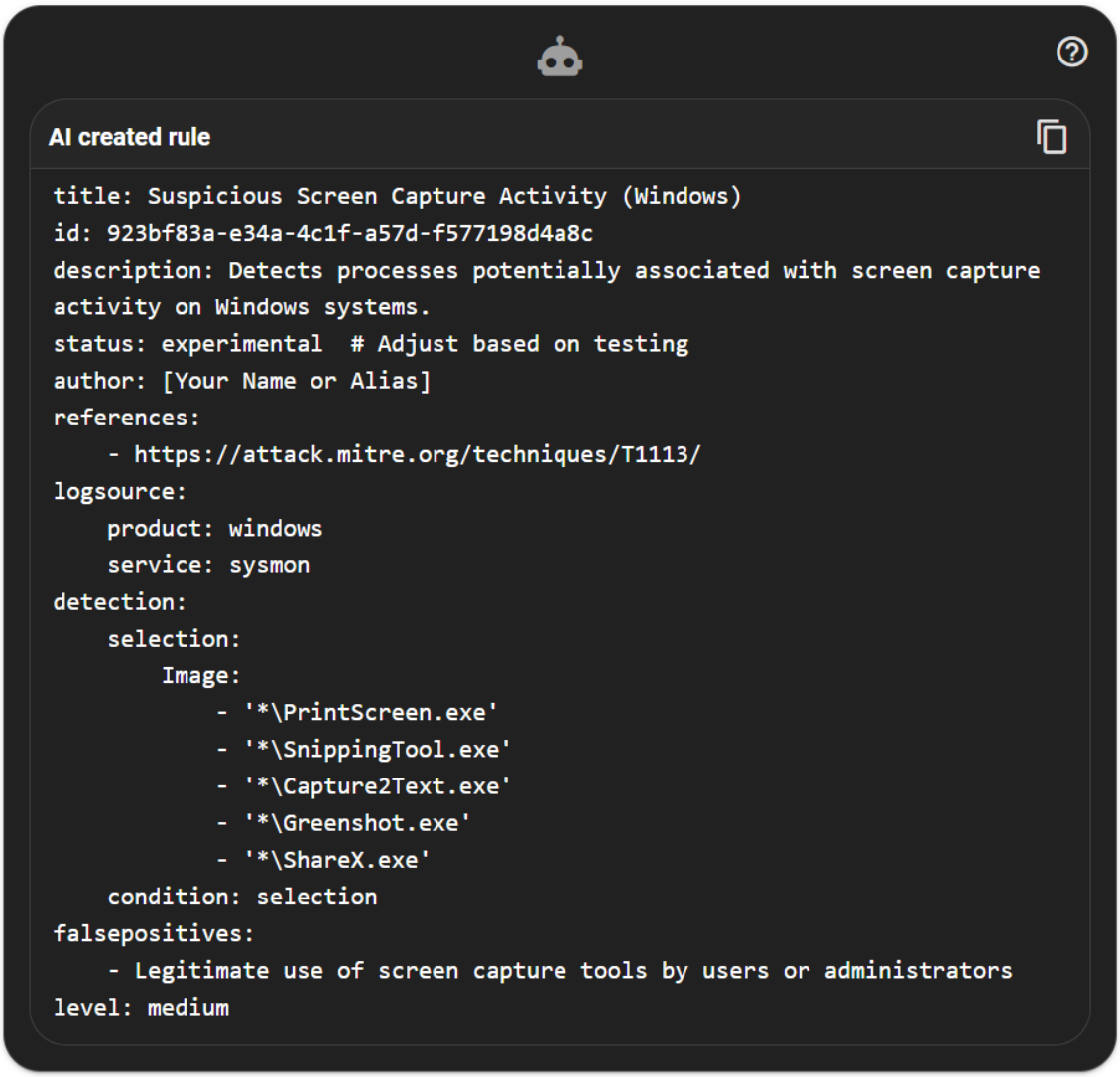
Obr. 6.2: Pohled při otevřeném modálním okně.



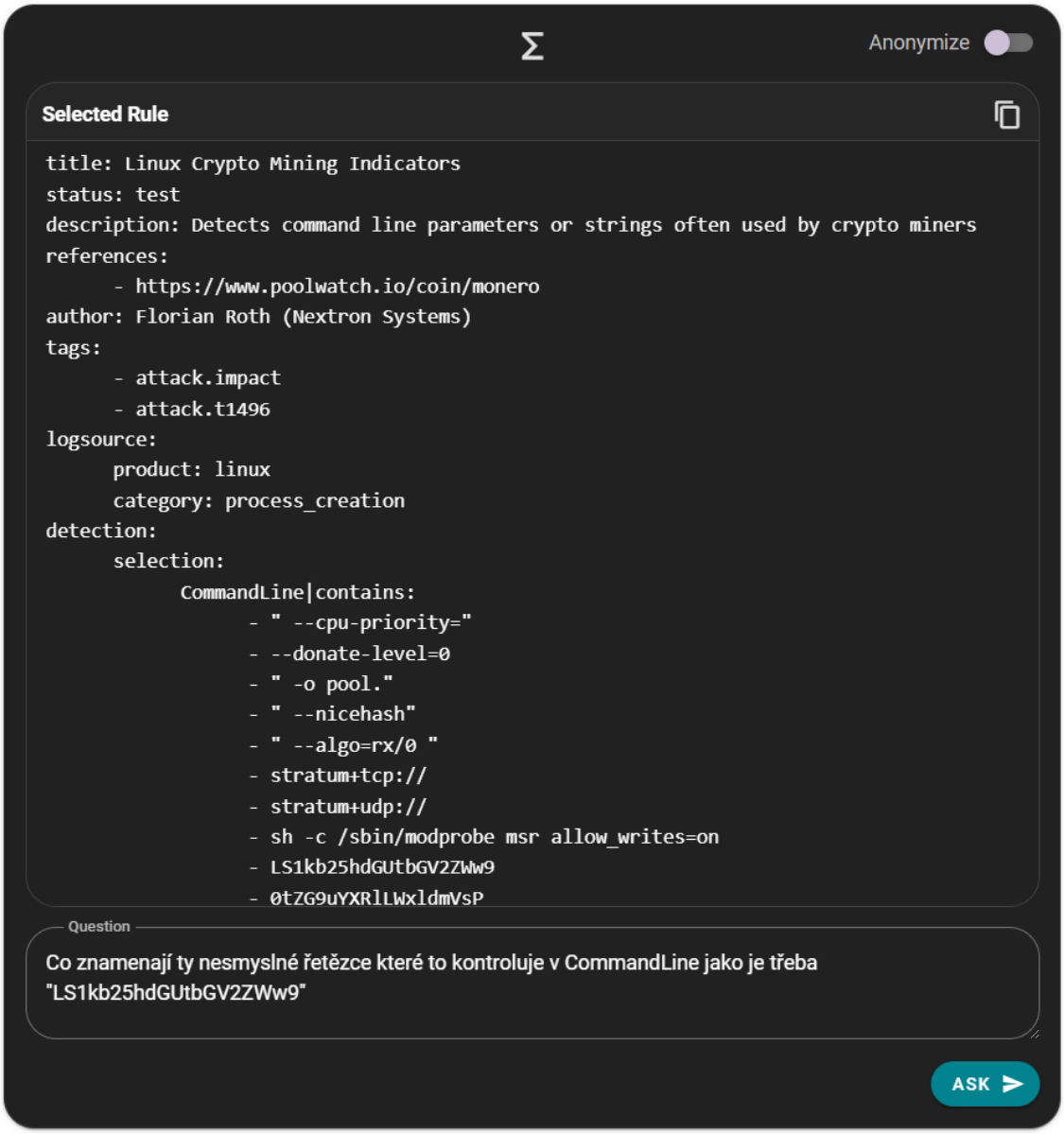
Obr. 6.3: Vstupní část při tvorbě nového korelačního pravidla.

Nachází se tedy v prostoru s pro něj neznámými pravidly, které převážně u rozsáhlejších pravidel nemusí být jednoduché na pochopení. Proto v případě, že si prohlíží pravidlo, má možnost pokládat dotazy jednotlivě na samotné pravidlo, nebo má mimo jiné možnost využít tak zvanou předdefinovanou sadu otázek, kterou si sám stanoví podle jeho preferencí, které se mohou vztahovat ke společnosti pro kterou pracuje, k operačním systémům, které využívá, k jednotlivým síťovým prvkům, které se nachází v organizaci pro kterou pravidlo zamýšlí a spoustu dalších. Předdefinovaná sada otázek nám tedy personalizuje prostor pro uživatele a urychluje případný výběr a nasazení korelačních pravidel, protože eliminuje nutnost posílat neustále každou otázku zvlášť u každého pravidla. Vizuální okno pro zobrazení zrovna vybraného pravidla a kladení jednotlivých dotazů na něj je zobrazeno na obrázku 6.5 a možnost správy, tedy přidávání a odebírání otázek z definované sady lze vidět na obrázku 6.7

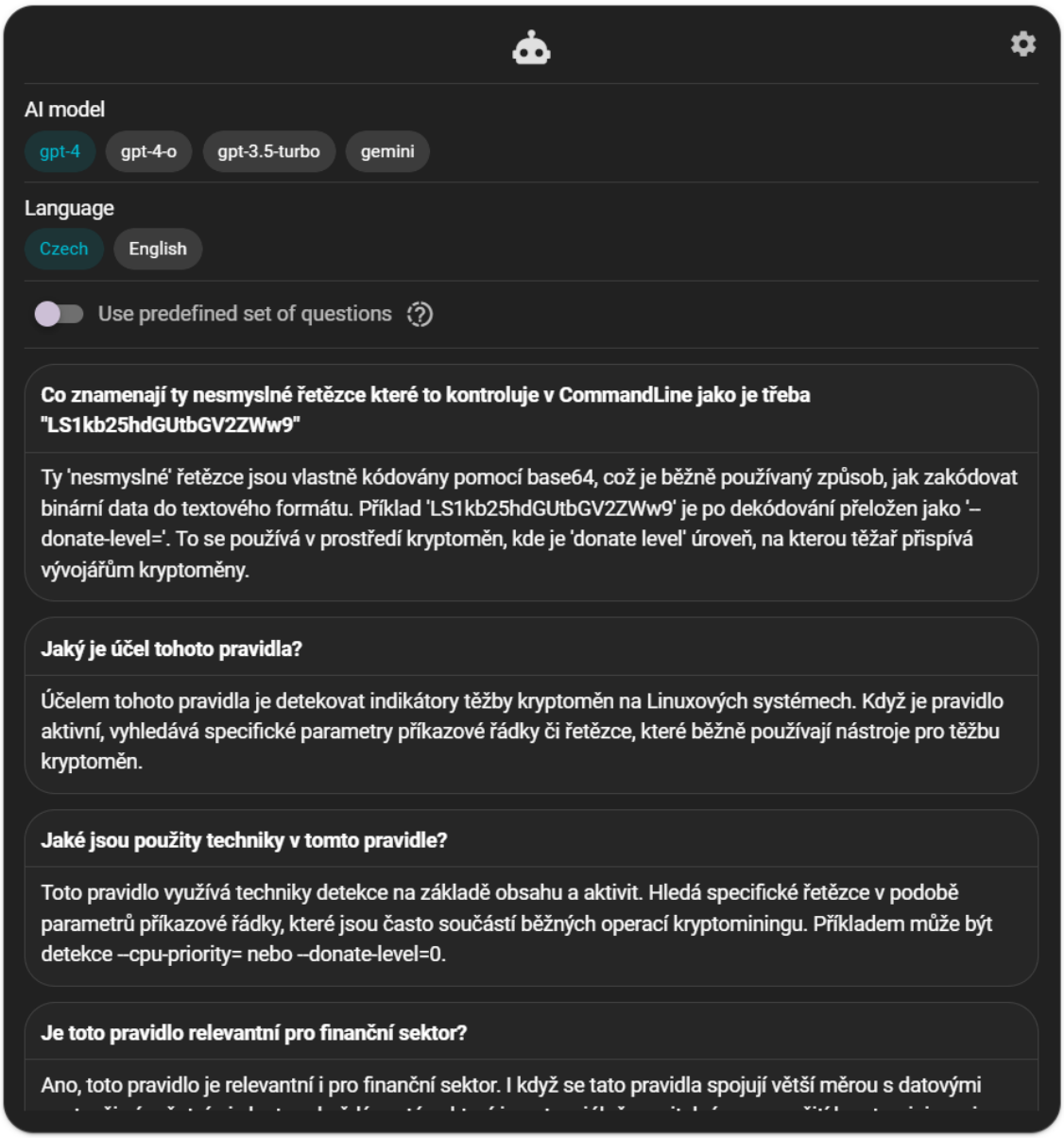
V odpovědi od umělé inteligence zobrazené na obrázku 6.6 lze potom vidět odpovědi jak na vzorovou předdefinovanou sadu otázek zobrazených na obrázku 6.7, tak i na jednotlivý dotaz položený na obrázku 6.5. Odpovědi jsou vhodně konstruované a zodpovězeny stručně a výstižně, což vede k rychlejšímu pochopení korelačního pravidla a případnému rychlejšímu nasazení do monitorovaného prostředí.



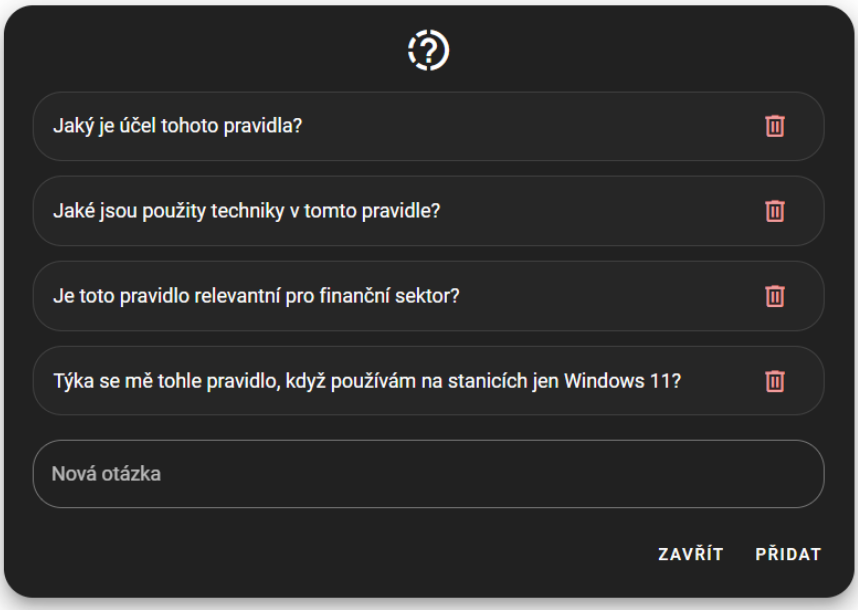
Obr. 6.4: Výstupní část při tvorbě nového korelačního pravidla.



Obr. 6.5: Vstupní část při dotazování na korelačního pravidla.



Obr. 6.6: Výstupní část při dotazování na korelačního pravidla.

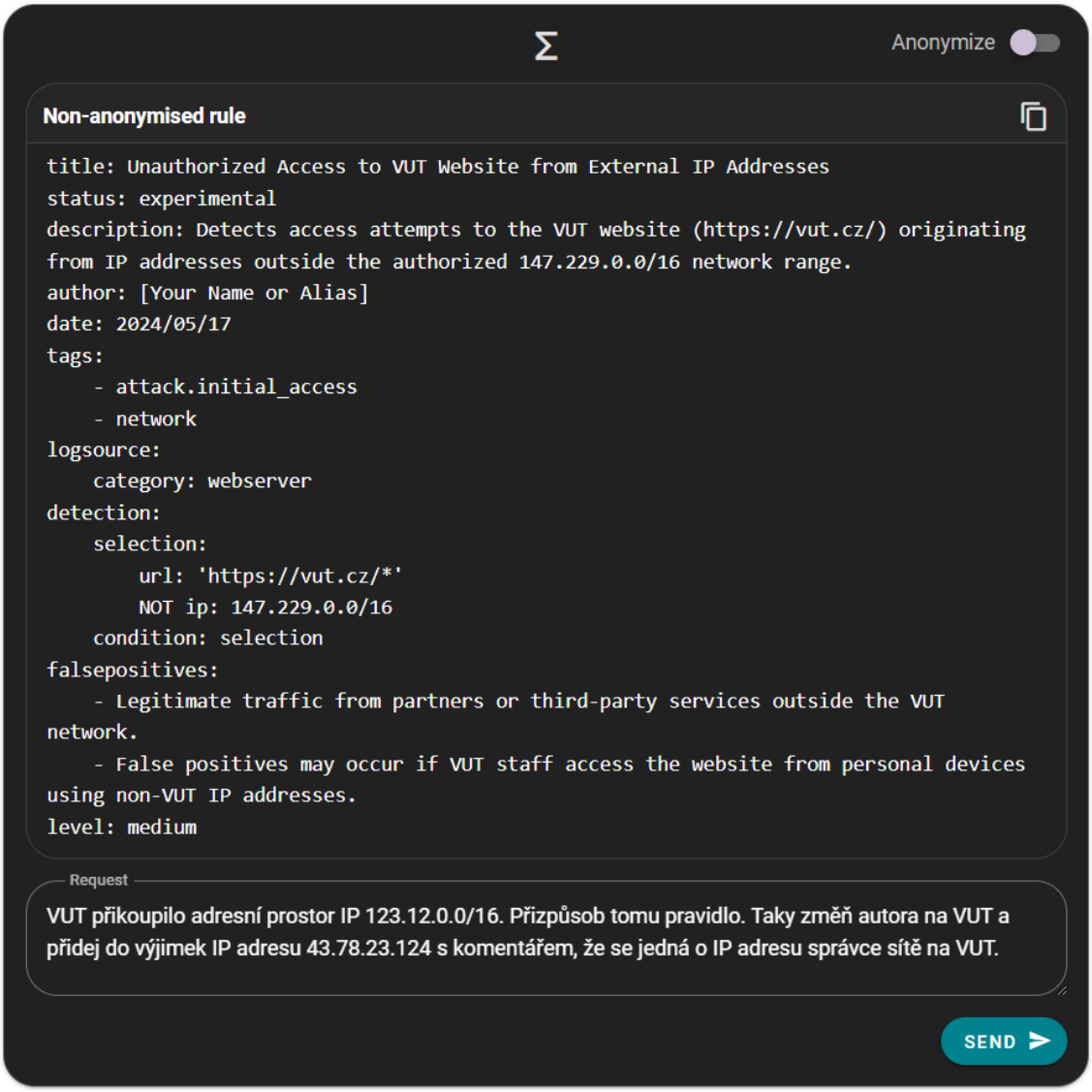


Obr. 6.7: Vzor předdefinované sady otázek.

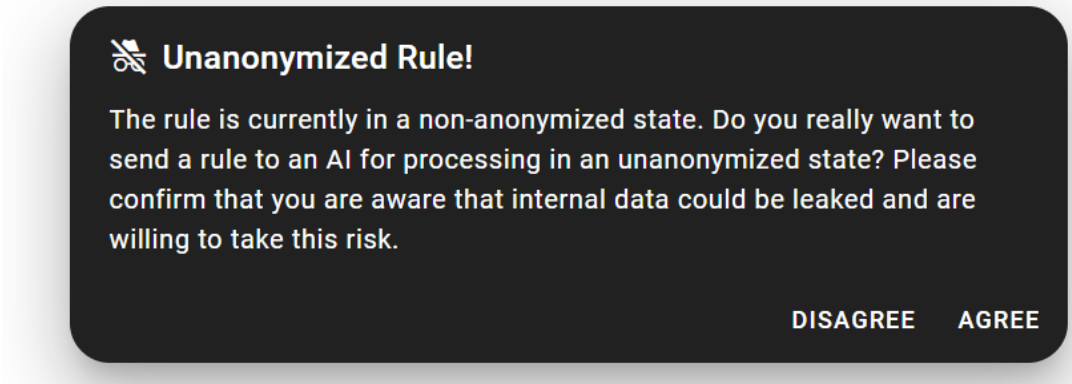
6.1.3 Úprava existujících korelačních pravidel

Co se týče úpravy existujících korelačních pravidel, tento prostor byl koncipován tak, že se jedná o místo, kde má uživatel svůj prostor se svými korelačními pravidly, které jsou již přizpůsobené pro jeho infrastrukturu. Jelikož se již nejedná o veřejný prostor ze kterého si uživatel teprve vybírá, která pravidla bude implementovat do svého bezpečnostního monitoringu, ale jedná se o prostor, kde se v korelačních pravidlech mohou nacházet e-mailové adresy, ip adresy a další citlivé údaje... je tu kladen vyšší důraz na anonymizaci a ochranu citlivých údajů. Vizuálně lze opět vidět vybrané pravidlo. Tentokrát místo dotazu ohledně pravidla lze vidět prostor pro specifikacy požadavku pro změnu pravidla. To lze vidět na obrázku 6.8.

V případě jako tento, kdy by uživatel zapomněl na anonymizaci dat před odesláním ke zpracování umělé inteligenci a nacházel by se v prostoru, kde má již svá existující korelační pravidla, které mohly podléhat již předchozím úpravám a přizpůsobením na míru dané společnosti a mohly by se v nich vyskytovat citlivá data, vyskočí uživateli upozornění o tom, že data neanonymizoval (viditelné na obrázku 6.9). V případě, že si je uživatel jistý, že se v pravidle nenachází žádná citlivá data, může požadavek odeslat neanonymizovaný. V případě, že uživatel data anonymizuje, lze vizuálně jednoduše poznat jaká data byla anonymizovaná jejich podbarvením. Anonymizovaná data lze vidět na obrázku 6.10.



Obr. 6.8: Vstupní část při úpravě korelačního pravidla (neanonymizovaná).



Obr. 6.9: Ověření při pokusu odeslat neanonymizovaná data.

V odpovědi od umělé inteligence kde jsme ji žádali o úpravu pravidla lze potom vidět upravené pravidlo, které je vizuálně podbarveno tak, aby šlo jednoduše určit, které části pravidla byly modifikovány. Řádky, které byly oddělány jsou podbarveny červeně a jsou přeškrtnuty, za to řádky které byly přidány jsou podbarveny zeleně a jsou podtrženy, což zajišťuje snadnou a rychlou identifikaci změn, které byly provedeny. Obzvláště v této komponentě uživatel potom využije pokročilé tlačítko pro kopírování, které zkopíruje upravené pravidlo bez řádků, které byly odstraněny.

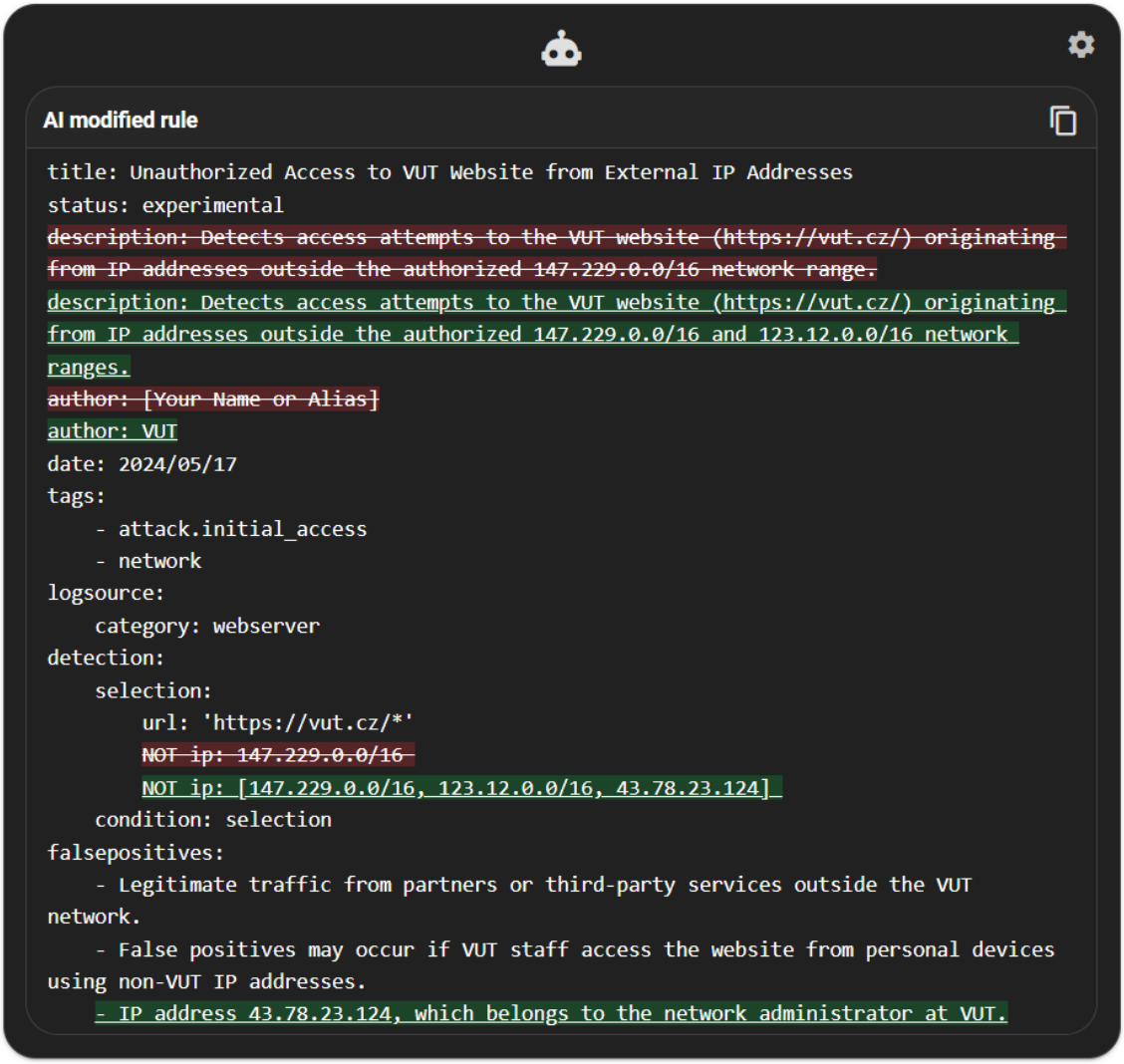
6.2 Vyhodnocení efektivity a přesnosti generovaných pravidel

K vyhodnocení efektivity a přesnosti lze podoknout, že co se týče generování nových korelačních pravidel, nejvíc obstojný se jevil model Gemini od společnosti Google, jelikož vygenerovaná pravidla od tohoto modelu se blížili či byly nerozeznatelné od člověkem napsaných pravidel. U nejschopnějšího modelu od společnosti OpenAI gpt-4 byly pravidla rovněž na vysoké úrovni, ale vzhledem k tomu, že nemá přístup k internetu, tak jeho výstupy nebyly vždy dokonalé a stávalo se, že zapomínal na nějaké atributy.

Co se týče kladení dotazů a vysvětlování korelačních pravidel. Nejvíce v tomto obstával model gpt-4 od společností OpenAI hlavně z toho důvodu, že dostupné API tohoto modelu poskytuje vlastní formátování výstupu od generativního modelu, což umožnilo přizpůsobit model tak, aby dokázal zpracovávat více požadavků společně.

Nakonec co se týče úpravy existujících korelačních pravidel, zde byl nejvhodnější

Obr. 6.10: Vstupní část při úpravě korelačního pravidla (anonymizovaná).



Obr. 6.11: Výstupní část při úpravě existujícího korelačního pravidla.

model opět gpt-4 od společnosti OpenAI, jelikož se dokázal držet předem specifikovaných pravidel a formátu tak, že byl nejkonzistentnější, což je požadované při práci s umělou inteligencí.

6.3 Možnosti budoucího vývoje a rozšíření

V rámci budoucího vývoje aplikace je plánováno rozšíření o integraci většího množství generativních modelů umělé inteligence. Vhodnou úpravou by bylo napojení na oficiální API předtrénovaných generativních modelů společnosti Google, pokud budou zpřístupněny veřejnosti.

Dalším krokem pro zlepšení aplikace je optimalizace ukládání předdefinované sady otázek. Namísto využívání local storage v prohlížeči uživatele, což může být méně spolehlivé, je plánován přechod na robustnější řešení s využitím databáze na serveru. Tím bude zajištěna lepší správa, dostupnost a škálovatelnost otázek, což přispěje k vyšší efektivitě.

Kromě toho je kladen důraz na vývoj pokročilejších kontrolních mechanismů pro výstupy generované umělou inteligencí. Cílem je minimalizovat výskyt nesprávných nebo nepřesných výsledků a zajistit konzistentnější a spolehlivější kvalitu generovaného obsahu. Implementací těchto kontrolních mechanismů bude aplikace schopna lépe vyhodnocovat a filtrovat výstupy, což povede ke zvýšení počtu úspěšných a relevantních výsledků pro uživatele.

Závěr

Tato bakalářské práce se zabývala využitím generativní umělé inteligence pro tvorbu, úpravu a vysvětlení korelačních pravidel v oblasti bezpečnostního monitoringu. Cílem bylo usnadnit a zefektivnit práci bezpečnostním analytikům při tvorbě a správě těchto pravidel.

V teoretické části práce byly představeny klíčové pojmy a technologie. Byly popsány základní aspekty zpracování přirozeného jazyka (*NLP*) [1.1] a moderní neuronové sítě, zejména architektura transformers (1.3), která je základem mnoha současných generativních modelů. Dále byly představeny principy bezpečnostního monitoringu kyberprostoru (2), systémy SIEM (2.1) a korelační pravidla (2.2).

Praktická část práce se zaměřila na návrh a implementaci webové aplikace, která integruje generativní umělou inteligenci do procesu tvorby a správy korelačních pravidel. Aplikace umožňuje generování nových pravidel na základě uživatelských požadavků, úpravu existujících pravidel a poskytuje vysvětlení pravidel s využitím umělé inteligence. (4, 5)

Výsledky testování aplikace ukázaly, že generativní umělá inteligence může být efektivním nástrojem pro tvorbu a správu korelačních pravidel. Model Gemini od společnosti Google se ukázal jako nejvhodnější pro generování nových pravidel, zatímco model GPT-4 od společnosti OpenAI byl nejefektivnější při vysvětlování a úpravě pravidel. (6.1, 6.2)

V rámci budoucího rozvoje by bylo vhodné rozšířit aplikaci o podporu více generativních modelů a implementovat pokročilejší kontroly výstupů umělé inteligence. Dále by bylo vhodné nahradit local storage v prohlížeči databází na serveru pro ukládání předdefinovaných sad otázek. (6.3)

Celkově lze dodat, že tato práce přispěla k prozkoumání možností využití generativní umělé inteligence v oblasti bezpečnostního monitoringu. Vytvořená webová aplikace má potenciál zefektivnit a zrychlit proces tvorby a správy korelačních pravidel a vést tak ke zvýšení bezpečnosti v kyberprostoru.

Literatura

[1] IBM. *What is natural language processing (NLP)?*, Online. Dostupné z: https://www.ibm.com/topics/natural-language-processing. [cit. 2023-11-08].

[2] VASWANI, Ashish, Noam SHAZEER, Niki PARMAR, Jakob USZKOREIT, Llion JONES, Aidan N. GOMEZ, Lukasz KAISER a Illia POLOSUKHIN. *Attention Is All You Need*, Online. Dostupné z: https://arxiv.org/abs/1706.03762. [cit. 2023-12-02].

[3] SLATOR. *What is the Difference Between NLP, NLU, and NLG?*, Online. Dostupné z: https://slator.com/resources/what-is-the-difference-between-nlp-nlu-nlg/. [cit. 2023-12-01].

[4] GEEKSFORGEEKS. *NLP vs NLU vs NLG*, Online. Dostupné z: https://www.geeksforgeeks.org/nlp-vs-nlu-vs-nlg/. [cit. 2023-12-02].

[5] PROJECTPRO. *10 NLP Techniques Every Data Scientist Should Know*, Online. Dostupné z: https://www.projectpro.io/article/10-nlp-techniques-every-data-scientist-should-know/415#toc-1. [cit. 2023-11-08].

[6] *What are neural networks?*, Online. Dostupné z: https://www.ibm.com/topics/neural-networks. [cit. 2023-11-12].

[7] GÉRON, Aurélien, 2018. *Neural networks and deep learning*, Online. O'Reilly Media. Dostupné z: https://learning.oreilly.com/library/view/neural-networks-and/9781492037354/. [cit. 2023-11-27].

[8] *Neural Network Architecture*, Online. Dostupné z: https://www.dremio.com/wiki/neural-network-architecture/. [cit. 2023-12-03].

[9] MELCHER, Kathrin. *A Friendly Introduction to [Deep] Neural Networks*, Online. Dostupné z: https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks. [cit. 2023-11-16].

[10] TUNSTALL, Lewis, Leandro von WERRA a Thomas WOLF, 2022. *Natural Language Processing with Transformers*, Online. O'Reilly Media. Dostupné z: https://learning.oreilly.com/library/view/-/9781098136789/. [cit. 2023-11-23].

[11] FOSTER, David, 2023. *Generative Deep Learning, 2nd Edition*, Online. O'Reilly Media. Dostupné z: https://learning.oreilly.com/library/view/generative-deep-learning/9781098134174/. [cit. 2023-11-28].

[12] *A Gentle Introduction to Generative Adversarial Networks (GANs)*, Online. Dostupné z: https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/. [cit. 2023-12-03].

[13] *Generative Adversarial Network (GAN)*, Online. Dostupné z: https://www.geeksforgeeks.org/generative-adversarial-network-gan/. [cit. 2023-12-03].

[14] Wikipedia přispěvatelé. *Transformer (machine learning model)*, In: Wikipedia: the free encyclopedia, Online. San Francisco (CA): Wikimedia Foundation. Dostupné z: https://en.wikipedia.org/w/index.php?title=Transformer_(machine_learning_model)&oldid=1187863034. [cit. 2023-12-02].

[15] Wikipedia přispěvatelé. *LSTM*, In: Wikipedia: the free encyclopedia, Online. San Francisco (CA): Wikimedia Foundation. Dostupné z: https://cs.wikipedia.org/w/index.php?title=LSTM&oldid=23403615. [cit. 2023-12-02].

[16] SAFONOV, Yehor, 2019. *Filtrování spamových zpráv pomocí metod umělé inteligence*. Brno, 150 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Martin Kolařík. [cit. 2023-12-02].

[17] Wikipedia přispěvatelé. *Generative adversarial network*, In: Wikipedia: the free encyclopedia, Online. San Francisco (CA): Wikimedia Foundation. Dostupné z: https://en.wikipedia.org/wiki/Generative_adversarial_network. [cit. 2023-11-02].

[18] *Generative AI and ChatGPT Enterprise Risks*, Online. 2023. Team8 CISO Village. Dostupné z: https://team8.vc/wp-content/uploads/2023/04/Team8-Generative-AI-and-ChatGPT-Enterprise-Risks.pdf. [cit. 2023-12-09].

[19] *Top 5 Pros and Cons of Generative AI*, Online. CloudTern Solutions. Dostupné z: https://www.linkedin.com/pulse/top-5-pros-cons-generative-ai-cloudtern. [cit. 2023-12-09].

[20] *The Benefits Of Artificial Intelligence And Machine Learning In SaaS Businesses*, Online. Dostupné z: https://www.forbes.com/sites/theyec/2023/03/23/the-benefits-of-artificial-intelligence-and-machine-learning-in-saas-businesses/. [cit. 2023-12-07].

[21] *AI in SaaS: What's The Future Going To Hold For SaaS Companies?*, Online. Dostupné z: https://www.mailmunch.com/blog/ai-in-saas. [cit. 2023-12-07].

[22] *Bard vs ChatGPT: Side-by-Side Comparison*, Online. Dostupné z: https://team-gpt.com/blog/bard-vs-chatgpt/#. [cit. 2023-12-09].

[23] LIBOVICKÝ, Jindřich. *Otázky a odpovědi o ChatGPT a velkých jazykových modelech*, Online. 2023. Dostupné z: https://jlibovicky.github.io/2023/02/07/Otazky-a-odpovedi-o-ChatGPT-a-jazykovych-modelech.html. [cit. 2023-12-02].

[24] INTERNATIONAL BUSINESS MACHINES. *What is SIEM?*, Online. Dostupné z: https://www.ibm.com/topics/siem. [cit. 2023-12-02].

[25] STOLTZFUS Justin. *What's the difference between SEM, SIM and SIEM?*, Online. Dostupné z: https://www.techopedia.com/7/31201/security/whats-the-difference-between-sem-sim-and-siem. [cit. 2023-11-27].

[26] *SIM, SEM, and SIEM: Definitions and Choosing the Right Enterprise Solution*, Online. Dostupné z: https://community.microfocus.com/cyberres/b/cybersecurity-blog/posts/sim-sem-and-siem-definitions-and-choosing-the-right-enterprise-solution. [cit. 2023-11-27].

[27] MICROSOFT. *What is SIEM?*, Online. Dostupné z: https://www.microsoft.com/cs-cz/security/business/security-101/what-is-siem. [cit. 2023-11-27].

[28] MICROSOFT. *What is SOAR?*, Online. Dostupné z: https://www.microsoft.com/cs-cz/security/business/security-101/what-is-soar. [cit. 2023-11-27].

[29] *Correlation rules and related events*, Online. Dostupné z: https://www.ibm.com/docs/en/noi/1.6.2?topic=events-correlation-rules-related. [cit. 2023-12-01].

[30] *Understanding correlation*, Online. Dostupné z: https://www.manageengine.com/products/eventlog/help/StandaloneManagedServer-UserGuide/Real-timeEventCorrelation/correlation-concepts.html. [cit. 2023-12-01].

[31] *What Are Sigma Rules?*, Online. Dostupné z: https://www.picussecurity.com/resource/glossary/what-is-sigma-rule. [cit. 2023-12-07].

[32] MILLER, David. *Security Information and Event Management (SIEM) Implementation*, 2010. [cit. 2024-05-12].

[33] *MITRE ATT&CK®*, Online. Dostupné z: `https://attack.mitre.org/`. [cit.,2024-05-07].

[34] *Advanced Persistent Threat (APT) Groups: What They Are and Where They Are Found*, Online. Dostupné z: `https://flashpoint.io/intelligence-101/advanced-persistent-threat/`. [cit.,2024-05-07].

[35] *AI SIEM: How SIEM with AI/ML is Revolutionizing the SOC*, Exabeam, Online. Dostupné z: `https://www.exabeam.com/explainers/siem/ai-siem-how-siem-with-ai-ml-is-revolutionizing-the-soc/`. [cit. 2024-05-19].

[36] *Future Of SIEM: The Age Of AI, Automation, And Cloud Technology*, BlackLight.ai, Online. Dostupné z: `https://blacklightai.com/insights/future-of-siem-the-age-of-ai-automation-and-cloud-technology/`. [cit. 2024-05-19].

[37] *Microservices vs. Monolith*, Atlassian, Online. Dostupné z: `https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith`. [cit. 2024-05-19].

[38] CHOLLET, François. *Deep learning v jazyku Python: knihovny Keras, Tensorflow*. Přeložil Rudolf PECINOVSKÝ. Knihovna programátora. Praha: Grada Publishing, 2019. ISBN 978-80-247-3100-1. [cit. 2023-11-27].

[39] *What is Flask Python*, Online. Dostupné z: `https://pythonbasics.org/what-is-flask-python/`. [cit. 2023-12-07].

[40] *Docker overview*, Online. Dostupné z: `https://docs.docker.com/get-started/overview/`. [cit. 2023-12-07].

[41] *What Is Vue JS?*, Online. Dostupné z: `https://builtin.com/software-engineering-perspectives/vue-js`. [cit. 2023-12-07].

[42] *Implement API Gateways with Ocelot*, Online. Dostupné z: `https://learn.microsoft.com/en-us/dotnet/architecture/microservices/multi-container-microservice-net-applications/implement-api-gateways-with-ocelot`. [cit. 2023-12-07].

Seznam symbolů a zkratek

AI *Artificial Intelligence* – Umělá inteligence

API *Application Programming Interface* – Rozhraní pro programování aplikací

APT *Advanced Persistent Threat* – Pokročilá trvalá hrozba

CISO *Chief Information Security Officer* – Ředitel pro informační bezpečnost

CSS *Cascading Style Sheets* – Kaskádové styly

EPL *Esper Processing Language* – Jazyk podobný SQL na korelaci dat RSA NetWitness

GenAI *Generative Artificial Intelligence* – Generativní umělá inteligence

HTML *Hypertext Markup Language* – Hypertextový značkovací jazyk

HTTP *Hypertext Transfer Protocol* – Hypertextový přenosový protokol

HTTPS *Hypertext Transfer Protocol Secure* – Zabezpečený hypertextový přenosový protokol

IoC *Indicator of Compromise* – Indikátor napadení

JS *JavaScript* – JavaScript

JSON *JavaScript Object Notation* – Zápis objektu JavaScriptu

LSTM *Long Short-Term Memory* – Dlouhá krátkodobá paměť (typ neuronové sítě)

NLG *Natural Language Generation* – Generování přirozeného jazyka

NLP *Natural Language Processing* – Zpracování přirozeného jazyka

NLU *Natural Language Understanding* – Porozumění přirozenému jazyku

SEM *Security Event Management* – Správa bezpečnostních událostí

SIEM *Security Information and Event Management* – Správa bezpečnostních informací a událostí

SIM *Security Information Management* – Správa bezpečnostních informací

SOAR *Security Orchestration, Automation, and Response* – Orchestrace, automatizace a reakce na bezpečnostní hrozby

SaaS *Software as a Service* – Software jako služba

TS *TypeScript* – TypeScript

URL *Uniform Resource Locator* – Jednotný lokátor zdroje

WAG *Web API Gateway* – Brána co příjmá HTTP požadavky, agreguje služby potřebné na jejich splnění a vrací příslušné odpovědi

YAML *YAML Ain't Markup Language* – YAML není značkovací jazyk

Seznam příloh

A Návod na spuštění aplikace

Jelikož aplikace využívá externí placená API, na kterých je závislá a bylo by potřeba si obstarat vlastní přístupové API klíče k zajištění její funkčnosti a rovněž z důvodu, že se aplikace nachází v zabezpečené infrastruktuře pana Ing. Yehora Safonova, do které lze přistoupit jen v případě udělení přístupu, bylo nahráno video představující samotnou aplikaci, které je dostupné na odkaze `https://youtu.be/hNLeckWScf8`.

V případě obstarání si přístupových API klíčů a bezpečnostních údajů k vstupu do infrastruktury je třeba si ke spuštění aplikace nejprve stáhnout elektronickou přílohu, která obsahuje jak klientskou tak serverovou část aplikace. Po stažení rozbalte zip archiv do libovolného umístění a přesuňte se do kořenového adresáře projektu. V případě potřeby naleznete strukturu obsahu elektronické přílohy níže v příloze C.

A.1 Spuštění aplikace pomocí dockeru

Zjednodušený start aplikace je možný prostřednictvím dockeru. Otevřete příkazovou řádku v kořenovém adresáři projektu a spusťte příkaz:

```
$ docker-compose up
```

(před spuštěním aplikace pomocí dockeru je nutné mít nainstalovaný a spuštěný docker).

A.2 Alternativní způsob spuštění aplikace

Alternativně lze spustit klientskou část z adresáře `frontend` pomocí příkazů:

```
$ npm install
```

```
$ npm run serve
```

Následně se je nutno se přesunout do adresáře `microservices/ms-aiIntegration` a doinstalovat potřebné balíčky a spustit serverovou část pomocí příkazů:

```
python -m pip install -r requirements.txt
```

```
flask run --host 127.0.0.1 --port 5002
```

Po úspěšném spuštění bude aplikace dostupná na adrese `http://localhost:5001/`.

B Zpracování požadavků na pozadí

Výpis B.1: Skladba požadavku sestaveného při odeslání na backend

```
POST /ai HTTP/1.1
Host: {ip-backend}:5002
Content-Length: 1753
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome
   /124.0.6367.118 Safari/537.36
Content-Type: application/json
Origin: http://{ip-frontend}:5001
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://{ip-frontend}:5001/
Accept-Encoding: gzip, deflate, br
Accept-Language: cs-CZ,cs;q=0.9
Connection: close

{
  "prompt": "Obsahuje␣korelační␣pravidlo␣společně␣s␣
     detailním␣popisem␣cíle,␣který␣je␣poskládaný␣na␣zá
     kladě␣požadavků␣uživatele␣a␣atributů,␣které␣si␣
     zvolil␣před␣odesláním␣požadavku␣a␣pravidly␣podle␣
     kterých␣se␣má␣model␣chovat.",
  "model": "gpt-4",
  "anon": "true",
  "mode": "advanced"
}
```

Výpis B.2: Ukázka konzolového výpisu anonymizované odpovědi AI na backendu.

```
[i] Anonymizovaná odpověď
{
  "answers": [
    "title: Unauthorized Access to VUT Website from
       External IP Addresses\nstatus: experimental\n<del
       style='background-color: #542426'>description:
       Detects access attempts to the VUT website (https
       ://johnson.biz/) originating from IP addresses
       outside the authorized 102.200.11.194/16 network
       range.</del>\n<ins style='background-color: #1
       c4428'>description: Detects access attempts to the
       VUT website (https://johnson.biz/) originating
       from IP addresses outside the authorized
       102.200.11.194/16 and 51.23.138.90/16 network
       ranges.</ins>\n<del style='background-color:
       #542426'>author: [Your Name or Alias]</del>\n<ins
       style='background-color: #1c4428'>author: VUT</ins
       >\ndate: 2024/05/17\ntags:\n    - attack.
       initial_access\n    - network\nlogsource:\n
       category: webserver\ndetection:\n    selection:\n
       url: 'https://johnson.biz/*'\n        <del
       style='background-color: #542426'>NOT ip:
       102.200.11.194/16 </del>\n        <ins style='
       background-color: #1c4428'>NOT ip:
       [102.200.11.194/16, 51.23.138.90/16,
       169.200.74.51] </ins>\n    condition: selection\
       nfalsepositives:\n    - Legitimate traffic from
       partners or third-party services outside the VUT
       network.\n    - False positives may occur if VUT
       staff access the website from personal devices
       using non-VUT IP addresses.\n    <ins style='
       background-color: #1c4428'>- IP address
       169.200.74.51, which belongs to the network
       administrator at VUT.</ins>\nlevel: medium"
  ]
}
```

Výpis B.3: Ukázka konzolového výpisu deanonymizované odpovědi AI na backendu.

```
[i] Denonymizovaná odpověď
{"answers": ["title: Unauthorized Access to VUT Website
   from External IP Addresses\nstatus: experimental\n<del
   style='background-color: #542426'>description:
   Detects access attempts to the VUT website (https://
   vut.cz/) originating from IP addresses outside the
   authorized 147.229.0.0/16 network range.</del>\n<ins
   style='background-color: #1c4428'>description: Detects
   access attempts to the VUT website (https://vut.cz/)
   originating from IP addresses outside the authorized
   147.229.0.0/16 and 123.12.0.0/16 network ranges.</ins
   >\n<del style='background-color: #542426'>author: [
   Your Name or Alias]</del>\n<ins style='background-
   color: #1c4428'>author: VUT</ins>\ndate: 2024/05/17\
   ntags:\n    - attack.initial_access\n    - network\
   nlogsource:\n    category: webserver\ndetection:\n    
   selection:\n        url: 'https://vut.cz/*'\n        <
   del style='background-color: #542426'>NOT ip:
   147.229.0.0/16 </del>\n        <ins style='background-
   color: #1c4428'>NOT ip: [147.229.0.0/16, 
   123.12.0.0/16, 43.78.23.124] </ins>\n    condition:
   selection\nfalsepositives:\n    - Legitimate traffic
   from partners or third-party services outside the VUT
   network.\n    - False positives may occur if VUT staff
   access the website from personal devices using non-
   VUT IP addresses.\n    <ins style='background-color:
   #1c4428'>- IP address 43.78.23.124, which belongs to
   the network administrator at VUT.</ins>\nlevel: medium
   "]}
```

C Obsah elektronické přílohy

Součástí bakalářské práce je elektronická příloha obsahující zdrojový kód klientské a serverové části aplikace, která má následující adresářovou strukturu:

- `/` ..... kořenový adresář přílohy
  - `frontend` ..... adresář klientské části aplikace
    - `public` ..... veřejné přílohy
      - `index.html` ..... předvolená stránka aplikace
    - `src`
      - `assets` ..... přílohy aplikace
      - `components` ..... části, do kterých jsou sekce rozdělené
        - `DynamicContentObserver` ... modální okno pro sběr dat a práci s nimi
      - `interfaces` ..... definice struktury použitých objektů
      - `plugins` ..... definice využitých pluginů
      - `router` ..... renderování požadované sekce
      - `services` ..... funkce potřebné na chod aplikace
      - `stores` ..... Pinia store
      - `utility` ..... pomocné funkce
      - `views` ..... obsah sekcí uživatelského rozhraní
      - `App.vue` ..... předvolený zdrojový soubor
      - `main.ts` ..... předvolený konfigurační soubor
    - `Dockerfile` ..... příkazy na sestrojení docker obrazu
    - `package.json` ..... metadata aplikace
  - `microservices` ..... adresář serverové části aplikace
    - `ms-aiIntegration`
      - `modules` ..... adresář modulů serverové části aplikace
        - `anon.py` ..... modul pro anonymizaci uživatelských vstupů
        - `functions.py` ..... výpomocné funkce pro anonymizační modul
        - `gemini-api.py` ..... modul pro napojení na neoficiální Gemini API
        - `openai-api.py` ..... modul pro napojení na oficiální OpenAI API
      - `.gitignore` .... specifikace souborů a adresářů, které mají být ignorovány nástrojem git
      - `app.py` ..... předvolený soubor FLASK aplikace
      - `Dockerfile` ..... příkazy na sestrojení docker obrazu
      - `README.md` ..... stručný popis serverové části aplikace
      - `requirements.txt` ..... požadavky potřebné na správný chod aplikace
  - `.dockerignore` ..... specifikace souborů a adresářů, které mají být ignorovány nástrojem docker
  - `.gitignore` specifikace souborů a adresářů, které mají být ignorovány nástrojem git
  - `.gitmodules` ..... konfigurace pro git submoduly
  - `docker-compose.yml` ..... konfigurační soubor pro orchestraci docker kontejnerů
  - `README.md` ..... Stručný popis jak spustit aplikaci