

**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

**EVIDENCIA IT KOMPONENTOV
TÍMOVÝ PROJEKT**

2025

Bc. Marek Štrba, Bc. Natália Koprivňáková,

Bc. Antonio Kiš, Bc. Marcel Soták, Bc. Samuel michalčík

Obsah

Úvod	1
Ponuka na vypracovanie tímového projektu	2
1 Terminológia	5
1.1 Čo je to sieť	5
1.1.1 Sieťové modely	5
1.1.2 Fyzické a logické siete	6
1.2 Čo je to zariadenie	6
1.2.1 IoT a jeho rola v moderných sieťach	7
1.3 Ako je zariadenie rozpoznané v sieti	7
1.4 Ako funguje scan siete a evidencia komponentov	8
2 Analýza existujúcich riešení a ciele práce	10
2.1 Funkcionálne a nefunkcionálne požiadavky	10
2.1.1 Funkcionálne požiadavky	10
2.1.2 Nefunkcionálne požiadavky	10
2.2 Aké riešenia existujú na evidenciu siete	11
2.3 Aké nástroje/príkazy existujú na scan siete	11
2.4 Výhody a nevýhody nami vybraného riešenia	12
2.5 Ciele práce	12
3 Návrh riešenia	14
4 Implementácia	17
4.1 Konfigurácia a autentifikácia	17
4.1.1 parse_arguments()	18
4.1.2 validate_config()	18
4.1.3 parse_subnet(args.address)	18
4.2 Skenovanie siete a spracovanie dát	18
4.2.1 Funkcia main()	18
4.2.2 run_nmap_scan(subnet)	19
4.2.3 execute_nmap(subnet=subnet)	19
4.2.4 parse_nmap_xml(result)	19
4.2.5 include_host(hosts)	20

4.2.6	get_local_ips()	20
4.2.7	upload(hosts, config, prefix)	20
4.2.8	export(args.output, hosts, hosts)	21
4.2.9	export_hosts_to_csv(hosts, filename, include_ids=False)	21
4.2.10	create_devices(selected_hosts, tenant, site, config)	21
4.2.11	create_interfaces(selected_hosts, config)	21
4.2.12	create_addresses(selected_hosts, tenant, config, prefix)	21
4.2.13	create_manufacturers(selected_hosts, config)	22
4.2.14	create_device_types(selected_hosts, config)	22
4.2.15	update_device(selected_hosts, config)	22
5	Nasadenie a testovanie	23
5.1	Testovanie	23
6	Návod na inštaláciu programu	27
6.1	Požiadavky	27
6.2	Inštalácia	27
7	Návod na obsluhu Netbox-u	29
7.1	Pridávanie nových používateľov	29
7.1.1	Prihlásenie a navigácia	29
7.1.2	Vytvorenie používateľa	30
7.1.3	Uloženie a aktivácia	32
7.2	Správa oprávnení - vytvorenie, priradenie a použitie	32
7.2.1	Vytvorenie novej skupiny	33
7.2.2	Priradenie oprávnení skupine	34
7.2.3	Priradenie skupiny používateľovi	36
7.2.4	Poznámky a ďalšie možnosti	37
7.3	Vytvorenie a správa API tokenov	37
7.3.1	Vytvorenie nového tokenu	38
7.3.2	Použitie API tokenu	40
7.4	Ďalšie dôležité položky	41
8	Návod na obsluhu programu	43
8.1	Základné spustenie skriptu	43
8.2	Možnosti importovania a výber ústavu a bloku	44
8.3	Výber role	44

8.4	Informácie o úspešnom nahratí a voľba exportovania	45
9	Prihlasovacie údaje a nosič	47
9.1	NetBox server	47
9.2	Softvérové zdroje a dokumentácia	47
10	Zápisnice	48
	Zoznam použitej literatúry	63

Zoznam obrázkov a tabuliek

Obrázok 1	Logika navrhovaného systému	14
Obrázok 2	Ústavy na fakulte	14
Obrázok 3	Ústavy na fakulte	15
Obrázok 4	Štruktúra databázy (prepojenie tabuliek ip address a device cez interface)	16
Obrázok 5	Ukážka štruktúry používateľov, ich privilégií a priradenie k ústavu	16
Obrázok 6	Importované zariadenia v NetBoxe, UIM, Blok A	25
Obrázok 7	Dostupné role pre zariadenia	25
Obrázok 8	Importované typy zariadení podľa výrobcov	26
Obrázok 9	Importovaní výrobcovia	26
Obrázok 10	Importované IP adresy	26
Obrázok 11	Prihlásený používateľ	29
Obrázok 12	Navigácia k používateľom	30
Obrázok 13	Tabuľka používateľov	30
Obrázok 14	Formulár pre vytvorenie nového používateľa	32
Obrázok 15	Tabuľka skupín	33
Obrázok 16	Vytvorenie novej skupiny	34
Obrázok 17	Tabuľka oprávnení	34
Obrázok 18	Vytvorenie oprávnenia pre UIM administrátorov podľa návrhu .	36
Obrázok 19	Priradenie skupín používateľovi	37
Obrázok 20	Tabuľka tokenov	38
Obrázok 21	Navigácia k správy vlastných tokenov	38
Obrázok 22	Tabuľka vlastných tokenov	39
Obrázok 23	Priradenie nového tokenu používateľovi	40
Obrázok 24	Navigácia k zoznamu organizačných informácií a zoznamu ústavov (Tenants)	41
Obrázok 25	Navigácia na adminovu stránku	41
Obrázok 26	Navigácia k zoznamu zariadení	42
Obrázok 27	Navigácia k zoznamu IP adries	42
Obrázok 28	Zoznam nájdených zariadení	43
Obrázok 29	Možnosti importu a voľba ústavu/bloku	44
Obrázok 30	Nastavenie role zariadeniam	45

Obrázok 31	Výpis o úspešnom importovaní a možnosť exportovania	46
------------	---	----

Úvod

Pri reakcii na incident je potrebné, aby CSIRT tímy mali včas k dispozícii informácie o štruktúre siete a o komponentoch a zariadeniach, ktoré sú v sieti prítomné. Preto je potrebné, aby administrátori sietí mali k dispozícii vhodné nástroje a riešenia pre evidenciu komponentov a potrebných informácií o nich (MAC adresa, IP adresa, typ zariadenia, bežiacie služby, atď.). Cieľom zadania je vytvorenie systému alebo implementácia vhodného už existujúceho riešenia pre evidenciu komponentov v sieťovej a IT infraštruktúre.

Znenie zadania: Úlohou zadania je návrh vlastného riešenia alebo implementácia vybraného existujúceho riešenia pre evidenciu a zber informácií o komponentoch v IT a sieťovej infraštruktúre pre potreby administrátorov, ale aj CSIRT tímov. Riešenie by malo poskytovať podobnú funkcionality a možnosť importu a exportu dát, akú má open-source riešenie NetBox (<https://github.com/netbox-community/netbox>), a tiež možnosť detekcie nových zariadení, prípadne skenovanie informácií a zmien. Riešenie by malo umožňovať evidenciu komponentov pre viacerých používateľov (administrátorov). Riešenie by malo byť nasadené na Ústave informatiky a matematiky.

Úlohy:

1. Analyzujte existujúce riešenia pre evidenciu komponentov IT infraštruktúry.
2. Vyberte vhodné riešenie alebo implementujte vlastné riešenie.
3. Analyzujte potrebné informácie pre zber a ukladanie, ktoré sú nevyhnutné pre administrátorov všeobecne.
4. Analyzujte potrebné informácie pre zber na Ústave informatiky a matematiky.
5. V spolupráci s administrátorom vykonajte analýzu komponentov a ich evidenciu v navrhnutom systéme.
6. Overte správnosť údajov, ich vzájomné prepojenie a súvislosti a otestujte viac užívateľský prístup.
7. V spolupráci s administrátorom otestujte riešenie a porovnajte ho s inými riešeniami, zhodnotte výhody a nevýhody riešenia.

Ponuka na vypracovanie tímového projektu

Ponuka na vypracovanie projektu na tému:

Evidencia IT komponentov

1. Náš tím:

- Natália Koprivňáková
 - Zameranie: API – kybernetická bezpečnosť
 - Predchádzajúce skúsenosti: Potvrdenie incidentu z IDS logov s využitím AI (bakalárska práca) - práca bola zameraná na simulovanie kybernetických útokov a ich následné potvrdzovanie
 - Predmety: Počítačová kriminalita, Klasické šifry, Základy kryptografie, Úvod do počítačovej bezpečnosti
 - Technológie a znalosti: rest API, CSS, PHP (základy), MySQL, Elasticsearch
- Marek Štrba
 - Zameranie: API - umelá inteligencia
 - Predchádzajúce skúsenosti: web crawler v bezpečnostnej doméne v pythone (bakalárska práca) - v práci som sa venoval aj napríklad zaťažovaniu crawlovaných stránok requestami čo by mohlo byť potenciálne zavrhnuté pri testovaní, manuálne testovanie aplikácie (pokladničný systém), automatizované testy, nasadzovanie aktualnych verzii systému (pokladničný systém na pokladne)
 - Predmety: Počítačová kriminalita, Klasické šifry, Úvod do počítačovej bezpečnosti
 - Technológie a znalosti: TypeScript, Cypress, ERP, SAP, rest API, základy .NET
- Marcel Soták
 - Zameranie: API – kybernetická bezpečnosť
 - Predchádzajúce skúsenosti: vývoj fullstackových webových stránok, vývoj webovej aplikácie na riadenie a vizualizáciu dynamického systému (bakalárska práca) - návrh a vývoj webovej aplikácie schopnej komunikovať s arduino-obvodom cez node-RED server, školský záverečný projekt - webová stránka na správu študentov v Laraveli.
 - Predmety: Počítačová kriminalita, Klasické šifry, Základy kryptografie, Úvod do počítačovej bezpečnosti
 - Technológie a znalosti: Laravel, React, CSS, JavaScript, Java, PHP, REST, Unity

- Samuel Michalčík
 - Zameranie: API – kybernetická bezpečnosť
 - Predchádzajúce skúsenosti: vývoj fullstackových webových aplikácií a webových stránok
 - Predmety: Webové technológie (absolvované), Základy kryptografie, Úvod do počítačovej bezpečnosti, Spoločenské, morálne a právne súvislosti vývoja informačných systémov
 - Technológie a znalosti: Laravel, React, Tailwind Css, obsluha databázy, API, základné znalosti v oblasti správy VPS (Linux + Nginx)
- Antonio Kiš
 - Zameranie: API - inteligentné softvérové systémy
 - Predchádzajúce skúsenosti: testovanie webových aplikácií z užívateľskej stany,
 - Predmety: Úvod do počítačovej bezpečnosti
 - Technológie a znalosti: obsluha databázy, API, základné znalosti v Laraveli, Unity Game Development, ML Agents (začiatočník)

2. Motivácia:

Hlavnou motiváciou na výber témy je aktuálne inžinierske zameranie (kybernetická bezpečnosť), prípadne predchádzajúce bakalárske zameranie (bezpečnosť informačných systémov) väčšiny členov tímu. Radi by sme zvýšili mieru zabezpečenia (fakultnej) infraštruktúry, respektíve sa podieľali na podpore CSIRT-u a prispeli tak k jednoduchšiemu sledovaniu zariadení, pripojených na danú sieť. Za pridanú hodnotu tohto projektu považujeme získanie znalostí v oblasti bezpečnosti a skúseností s prácou v tíme. Ďalším dôvodom, pre ktorý sme si tému vybrali, je voľba vedúceho tímu, pretože si myslíme, že bude pre náš tím oporou, poskytne nám potrebnú podporu a bude nás motivovať k dosahovaniu lepších výsledkov počas riešenia projektu.

3. Čo môžeme poskytnúť:

- a) Preskúmame aktuálne dostupné riešenia a na základe požiadaviek zadávateľa projektu a našej analýzy určíme, či pôjde o vývoj nového riešenia, alebo implementáciu už existujúceho riešenia.
- b) Vypracujeme návrh a stanovíme ciele na základe úloh a vstupnej konzultácii so zadávateľom projektu.
- c) Implementujeme navrhnuté riešenie a otestujeme evidenciu IT komponentov.

Vzhľadom na to, že nie sme členmi tímu CSIRT a nepoznáme fungovanie infraštruktúry Ústavu matematiky a informatiky, považujeme za potrebné absolvovať vstupnú konzultáciu so zadávateľom projektu prípadne administrátorom siete a bližšie si stanoviť ciele a špecifikácie riešenia.

4. Predpokladané zdroje:

- Použité technológie: určíme po konzultácii a na základe rozhodnutia, či pôjde o implementáciu existujúceho riešenia alebo vývoj nového riešenia.
- Časové požiadavky: Na základe kreditového systému (1 kredit = 25 - 30 hodín) a súčasnom hodnotení predmetov Tímový projekt 1 (2 kredity) a Tímový projekt 2 (4 kredity) predpokladáme, že sa každý člen tímu bude zaoberať projektom v priemere aspoň 6 hodín týždenne po dobu dvoch semestrov.
- Miestnosť: potrebovali by sme k dispozícii miestnosť (na FEI STU) na pravidelné stretnutia tímu

5. Návrhy na zmenu organizácie:

Aktuálne nemáme žiadne návrhy na zmenu organizácie, dodatočne by sme ich odkomunikovali a doplnili na základe konzultácie. Z hľadiska kontroly a bodovania by sme privítali, ak by sa bral ohľad na to, že niektoré týždne máme ako študenti vyťaženejšie kvôli iným zadaniam. Naopak v časovo flexibilnejších týždňoch, by sme sa tímovému projektu venovali nad rámec týždenných predpokladov, aby sme dodržali stanovený priemer.

1 Terminológia

Predtým, než sa začneme zaoberať podrobnejším rozborom problematiky a popisom implementačnej časti, považujeme za dôležité najskôr objasniť niekoľko základných pojmov, ktoré sa v našej práci často vyskytujú.

1.1 Čo je to sieť

Sieť je systém prepojených zariadení, ktoré umožňujú vzájomnú výmenu údajov. Počítačová sieť je skupina vzájomne prepojených uzlov alebo výpočtových zariadení, ktoré si navzájom vymieňajú údaje a zdroje. Sieťové spojenie medzi týmito zariadeniami je možné vytvoriť pomocou kábla alebo bezdrôtového média. [1] Siete sa môžu líšiť rozsahom, účelom a technológiami, ktoré ich tvoria. Základné typy sietí zahŕňajú:

- **LAN (Local Area Network):** Lokálna sieť pokrýva malú geografickú oblasť, ako je dom, kancelária alebo školská budova. Používa sa na zdieľanie zdrojov, ako sú tlačiarne, súbory a internetové pripojenie. Typickým príkladom je sieť spojená prostredníctvom ethernetového kábla alebo Wi-Fi.
- **MAN (Metropolitan Area Network):** Metropolitná sieť pokrýva väčšiu oblasť, napríklad mesto alebo veľkú univerzitnú areu. MAN siete často spájajú viacero LAN sietí prostredníctvom vysokorýchlostných spojení, ako sú optické vlákna.
- **WAN (Wide Area Network):** Globálna sieť pokrýva veľké geografické oblasti, často kontinenty alebo celý svet. Internet je najväčšou a najznámejšou WAN sieťou. WAN siete sú vytvorené kombináciou verejných a súkromných spojení.
- **PAN (Personal Area Network):** Osobná sieť je navrhnutá pre jednotlivcov a pokrýva malý dosah, zvyčajne do niekoľkých metrov. Typické príklady zahŕňajú pripojenie medzi smartfónom a Bluetooth slúchadlami alebo medzi notebookom a myšou cez Wi-Fi. [1] [2]

Okrem týchto typov môžu siete zahŕňať aj špecializované formy, ako sú SAN (Storage Area Network) na správu úložísk alebo VPN (Virtual Private Network) na bezpečný prístup cez verejný internet.

1.1.1 Sieťové modely

Na pochopenie fungovania sietí sú kľúčové dva hlavné sieťové modely:

- **OSI model (Open Systems Interconnection):** OSI model má sedem vrstiev: fyzická, linková, sieťová, transportná, relačná, prezentačná a aplikačná. Každá vrstva vykonáva špecifické úlohy, ktoré spolu zabezpečujú spoľahlivú komunikáciu medzi zariadeniami. [3]
- **TCP/IP model:** Tento model je zjednodušenou a praktickejšou alternatívou k OSI modelu. Má štyri vrstvy: sieťový prístup, internetová vrstva, transportná vrstva a aplikačná vrstva. TCP/IP model sa používa na popis protokolov, ktoré sú základom internetu.

1.1.2 Fyzické a logické siete

Siete môžu byť chápané na dvoch úrovniach:

- **Fyzická sieť:** Zahŕňa hardvérové komponenty, ako sú káble, switche, routery a samotné zariadenia. Fyzická sieť určuje, ako sú zariadenia skutočne prepojené.
- **Logická sieť:** Definuje spôsob, akým zariadenia komunikujú, vrátane topológie, protokolov a pravidiel. Príkladom logickej siete je IP adresovanie a smerovanie paketov v rámci internetu. [4]

Fyzická a logická vrstva často spolupracujú. Napríklad fyzická topológia môže byť zmesou hviezdicovej a kruhovej topológie, zatiaľ čo logická topológia môže definovať, že všetky zariadenia komunikujú cez centrálného routera. [1]

1.2 Čo je to zariadenie

Zariadenie v sieti je akýkoľvek fyzický alebo virtuálny objekt, ktorý môže odosielať, prijímať alebo spracovávať údaje. Zariadenia môžeme rozdeliť na:

- **Klienti:** Sú to koncové zariadenia, ktoré využívajú služby siete. Patria sem počítače, notebooky, mobilné telefóny, tablety alebo IoT zariadenia.
- **Servery:** Zariadenia poskytujúce služby klientom. Servery môžu byť webové, databázové, aplikačné alebo súborové, a sú zvyčajne výkonné počítače optimalizované na nepretržitý chod.
- **Aktívne prvky siete:** Zahŕňajú komponenty ako switche, routery a access pointy, ktoré riadia prenos údajov a zabezpečujú konektivitu v sieti. Switche spracovávajú údaje na linkovej vrstve, zatiaľ čo routery zabezpečujú smerovanie na sieťovej vrstve.

- **IoT zariadenia (Internet of Things):** popisuje sieť fyzických objektov – „vecí“ – ktoré sú zabudované senzormi, softvérom a inými technológiami na účely pripojenia a výmeny údajov s inými zariadeniami a systémami cez internet. [5] Typické IoT zariadenia zahŕňajú smart senzory, domáce asistentov (napr. Amazon Echo), alebo zariadenia na monitorovanie zdravotného stavu.

1.2.1 IoT a jeho rola v moderných sieťach

IoT (Internet of Things) predstavuje sieť fyzických zariadení, ktoré sú schopné zbierať a zdieľať dáta cez internet. IoT zariadenia zvyšujú efektivitu a automatizáciu v rôznych oblastiach, ako sú:

- **Priemysel 4.0:** Automatizované výrobné procesy a monitorovanie prostredníctvom senzorov.
- **Inteligentné domácnosti:** Zariadenia ako termostaty, osvetlenie alebo bezpečnostné kamery, ktoré môžu byť ovládané na diaľku.
- **Zdravotníctvo:** IoT zariadenia monitorujúce vitálne funkcie pacientov a odosielajúce dáta lekárom.
- **Doprava:** Smart navigačné systémy a autonómne vozidlá komunikujúce s inými zariadeniami v reálnom čase.

IoT však prináša aj výzvy, ako sú bezpečnosť a správa veľkého množstva dát. Tieto zariadenia sú často cieľom kybernetických útokov, preto je potrebné zabezpečiť ich spoľahlivosť a integritu v sieti.

1.3 Ako je zariadenie rozpoznané v sieti

Na identifikáciu zariadení v sieti sa používajú rôzne mechanizmy a adresy, ktoré zabezpečujú jednoznačné určenie ich identity. Medzi hlavné metódy patrí:

- **IP adresa (Internet Protocol Address):** Slúži na identifikáciu zariadení v rámci logickej topológie siete. IP adresy môžu byť statické (pridelené manuálne) alebo dynamické (pridelené prostredníctvom DHCP). Tieto adresy umožňujú komunikáciu medzi zariadeniami v rámci rovnakého subnetu alebo naprieč rôznymi sieťami. [6]
- **MAC adresa (Media Access Control Address):** Fyzická adresa zariadenia, pevne priradená sieťovej karte výrobcom. MAC adresa je jedinečná pre každé zariadenie a

používa sa na identifikáciu zariadení na linkovej vrstve OSI modelu. Vďaka nej môže switch efektívne smerovať pakety v rámci lokálnej siete. [7]

- **Hostname:** Každé zariadenie v sieti môže mať pridelený unikátny názov (hostname), ktorý je jednoduchšie zapamätateľný ako IP adresa. Hostname sa často mapuje na IP adresu pomocou DNS (Domain Name System).
- **Identifikátory na aplikačnej vrstve:** Niektoré aplikácie alebo služby môžu používať svoje vlastné identifikátory na rozpoznávanie zariadení. Napríklad VoIP systémy používajú SIP adresy na identifikáciu telefónov.
- **Digitálne certifikáty:** V prípade šifrovaných komunikácií môžu zariadenia používať digitálne certifikáty na autentifikáciu a identifikáciu. Certifikáty sú obvykle vydané certifikačnými autoritami (CA) a obsahujú informácie o zariadení.
- **ARP tabuľka (Address Resolution Protocol):** Používa sa na mapovanie IP adries na MAC adresy v lokálnych sieťach. Keď zariadenie komunikuje s iným zariadením, najprv zistí jeho MAC adresu pomocou ARP. [8]
- **SNMP identifikátory:** V rámci správy siete sa používa protokol SNMP (Simple Network Management Protocol), ktorý umožňuje monitorovanie zariadení pomocou unikátnych identifikátorov, ako sú OID (Object Identifiers).
- **Porty a služby:** Zariadenia môžu byť identifikované aj na základe otvorených portov a bežiacich služieb. Napríklad webový server je rozpoznateľný na porte 80 (HTTP) alebo 443 (HTTPS).

Tieto metódy spolupracujú na zabezpečení, že každé zariadenie v sieti je jednoznačne identifikovateľné a správne integrované do infraštruktúry siete.

1.4 Ako funguje scan siete a evidencia komponentov

Scan siete je proces zisťovania aktívnych zariadení a ich vlastností v konkrétnej sieti. Na tento účel sa využívajú rôzne techniky a nástroje, ktoré umožňujú detailnú analýzu siete. Medzi hlavné kroky patrí:

- **Zisťovanie aktívnych IP adries:** Používajú sa nástroje, ako je *ping sweep*, ktoré zisťujú, ktoré IP adresy v sieti odpovedajú.

- **Identifikácia otvorených portov:** Nástroje, ako je *Nmap*, umožňujú identifikovať, ktoré porty sú otvorené na jednotlivých zariadeniach, čo poskytuje informácie o bežiacich službách.
- **Zisťovanie operačných systémov:** Pomocou techník, ako je *OS fingerprinting*, je možné určiť, aký operačný systém beží na konkrétnom zariadení.
- **Detekcia zraniteľností:** Niektoré nástroje dokážu skenovať známe zraniteľnosti na zariadeniach a službách, čím poskytujú informácie pre zvýšenie bezpečnosti.
- **Mapovanie topológie siete:** Nástroje, ako je *Traceroute*, umožňujú identifikovať trasu medzi zariadeniami a vytvárať mapu topológie siete.
- **Zaznamenávanie údajov:** Informácie o zariadeniach, ako sú IP a MAC adresy, názvy zariadení, ich poloha a funkcia, sa zapisujú do centrálnej databázy alebo systému správy siete, ako je *NetBox*.
- **Monitorovanie siete:** Protokoly, ako je SNMP, sa používajú na kontinuálne sledovanie zariadení, ich výkonu a dostupnosti.

Evidencia komponentov predstavuje systematické zaznamenávanie všetkých relevantných údajov o zariadeniach v sieti. To zahŕňa:

- **Kategorizáciu zariadení:** Rozdelenie zariadení podľa typu (router, switch, server, IoT zariadenie atď.).
- **Zaznamenávanie konfigurácie:** Uloženie detailov o nastavení zariadenia, ako sú sieťové parametre, prístupové údaje a priradené služby.
- **Umiestnenie v infraštruktúre:** Zaznamenanie fyzickej polohy zariadenia (napríklad v konkrétnej miestnosti alebo racku).
- **História zmien:** Evidovanie zmien konfigurácie alebo údržby zariadenia.

Tieto kroky a procesy zabezpečujú, že sieťová infraštruktúra je nielen efektívne spravovaná, ale aj dobre dokumentovaná, čo zjednodušuje jej údržbu a rozvoj.

2 Analýza existujúcich riešení a ciele práce

Pri analyzovaní existujúcich riešení sme konzultovali s kompetentnými osobami ústavu ÚIM. Naše zariadenie by malo zlepšiť dostupnosť informácií o zariadeniach v sieti, priradiť hierarchiu pre administrátorov jednotlivých ústavov školy a možnosť jednoduchého CRUD.

2.1 Funkcionálne a nefunkcionálne požiadavky

Systémové požiadavky môžeme kategorizovať ako **funkcionálne** a **nefunkcionálne** požiadavky. Slúžia ako súhrn implementovaných procesov systému.

2.1.1 Funkcionálne požiadavky

Funkcionálne požiadavky môžu byť definované, ako niečo, čo systém musí robiť. Ak má systém správne fungovať, musí dodržiavať tieto požiadavky. Koncept funkcionálnych požiadaviek tiež môžeme pochopiť, ako sa systém správa pri rôznych vstupoch a výstupoch. Funkcionálne požiadavky špecifikujú, ako systém reaguje na rôzne vstupné a výstupné operácie [9].

- **Prihlásenie administrátora do NetBoxu:** Verifikovanie administrátora ústavu do NetBox-u, kde môže vykonávať operácie v rámci ústavu, kde je administrátorom.
- **Skenovanie zariadení v sieti:** Sken siete pomocou príkazu *nmap*, alebo jeho ekvivalentom. Skenovanie indentifikuje informácie ako IP a MAC adresa, typ zariadenia. Tieto údaje sú ukladané v JSON formáte pre importovanie do NetBoxu.
- **Importovanie skenovaných zariadení do NetBox-u:** Administrátor vie vykonávať CRUD operácie na základe importovaných údajov.
- **Spravovanie údajov:** Pre jednotlivé miestnosti, poschodia, špeciálne vlastnosti zariadení v NetBox-e.

2.1.2 Nefunkcionálne požiadavky

Nefunkcionálne požiadavky sú napríklad výpočtové požiadavky systému. Pri našom riešení zohľadňujeme nasledovné:

- **Dostupnosť systému:** Pre všetkých administrátorov, alebo super administrátora súčasne a časovo neobmedzene.
- **Jednoduché užívateľské rozhranie:** Musí byť intuitívne a prehľadné.

- **Výkonnosť:** Skript musí byť schopný naskenovať všetky zariadenia v sieti. NetBox musí byť schopný spracovať importované dáta, ako aj CRUD operácie.
- **Bezpečnosť:** Prístup do systému je chránený pomocou overenia administrátora (meno, heslo a prípadne ďalšie formy autentifikácie). Operácie v systéme a na jednotlivých ústavoch vie vykonávať len administrátorský účet s oprávnením.
- **Kompatibilita:** Systém musí byť kompatibilný s najnovšou verziou NetBox-u a operačnými systémami administrátorov.
- **Logovanie operácií a potencionálnych error hlásení:** História vykonaných operácií CRUD pre jednotlivé ústavy s časovou informáciou o vykonanej operácií. Zároveň tiež logovanie potencionálnych error hlásení.

2.2 Aké riešenia existujú na evidenciu siete

Existuje viacero softvérových nástrojov na evidenciu siete, vrátane:

- **NetBox:** Open-source nástroj na správu IP adresného priestoru a fyzickej i virtuálnej infraštruktúry siete.
- **SolarWinds Network Performance Monitor:** Komplexný komerčný nástroj na monitorovanie výkonu siete a správu zariadení.
- **ManageEngine OpManager:** Integrovaný nástroj na správu siete s podporou monitorovania, evidencie a analýzy.

2.3 Aké nástroje/príkazy existujú na scan siete

Na scanovanie siete sú často využívané tieto nástroje a príkazy:

- **Nmap:** Najrozšírenejší nástroj na zisťovanie aktívnych zariadení, otvorených portov, bežiacich služieb a ďalších sieťových parametrov. Poskytuje možnosť OS fingerprintingu, detekcie verzií služieb a dokonca aj zisťovania zraniteľností.
- **Angry IP Scanner:** Jednoduchý a rýchly nástroj vhodný na zisťovanie aktívnych IP adries v menších sieťach. Disponuje intuitívnym grafickým rozhraním.
- **Ping:** Základný príkaz na overenie dostupnosti zariadení. Slúži na odoslanie ICMP echo request paketov a vyhodnotenie odpovedí.

- **Traceroute:** Nástroj na sledovanie trasy paketov medzi zdrojom a cieľovým zariadením, ktorý pomáha identifikovať oneskorenia a problémové body.
- **Wireshark:** Pokročilý nástroj na analýzu sieťovej prevádzky, ktorý umožňuje zachytávať a interpretovať pakety v reálnom čase.
- **ZMap:** Špecializovaný nástroj na rýchle skenovanie veľkých IP rozsahov. Používa sa najmä v rozsiahlych výskumných projektoch.
- **SNMPwalk:** Príkazový nástroj na získavanie detailných informácií o zariadeniach pomocou protokolu SNMP. Je užitočný na diagnostiku a správu zariadení.
- **Netcat (nc):** Univerzálny nástroj na zisťovanie sieťových spojení a testovanie portov. Umožňuje manuálne kontrolovať komunikáciu medzi zariadeniami.

2.4 Výhody a nevýhody nami vybraného riešenia

Pre náš projekt sme sa rozhodli využiť **NetBox**. Toto open-source riešenie prináša množstvo výhod, ktoré z neho robia silného kandidáta na správu sieťových zariadení. Podpora API umožňuje jednoduchú integráciu s inými systémami, čo výrazne zjednodušuje automatizáciu procesov a zvyšuje efektivitu správy.

NetBox má tiež veľkú komunitu používateľov a rozsiahlu dokumentáciu, ktorá poskytuje užitočné zdroje pre riešenie problémov a implementáciu. Tieto faktory zaručujú, že používatelia majú prístup k najnovším aktualizáciám a môžu využiť osvedčené postupy od iných odborníkov.

Na druhej strane, NetBox má aj niekoľko nevýhod. Jeho inštalácia a konfigurácia vyžadujú určitú úroveň technických znalostí a skúseností, čo môže byť pre menej skúsených používateľov výzvou. Okrem toho si jeho prevádzka vyžaduje správu servera, čo znamená dodatočné náklady a úsilie na údržbu. Pre začiatočníkov môže byť rozhranie na prvý pohľad zložité, čo si môže vyžadovať istý čas na oboznámenie.

Napriek týmto nevýhodám je NetBox ideálnym riešením pre organizácie, ktoré hľadajú výkonný a prispôsobiteľný nástroj na správu siete. Jeho silné stránky prevažujú nad nedostatkami, najmä ak je dostupný technický personál schopný zvládnuť jeho implementáciu a údržbu.

2.5 Ciele práce

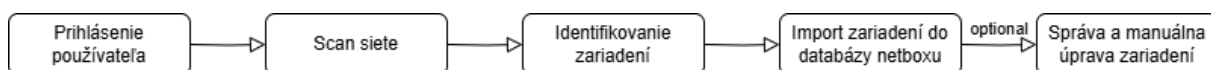
Hlavným cieľom tejto práce je navrhnúť a implementovať efektívny systém na správu a evidenciu sieťových zariadení. Takýto systém by mal byť schopný automatizovať procesy

spojené so scanovaním siete, čím sa odstránia manuálne zásahy a minimalizujú chyby pri zbere údajov o zariadeniach.

Okrem toho by mal tento systém umožniť jednoduchú a prehľadnú evidenciu všetkých zariadení, vrátane ich konfigurácií, fyzického umiestnenia a histórie zmien. Dôležitým aspektom je aj užívateľská prívetivosť systému, aby správa siete nebola zbytočne komplikovaná a umožňovala rýchly prístup k potrebným informáciám.

3 Návrh riešenia

Podľa funkcionálnych a nefunkcionálnych požiadaviek a na základe konzultácií so zadávateľmi projektu sme vypracovali návrh riešenia. Ako sme uviedli v analýze a cieľoch práce, používali sme open-source **NetBox**, ktorý sme rozšírili o vlastnú implementáciu - automatizovaný scan siete a evidenciu nájdených zariadení. Naše riešenie nie je plne automatizované a správne fungovanie si vyžaduje časť manuálneho zásahu zo strany administrátora siete (používateľa).



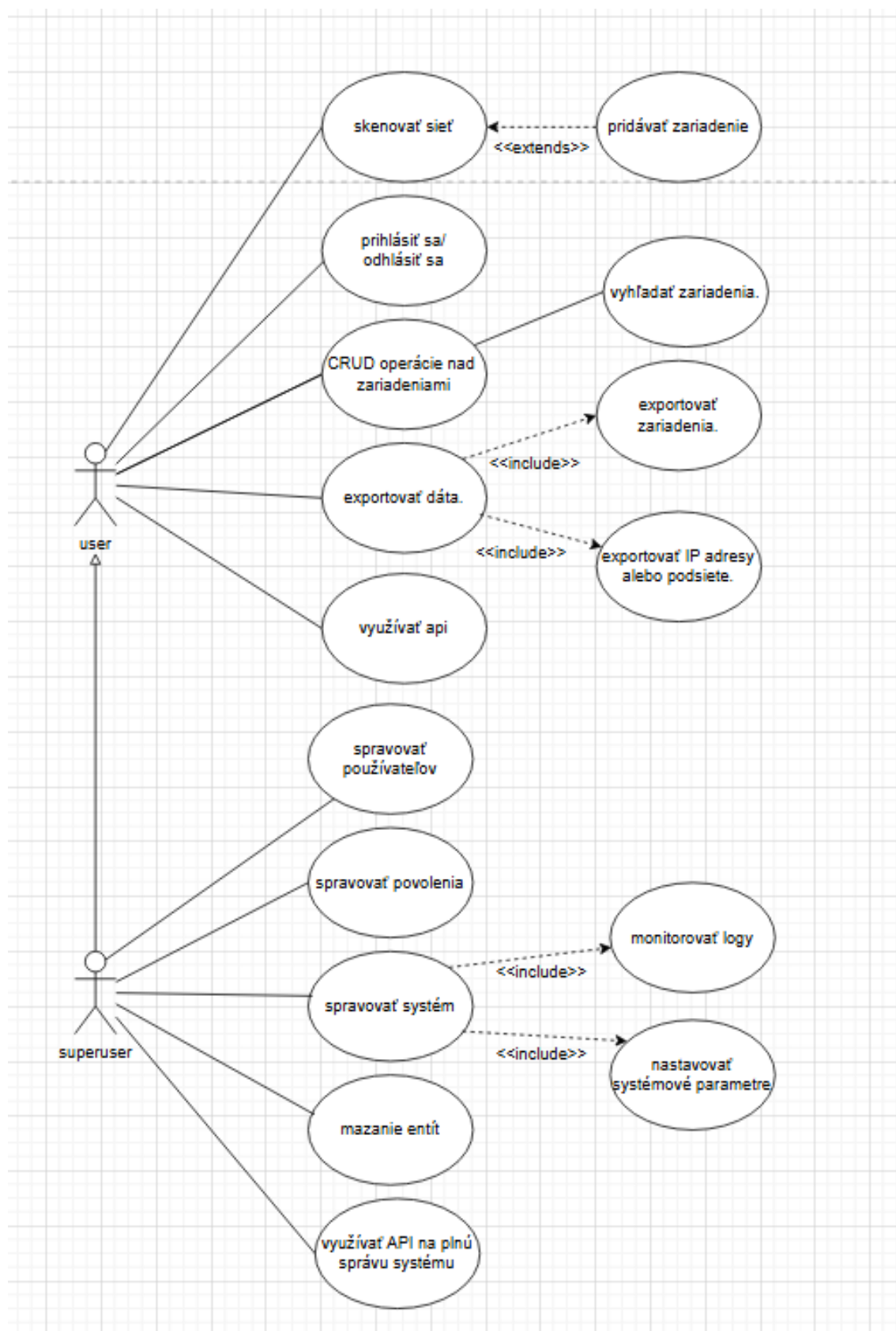
Obr. 1: Logika navrhovaného systému

Netbox umožňuje rozdelenie rolí používateľov, ktorí zároveň patria do určitej skupiny. V našom riešení sme použili rozdelenie na sieťových administrátorov - ide o bežného používateľa, ktorý bude podľa očakávaní správca siete a superadmina, ktorý má na starosti všetkých týchto administrátorov. Za skupiny považujeme takzvané tenanty, teda ústavy fakulty, ku ktorým sú administrátori priradený superadminom. Administrátor môže patriť aj do viacerých tenantov.

NAME	GROUP	DESCRIPTION	
IKAL	FEI Departments	Inštitút komunikácie a aplikovanej lingvistiky	✓ -
TIS	FEI Departments	Technologický inštitút športu	✓ -
UAMT	FEI Departments	Ústav automobilovej mechaniky	✓ -
UE	FEI Departments	Ústav elektrotechniky	✓ -
UEAE	FEI Departments	Ústav elektroenergetiky a aplikovanej elektrotechniky	✓ -
UEF	FEI Departments	Ústav elektrotechniky a fotoniky	✓ -
UIM	FEI Departments	Ústav informatiky a matematiky	✓ -
UJFI	FEI Departments	Ústav jadrového a fyzikálneho inžinierstva	✓ -
UMIKT	FEI Departments	Ústav multimediálnych informačných a komunikačných technológií	✓ -
URK	FEI Departments	Ústav robotiky a kybernetiky	✓ -

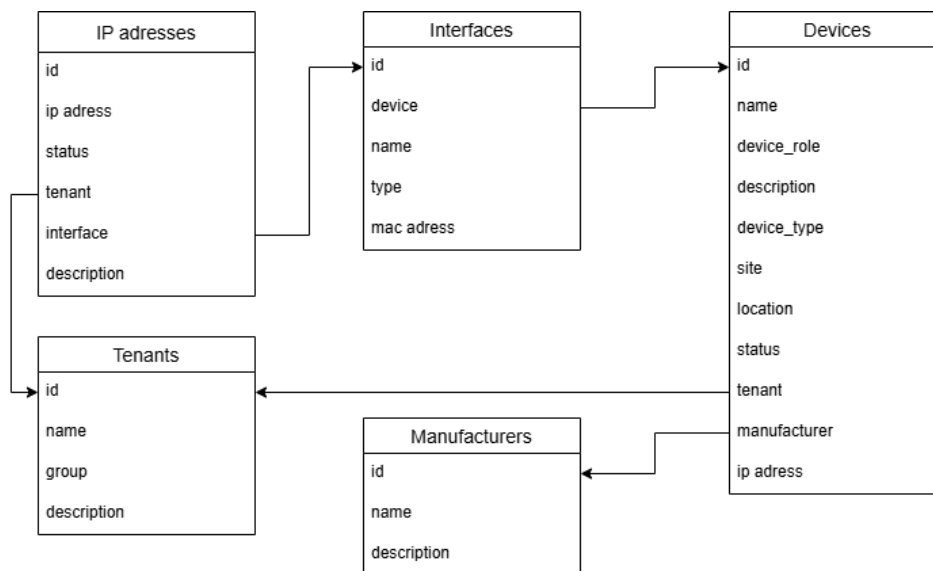
Obr. 2: Ústavy na fakulte

Najzákladnejšou a najjednoduchšou metódou vizualizácie funkcionality systému je use-case diagram. Tento diagram, ktorý možno voľne preložiť ako diagram prípadov použitia, zobrazuje správanie systému z používateľskej perspektívy. V use-case diagrame popisujeme funkcionality role bežného usera (administrátor siete daného ústavu) a role superadmina.

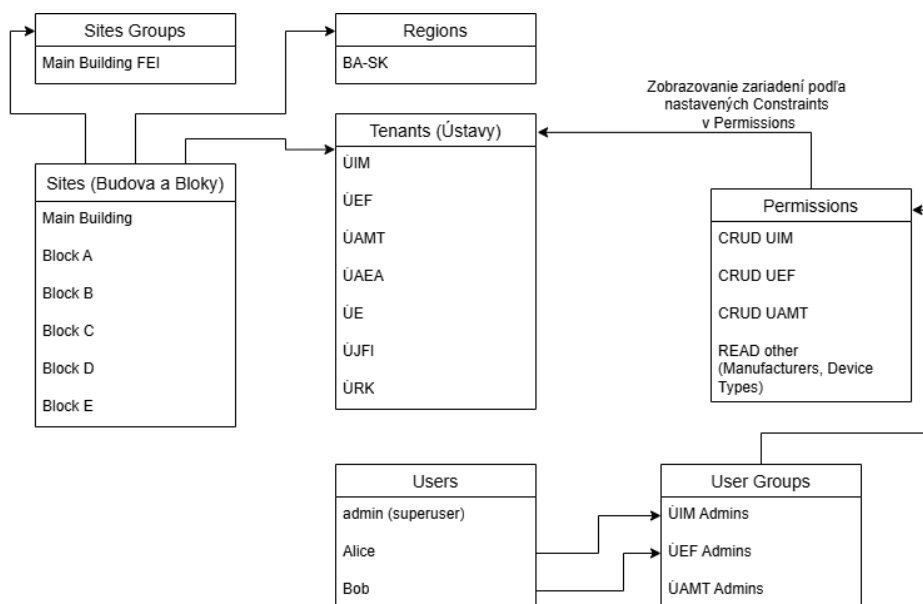


Obr. 3: Ústavy na fakulte

Následne sme preskúmali veľmi zložitú štruktúru databázy netboxu. Databázy netboxu vzhľadom na to ako komplexný nástroj netbox je, obsahuje stovky tabuliek. Uvádzame preto dva diagramy, zobrazujúce hlavné tabuľky reprezentujúce najdôležitejšie entity a ich vlastnosti, s ktorými pracujeme.



Obr. 4: Štruktúra databázy (prepojenie tabuliek ip address a device cez interface)



Obr. 5: Ukážka štruktúry používateľov, ich privilégií a priradenie k ústavu

4 Implementácia

Implementácia nášho riešenia sa zameriava na automatizáciu procesov skenovania siete a následnú evidenciu pripojených zariadení. Ako uvádzame v predošlých kapitolách, na evidenciu komponentov používame **Netbox**, ktorý sme rozšírili o nami implementovaný **Python** skript. Skript využíva open-source nástroje ako je nmap, a rozhranie **REST API** systému **Netbox** na zhromažďovanie, spracovanie a ukladanie informácií o zariadeniach v sieti.

Táto časť obsahuje podrobné vysvetlenie funkčnosti systému, jeho komponentov a logiky použitej na dosiahnutie automatizácie v požadovanom rozsahu podľa pokynov zadávateľov projektu. Celkový workflow skriptu môžeme zhrnúť nasledovne:

1. Validácia konfigurácie a API tokenov.
2. Skenovanie podsiete na nájdenie všetkých pripojených zariadení.
3. Načítanie ústavov, do ktorých používateľ, ktorý vykonáva scan patrí a následná výzva na výber ústavu, pod ktorý naskenované zariadenia budeme priradovať.
4. Vytvorenie objektu device, interface a ip_address pre každé nájdené zariadenie. V prípade nájdenia nových typov výrobcov, vytvorí sa aj object manufacturer. Každý objekt predstavuje riadok v prislúchajúcej tabuľke **NetBoxu**.
5. Aktualizovanie zariadenia.

4.1 Konfigurácia a autentifikácia

Na sprostredkovanie zabezpečenej interakcie s rozhraním **NetBox API**, skript používa premenné **NETBOX_URL** a **API_TOKEN**, ktoré sú importované z konfiguračného súboru (config.yml). Funkcia **validate_config()** overuje, či sú tieto parametre správne nastavené, skôr než sa pristúpi k ďalším operáciám. Ak je konfigurácia neplatná, skript ukončí vykonávanie, aby sa zabránilo chybám alebo neoprávnenému použitiu.

Skript umožňuje používateľovi priradiť zariadenia konkrétnym ústavom (podporuje aj možnosť priradiť zariadenie do skupiny „**no tenant**“, teda zariadenie nebude patriť pod konkrétny ústav a bude viditeľné pre všetkých používateľov). Táto funkcia je dôležitá pre organizáciu zariadení a priradenie ich vlastníka (ústav) prípadne konkrétnejšiu lokáciu ako je budova, blok, či miestnosť v rámci školy.

Funkcia **display_tenant()** načíta zoznam ústavov z rozhrania **NetBox API**. Používateľ (administrátor siete), ktorý vykonáva skenovanie na danej podsieti, má na výber pod aký ústav bude naskenované zariadenia priradovať. ID vybraného ústavu sa potom odovzdá nasledujúcim funkciám. Ak je načítaný zoznam ústavov prázdny, alebo sa volanie API nepodarí, skript ukončí vykonávanie s chybovou správou. Tým sa zabezpečí, že používateľ je upozornený na problémy a predíde sa chybnému priradeniu zariadení.

4.1.1 Funkcia **parse_arguments()**

Funkcia parsuje argumenty **nmap** skenu subnetu, ktoré používateľ zadá. Argument **-addr**, alebo **-address**, čo je IP adresa a **-o**, alebo **-output**.

4.1.2 Funkcia **validate_config()**

Načíta konfiguráciu a kontroluje, či konfigurácia obsahuje **api_token** a **base_url**. Ak nejaká hodnota chýba, alebo je prázdna, program vypíše varovnú správu a ukončí sa s výstupným kódom 1.

4.1.3 Funkcia **parse_subnet(provided_address)**

Funkcia so vstupným argumentom IP adresy kontroluje, či zadaná adresa je v správnom formáte. Ak nie, používateľ ju musí zadať v správnom formáte. Následne je rozdelená na dva **stringy**, jeden s jej adresou a druhý s jej prefixom. Ak je subnet v nesprávnom formáte, používateľ je varovaný výpisom do konzoly a program sa ukončí s výstupným kódom 1.

4.2 Skenovanie siete a spracovanie dát

Jednou zo základných funkcionalít skriptu je skenovanie siete, ktoré sa vykonáva pomocou nástroja **nmap**. Funkcia **run_nmap_scan(subnet)** vykoná skenovanie pomocou príkazu *nmap -sn ip_adresa_siete* na zadanej podsieti s cieľom zistiť aktívne zariadenia a ich metadáta, ako napríklad ip adresa, mac adresa, hostname, manufacturer, status a podobne. Tieto dáta sú ďalej spracované vo funkciách popísaných nižšie v dokumentácii.

4.2.1 Funkcia **main()**

Obsahuje hlavnú logiku programu. Postupne sa volajú jednotlivé funkcie pre spracovanie používateľských nastavení, ešte pred skenovaním po poradí:

- `args = parse_arguments()`

- `config = validate__config()`
- `subnet, prefix = parse__subnet(args.address)`

Následne sa volajú funkcie pre vykonanie skenu a *upload* dát do **NetBoxu**:

- `hosts = run__nmap__scan(subnet)`
- `hosts = include__host(hosts)`
- `hosts = upload(hosts, config, prefix)`

Na záver sa ešte dáta exportujú:

- `hosts = export(args.output, hosts, hosts)`

4.2.2 Funkcia `run__nmap__scan(subnet)`

Vykoná sa **nmap** na danom subnete pomocou funkcie `execute__nmap(subnet=subnet)`. Výstup z tejto funkcie sa následne parsuje pomocou `parse__nmap__xml(result)` aby boli extrahované informácie o hostovi (zariadení). Výstup je v **JSON** formáte a návratovou hodnotou funkcie je list obsahujúci nájdených **host**-ov s ich rozparsovanými vlastnosťami (napríklad IP, MAC, hostname atď.).

4.2.3 Funkcia `execute__nmap(subnet)`

Najskôr overí, či systém, z ktorého je spúšťaný sken má nainštalovaný **nmap**. Ak nie, používateľ je upozornený výstupom v konzole a program ukončí s výstupným kódom 1.

Ďalej zisťuje operačný systém používateľa, aby do príkazov pridal v prípade **macOS**, alebo **Linux**-u aj **sudo** privilégiá. Okrem spomínaných je podporovaný ešte **Windows**, inak program ukončí s varovnou správou do konzoly a s výstupným kódom 1.

Nakoniec sa spustí samotný **nmap** a ak nastane neočakávaná chyba, vypíše sa a program ukončí s kódom 1. Rovnako ak sa aj po prvotnom overení **nmap** nenájde, alebo nie je prístupný, tak sa do konzoly vypíše príslušná hláška a výstupný kód. Analogicky pre situáciu, keď je sken prerušený používateľom sa program správa rovnako.

4.2.4 Funkcia `parse__nmap__xml(xml_data, default_vendor)`

Spracováva **stdout** skenu **nmap** vo formáte XML. Extrahuje relevantné dáta - IP a MAC adresy, výrobcu a status. Návratová hodnota funkcie je zoznam vlastností nájdeného hosta s ID, interface ID, ID ip adresy a extrahované dáta.

4.2.5 Funkcia `include__host(hosts)`

Identifikuje hostov zo vstupných dát funkcie. Ak sa nejaké IP adresy zhodujú s IP adresou používateľa, používateľ je vyzvaný k výberu, či danú adresu chce obsiahnuť vo výsledku. Na zistenie lokálnej IP adresy sa volá pomocná funkcia `get__local__ips()`. Návratová hodnota funkcie je odfiltrovaný list adries **bez local host**, ak si tak používateľ vyberie.

4.2.6 Funkcia `get__local__ips()`

Získa všetky lokálne **IPv4** adresy asociované s hostom. Najprv získa hlavnú IP adresu pomocou `socket.gethostbyname()` a následne všetky ostatné pomocou `socket.getaddrinfo()`.

Návratová hodnota sú unikátne lokálne IP adresy.

4.2.7 Funkcia `upload(hosts, config, prefix)`

Funkcia **nahrá** naskenované zariadenia do NetBoxu, podľa **config** nastavení. Používateľ je pri **nahrávaní** vedený výstupmi v konzole k správne postupovaniu. Funkcia ponúka tri módy: **nenahrávať** dáta vôbec, **nahrание všetkých zariadení**, alebo **individuálny výber zariadení**, ktoré si želá nahráť.

Následne sa vytvárajú potrebné objekty v NetBoxe, na ktoré sa dáta namapujú. Priebeh je nasledovný:

- `tenant = display__tenants(config)`
- `site = display__sites(config)`
- `selected_hosts = create__manufacturers(selected_hosts, config)`
- `selected_hosts = create__device__types(selected_hosts, config)`
- `selected_hosts = create__devices(selected_hosts, tenant, site, config)`
- `selected_hosts = create__interfaces(selected_hosts, config)`
- `selected_hosts = create__addresses(selected_hosts, tenant, config, prefix)`
- `update__device(selected_hosts, config)`

Návratová hodnota je zoznam všetkých nahraných zariadení. Ak sa ich úspešne podarilo namapovať, budú mať zároveň príslušné **NetBox device ID**. Tie, ktoré sa namapovať nepodarilo toto ID mať nebudú.

4.2.8 Funkcia `export(cli_output, all_hosts, uploaded_hosts)`

Používateľ má opäť na výber z troch módov: **neexportovať dáta**, **exportovať všetky dáta**, alebo **exportovať iba nahrané zariadenia**. Pre samotné exportovanie je zavolaná pomocná funkcia `export_hosts_to_csv(hosts, filename, include_ids=False)`.

4.2.9 Funkcia `export_hosts_to_csv(hosts, filename)`

Funkcia exportuje všetky detaily o hostoch do súboru `.csv`. Používateľ si môže vybrať **ID zariadenia**, **interface-u**, alebo **IP adresy** ako súčasť exportovaných dát (jednotlivé ID sa zhodujú s ID v **NetBoxe**. Ak sa rozhodne si vybrať importovanie dát bez ID, importujú sa **ip**, a **mac adresa**, **manufacturer**, **status** a **hostname**. Zároveň si používateľ špecifikuje názov súboru, do ktorého sa tieto dáta uložia.

4.2.10 Funkcia `create_devices(hosts, tenant, site, config)`

Pre každé nájdené zariadenie počas skenovania používateľom zadanej siete sa skript pokúsi vytvoriť zodpovedajúci záznam zariadenia v systéme **NetBox**. Ide o objekt s rôznymi parametrami (stĺpce v tabuľke zariadení), ako napríklad názov zariadenia, rola, typ, status, ústav, prípadne konkrétna lokácia, kde sa zariadenie nachádza. Ak **hostname** nie je k dispozícii, ako názov zariadenia sa použije adresa MAC. Objektu `device` sa odošle do systému **NetBox** cez **POST request** a v podobe **response** sa nám vráti ID daného zariadenia (objektu `device`), keďže ID zariadenia je generované priamo systémom **NetBox**.

4.2.11 Funkcia `create_interfaces(hosts, config)`

Každému vytvorenému zariadeniu je priradené predvolené rozhranie (interface), ktoré prepája tabuľku ip adresy a zariadenia v systéme **NetBox**. Interface okrem iného obsahuje atribúty ako id zariadenia, meno, typ a mac adresu. s názvom „eth0“. **POST request** a **response** s vygenerovaným ID funguje obdobne ako pri predošlej funkcii.

4.2.12 Funkcia `create_addresses(hosts, tenant, config, prefix)`

Po vytvorení zariadení a rozhraní sa rozhraniam priradia IP adresy. Funkcia `create_address()` vytvorí objekt s parametrami ako adresa, status, ústav, či interface, s ktorým bude záznam v tabuľke ip adresy prepojený. **POST request** a **response** s vygenerovaným ID funguje opäť na rovnakom princípe, teda vygenerované ID si späť načítame a priradíme k nášmu objektu, ktorý reprezentuje ip adresu.

4.2.13 Funkcia `create__manufacturers(selected__hosts, config)`

Namapuje výrobcu daným hostom. Ak výrobca existuje, priradí hostovi ID tohto výrobcu. Ak výrobca ešte nie je evidovaný, vytvorí ho. Návratovou hodnotou funkcie je zoznam o informáciách o hostovi s pridaným `manufacturer__id`.

4.2.14 Funkcia `create__device__types(hosts, config)`

V závislosti od `manufacturer__id` a `manufacturer` sa pridáva typ zariadenia. Ak daný typ zariadenia ešte neexistuje, vytvorí sa. Návratová hodnota je opäť zoznam s informáciami o hostoch, s doplneným polom `device__type__id`.

4.2.15 Funkcia `update__devices(hosts, config)`

Po vytvorení všetkých komponentov (device - zariadenie, interface - rozhranie a ip_addr - adresa IP) skript aktualizuje atribút primárna IP adresa zariadenia v systéme **NetBox** pomocou funkcie `update_device()`. Týmto spôsobom sa prepojí tabuľka v systéme **NetBox**. Tabuľka ip adries sa namapuje na tabuľku interface, ktorá je analogicky prepojená s tabuľkou devices. Tým sa zabezpečí, že zariadenie je prepojené so svojou IP adresou v databáze **NetBoxu**.

5 Nasadenie a testovanie

Jedným z cieľov projektu bolo otestovať riešenie a nasadiť ho na fakultnú (pod)sieť. Keďže v rovnakom čase práve prebiehala rekonštrukcia fakulty a zmeny v sieti, dohodli sme sa so zadávateľmi projektu, že dané riešenie budeme testovať len na menšej podsieti v rámci serverovne, ku ktorej mal prístup sieťový administrátor UIM. Po úspešnej implementácii a testovaní na lokálnej domácej sieti, sme sa spojili s administrátorom siete UIM. Na jednom zo serverov (počítačov) v podsieti UIM na bloku A sme nainštalovali a spustili **NetBox**. Pre zabezpečenie neustálej dostupnosti **NetBoxu** sme spustili na počítači **Nginx** ako službu, ktorá na pozadí sprístupňuje **NetBox** cez **HTTPS**. Keďže sa tento počítač nachádza v zamknutej serverovni, dohodli sme si s administrátorom prístup cez ssh. **NetBox** je prístupný po prihlásení na adrese 147.175.106.21.

5.1 Testovanie

Na otestovanie sme sa spojili s druhým sieťovým administrátorom UJFI, ktorý nám počas osobného stretnutia dovolil naskenovať rozsah 128 IP adries na podsieti, ktorú spravuje, čím nám umožnil získať viac relevantných údajov a zariadení. Počas testovania nášho riešenia na UJFI sme odhalili niekoľko problémov:

- **Manufacturers a device types:** náš skript nesprávne mapoval výrobcov ku jednotlivým zariadeniam. Výrobcov sme počas skenovania našli, ale v **NetBoxe** boli pri všetkých zariadeniach napevno vložené „**Unspecified**“ hodnoty. Zároveň sme si uvedomili, že sme nesprávne pochopili pojem **device type** (považovali sme ho za totožný s **device role**, avšak zistili sme, že typ zariadenia súvisí s výrobcom, presnejšie typ zariadenia predstavuje konkrétny model, ktorý ale my zistiť príkazom **nmap** nevieme. Pre zjednodušenie sme sa rozhodli reprezentovať typ zariadenia totožný s výrobcom. Pre pochopenie, pod výrobcom si môžeme predstaviť výrobcu **Asus** a pod typom zariadenia konkrétny model napríklad počítaču/notebooku. **Vyriešené opravou implementácie.**
- **Slug pre manufacturer a device type:** počas testovania na UJFI sme objavili nových výrobcov. Noví výrobcovia jednotlivých zariadení obsahovali v názve zakázané znaky, teda znaky, ktoré nebolo možné použiť v **NetBoxe** pri vytváraní tzv. **slug** pre výrobcu. **Slug** je krátky identifikátor vhodný na použitie v **URL** a môže obsahovať iba malé písmená, čísla, podtržník a spojovník. Všetky nájdené zakázané znaky sme pred vkladáním do **NetBoxu** zmenili na spojovníky. **Vyriešené opravou**

implementácie.

- **Formát IP adresy:** zistili sme problém s ukladaním IP adresy do **NetBoxu**. Do **NetBoxu** sa ukladá ip adresa v CIDR formáte ipv4/xx, teda aj s prefixom. V našom kóde sme napevno vkladali ip adresu v CIDR formáte ipv4/24 bez ohľadu na to, aký prefix bol použitý pri skenovaní. **Vyriešené opravou implementácie.**
- **Selektívny import:** počas testovania sme dostali spätnú odozvu od administrátora s požiadavkou, či by bolo možné z pohľadu používateľa(administrátora siete) zvoliť, ktoré z naskenovaných zariadení budeme vkladať do **NetBoxu**. Rozhodli sme sa upraviť implementáciu tak, aby mal administrátor po naskenovaní všetkých zariadení na výber, či chce nahráť všetky zariadenia do **NetBoxu**, alebo bude môcť jednotlivo vybrať konkrétne zariadenia. Tak isto sme umožnili pri každom zariadení vybrať individuálne typ zariadenia spomínaný vyššie. **Vyriešené opravou implementácie.**
- **Výpisy:** počas testovania niektoré z výpisov neboli prehľadné a chýbali počty ako napríklad počet naskenovaných zariadení, alebo počet úspešne importovaných zariadení do **NetBoxu**. **Vyriešené opravou implementácie.**
- **Viacero IP pre jednu MAC:** pri skenovaní podsiete na UJFI sme našli zariadenia, ktoré disponovali jednou **MAC** adresou, ale boli im predelené dve/viacero rôznych **IP** adries, čo pri nahrávaní zariadení do netboxu spôsobovalo problémy. Nahrávanie aktuálne funguje tak, že zariadenie je importované do netboxu s prvou ip adresou, ktorá k danému zariadeniu respektíve k **MAC** adrese prislúcha. V exportovanom .csv súbore je viditeľné, ktoré zariadenie malo viacero ip adries. Keďže sme testovanie robili až na záver semestra, z časového hľadiska sme nedokázali ošetriť túto chybu. **Zaznamenané, pridané do zoznamu možných vylepšení/rozšírení.**

Po odhalení a oprave chýb, sme skenovanie spustili ešte raz v sieti UIM cez počítač, na ktorom sme mali zriadený ssh prístup. Pri skenovaní 256 (resp. 254) ip adries sme našli 26 zariadení, ktorých dáta sme rozparsovali, vypísali, zvolili nahratie všetkých zariadení do netboxu a jednotlivo priradili rolu každého zariadenia. Ukážka nahratých zariadení do **NetBoxu** vyzerá nasledovne:

<input type="checkbox"/>	NAME	STATUS	TENANT ×	SITE	LOCATION	RACK	ROLE	MANUFACTURER	TYPE	IP ADDRESS	
<input type="checkbox"/>	_gateway	Active	UIM	Block A	—	—	Router	Cisco Systems, Inc	Cisco Systems, Inc	147.175.106.1/24	
<input type="checkbox"/>	uim.elf.stuba.sk	Active	UIM	Block A	—	—	Web Server	Unspecified	Unspecified	147.175.106.2/24	
<input type="checkbox"/>	AC:1F:6B:0B:1C:66	Active	UIM	Block A	—	—	Desktop PC	Super Micro Computer	Super Micro Computer	147.175.106.8/24	
<input type="checkbox"/>	F4:6D:04:2E:6E:F1	Active	UIM	Block A	—	—	Desktop PC	Asustek Computer	Asustek Computer	147.175.106.10/24	
<input type="checkbox"/>	BC:EE:7B:9E:D0:0A	Active	UIM	Block A	—	—	Desktop PC	Asustek Computer	Asustek Computer	147.175.106.22/24	
<input type="checkbox"/>	90:1B:0E:0A:08:E5	Active	UIM	Block A	—	—	Desktop PC	Fujitsu Technology Solutions GmbH	Fujitsu Technology Solutions GmbH	147.175.106.23/24	
<input type="checkbox"/>	90:09:D0:04:6B:4F	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.125/24	
<input type="checkbox"/>	00:11:32:F5:C9:CD	Active	UIM	Block A	—	—	Network storage	Synology Incorporated	Synology Incorporated	147.175.106.126/24	
<input type="checkbox"/>	B6:B1:71:75:28:16	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.128/24	
<input type="checkbox"/>	08:BF:B8:74:C6:E2	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.145/24	
<input type="checkbox"/>	B8:27:EB:2E:94:2F	Active	UIM	Block A	—	—	Single Board Computers	Raspberry Pi Foundation	Raspberry Pi Foundation	147.175.106.165/24	
<input type="checkbox"/>	08:00:27:A6:E7:D2	Active	UIM	Block A	—	—	Virtual Machine	Oracle VirtualBox virtual NIC	Oracle VirtualBox virtual NIC	147.175.106.166/24	
<input type="checkbox"/>	58:11:22:B6:C4:D0	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.171/24	
<input type="checkbox"/>	04:7C:16:C1:9A:3D	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.172/24	
<input type="checkbox"/>	50:EB:F6:77:64:56	Active	UIM	Block A	—	—	Desktop PC	ASUSTek COMPUTER INC.	ASUSTek COMPUTER INC.	147.175.106.181/24	
<input type="checkbox"/>	04:7C:16:C1:9A:13	Active	UIM	Block A	—	—	Unspecified	Unspecified	Unspecified	147.175.106.187/24	
<input type="checkbox"/>	1C:1B:0D:07:69:3C	Active	UIM	Block A	—	—	Desktop PC	Giga-byte Technology	Giga-byte Technology	147.175.106.188/24	
<input type="checkbox"/>	08:BF:B8:03:4E:94	Active	UIM	Block A	—	—	Desktop PC	Unspecified	Unspecified	147.175.106.190/24	

Obr. 6: Importované zariadenia v NetBoxe, UIM, Blok A

Okrem zariadení sa do **NetBoxu** uložil aj zoznam **IP adries**, **Device types**, **Manufacturers** a jednotlivým zariadeniam boli priradené ich role, ktoré sme v **NetBoxe** manuálne definovali, ešte pred začiatkom skenovania. Nahraté dáta vizualizujeme na obrázkoch nižšie:

<input type="checkbox"/>	ID ▼	× NAME	DEVICES	VMS	COLOR
<input type="checkbox"/>	1	Firewall	0	0	
<input type="checkbox"/>	2	Access Point	0	0	
<input type="checkbox"/>	3	Desktop PC	34	0	
<input type="checkbox"/>	4	Router	1	0	
<input type="checkbox"/>	5	Web Server	1	0	
<input type="checkbox"/>	6	Switch	0	0	
<input type="checkbox"/>	7	Cable Related	0	0	
<input type="checkbox"/>	8	Unspecified	11	0	
<input type="checkbox"/>	9	Network storage	1	0	
<input type="checkbox"/>	10	Virtual Machine	1	0	
<input type="checkbox"/>	11	Single Board Computers	1	0	

Obr. 7: Dostupné role pre zariadenia

<input type="checkbox"/>	ID ▾ ×	DEVICE TYPE	MANUFACTURER	PART NUMBER	U HEIGHT	FULL DEPTH	INSTANCES	
<input type="checkbox"/>	4	Desktop PC	Unspecified	—	0	✓	24	
<input type="checkbox"/>	13	Unspecified	Unspecified	—	0	✓	13	
<input type="checkbox"/>	89	Cisco Systems, Inc	Cisco Systems, Inc	—	1	✓	1	
<input type="checkbox"/>	90	Super Micro Computer	Super Micro Computer	—	1	✓	1	
<input type="checkbox"/>	91	Asustek Computer	Asustek Computer	—	1	✓	4	
<input type="checkbox"/>	92	Fujitsu Technology Solutions GmbH	Fujitsu Technology Solutions GmbH	—	1	✓	1	
<input type="checkbox"/>	93	Synology Incorporated	Synology Incorporated	—	1	✓	1	
<input type="checkbox"/>	94	Raspberry Pi Foundation	Raspberry Pi Foundation	—	1	✓	1	
<input type="checkbox"/>	95	Oracle VirtualBox virtual NIC	Oracle VirtualBox virtual NIC	—	1	✓	1	
<input type="checkbox"/>	96	ASUSTek COMPUTER INC.	ASUSTek COMPUTER INC.	—	1	✓	2	
<input type="checkbox"/>	97	Giga-byte Technology	Giga-byte Technology	—	1	✓	1	

Obr. 8: Importované typy zariadení podľa výrobcov

<input type="checkbox"/>	NAME	DEVICE TYPES	MODULE TYPES	INVENTORY ITEMS	PLATFORMS	DESCRIPTION	SLUG	
<input type="checkbox"/>	Asustek Computer	1	0	0	0	—	asustek-computer	
<input type="checkbox"/>	ASUSTek COMPUTER INC.	1	0	0	0	—	asustek-computer-inc	
<input type="checkbox"/>	Cisco Systems, Inc	1	0	0	0	—	cisco-systems-inc	
<input type="checkbox"/>	Fujitsu Technology Solutions GmbH	1	0	0	0	—	fujitsu-technology-solutions-gmbh	
<input type="checkbox"/>	Giga-byte Technology	1	0	0	0	—	giga-byte-technology	
<input type="checkbox"/>	Oracle VirtualBox virtual NIC	1	0	0	0	—	oracle-virtualbox-virtual-nic	
<input type="checkbox"/>	Raspberry Pi Foundation	1	0	0	0	—	raspberry-pi-foundation	
<input type="checkbox"/>	Super Micro Computer	1	0	0	0	—	super-micro-computer	
<input type="checkbox"/>	Synology Incorporated	1	0	0	0	—	synology-incorporated	
<input type="checkbox"/>	Unspecified	2	0	0	0	Assign when manufacturer is unknown	unspecified	

Obr. 9: Importovaní výrobcovia

<input type="checkbox"/>	IP ADDRESS	VRF	STATUS	ROLE	TENANT	ASSIGNED	DNS NAME	DESCRIPTION	
<input type="checkbox"/>	147.175.106.1/24	Global	Active	—	UIM	✓	—	40:14:82:D6:AD:3D	
<input type="checkbox"/>	147.175.106.2/24	Global	Active	—	UIM	✓	—	18:FD:74:FE:E5:40	
<input type="checkbox"/>	147.175.106.8/24	Global	Active	—	UIM	✓	—	AC:1F:6B:0B:1C:66	
<input type="checkbox"/>	147.175.106.10/24	Global	Active	—	UIM	✓	—	F4:6D:04:2E:6E:F1	
<input type="checkbox"/>	147.175.106.21/24	Global	Active	—	UIM	✓	—	—	
<input type="checkbox"/>	147.175.106.22/24	Global	Active	—	UIM	✓	—	BC:EE:7B:9E:D0:0A	
<input type="checkbox"/>	147.175.106.23/24	Global	Active	—	UIM	✓	—	90:1B:0E:0A:08:E5	
<input type="checkbox"/>	147.175.106.125/24	Global	Active	—	UIM	✓	—	90:09:D0:04:6B:4F	
<input type="checkbox"/>	147.175.106.126/24	Global	Active	—	UIM	✓	—	00:11:32:F5:C9:CD	
<input type="checkbox"/>	147.175.106.128/24	Global	Active	—	UIM	✓	—	B6:B1:71:75:28:16	
<input type="checkbox"/>	147.175.106.145/24	Global	Active	—	UIM	✓	—	08:BF:B8:74:C6:E2	
<input type="checkbox"/>	147.175.106.165/24	Global	Active	—	UIM	✓	—	B8:27:E8:2E:94:2F	
<input type="checkbox"/>	147.175.106.166/24	Global	Active	—	UIM	✓	—	08:00:27:A6:E7:D2	
<input type="checkbox"/>	147.175.106.171/24	Global	Active	—	UIM	✓	—	58:11:22:B6:C4:D0	

Obr. 10: Importované IP adresy

6 Návod na inštaláciu programu

Skript, ktorý sme implementovali na automatizáciu skenovania sa potrebuje pripojiť do **NetBoxu**. Pre školské účely je **NetBox** už nainštalovaný na dostupnom počítači a sprístupnený cez ip adresu 147.175.106.21, avšak pre prípadnú budúcu inštaláciu prikľadáme aj link na inštaláciu. Taktiež je potrebné nainštalovať niekoľko závislostí, aby mohol náš program fungovať správne:

6.1 Požiadavky

- Operačný systém: Linux (napr. Ubuntu 22.04) alebo Windows
- Python 3.10 alebo novší
- NetBox (verzia 4.1.4 alebo novšia) - podrobná inštalácia samotného NetBoxu je prístupná tu: <https://netboxlabs.com/docs/netbox/en/stable/installation/3-netbox/>
- Nástroj nmap

6.2 Inštalácia

1. Naklonujte repozitár (prípadne rozbaľte zip, ktoré prikľadáme ako prílohu k práci):

```
git clone https://github.com/Faraon40/UniNetInsight
```

2. Presuňte sa do priečinka UniNetInsight

```
cd UniNetInsight
```

3. Vytvorte a aktivujte virtuálne prostredie:

```
# Linux
python3 -m venv venv
source venv/bin/activate
```

```
# Windows
```

```
python -m venv venv
venv\Scripts\activate
```

4. Nainštalujte „requirements“:

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

5. Nainštalujte `nmap`:

```
# Linux
```

```
sudo apt install nmap
```

```
# Windows
```

```
Stiahnite z: https://nmap.org/download.html
```

6. Nastavte konfiguračný súbor `configs/config.yml` (v prípade, že nemáte vygenerovaný token alebo vytvorený účet, postupujte podľa návodu na obsluhu NetBoxu v kapitole 7:

```
api_token: <váš_token>
```

```
base_url: <ip adresa netboxu> # aktuálne: https://147.175.106.21/
```

7 Návod na obsluhu Netbox-u

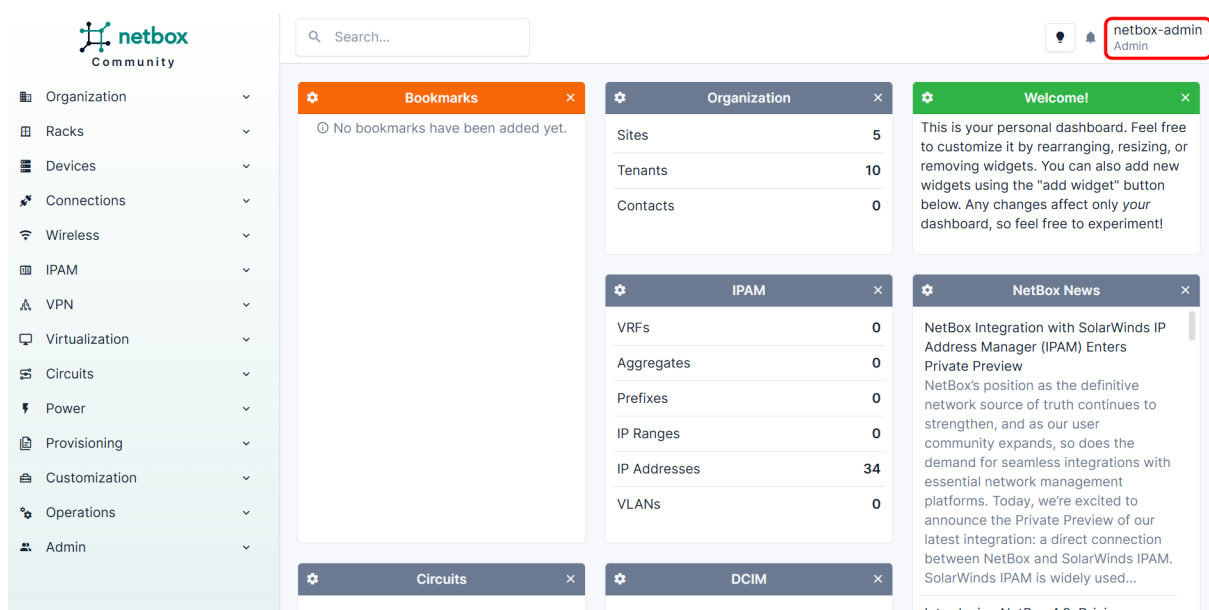
Tento návod slúži ako príručka na obsluhu **NetBoxu** z pozície super administrátora a bežného používateľa (správca siete na danom ústave)

7.1 Pridávanie nových používateľov

Vytváranie nových používateľov v NetBoxe je akcia, ktorú môže vykonať iba **super administrator**. Ostatní používatelia nemajú oprávnenie na vytvorenie, mazanie či modifikáciu práv iných účtov.

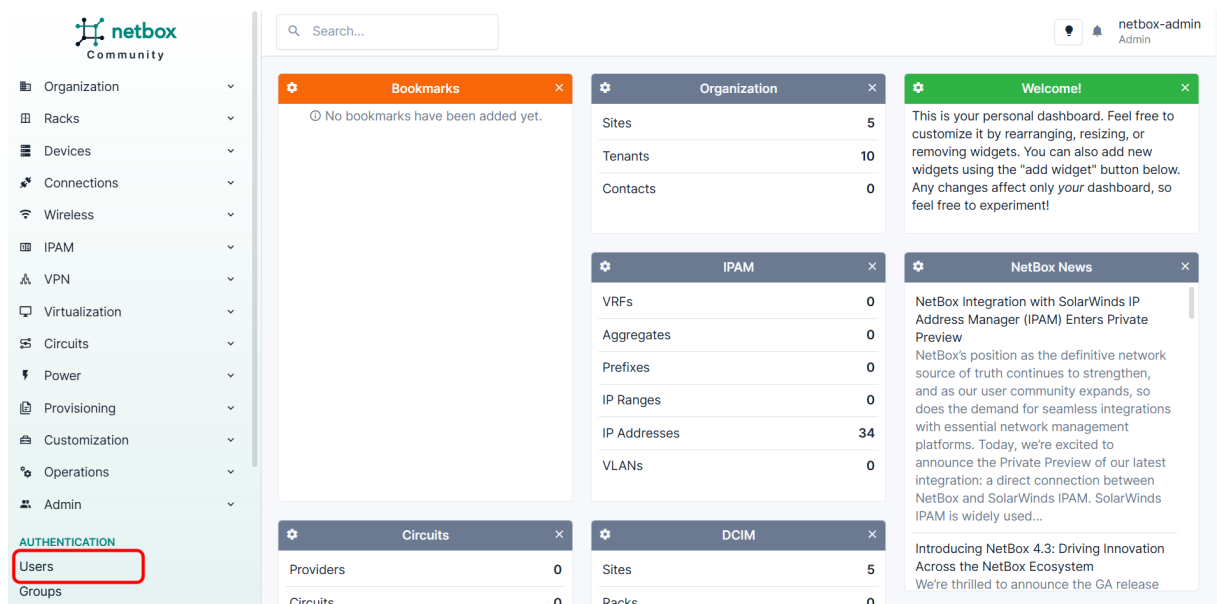
7.1.1 Prihlásenie a navigácia

Po otvorení webovej aplikácie NetBox sa najskôr prihláste ako superadmin. Po úspešnom prihlásení by ste mali v pravom hornom rohu vidieť svoje používateľské meno, ktoré potvrdzuje, že máte plné administrátorské práva.



Obr. 11: Prihlásený používateľ

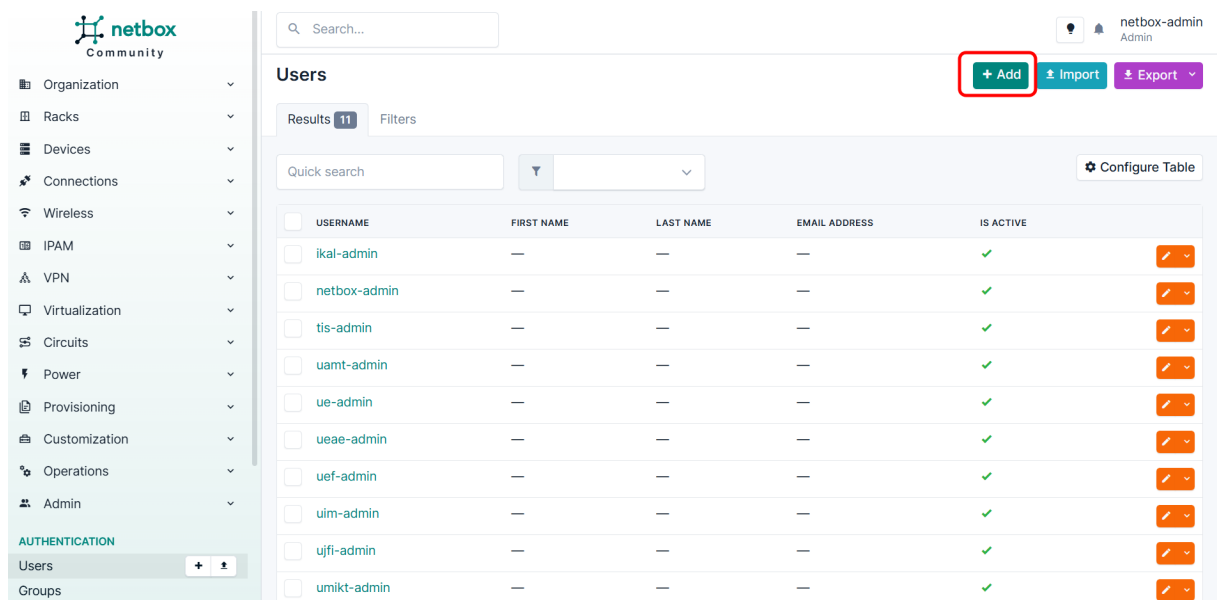
Po prihlásení prejdite v ľavom navigačnom menu do sekcie s názvom *Admin*. V rámci nej kliknite na možnosť *Users*, ktorá vás presmeruje na stránku so zoznamom všetkých aktuálne vytvorených používateľov.



Obr. 12: Navigácia k používateľom

7.1.2 Vytvorenie používateľa

Na stránke so zoznamom používateľov sa v pravom hornom rohu nachádza tlačidlo *Add*, ktorým spustíte proces vytvárania nového účtu. Po jeho stlačení sa zobrazí formulár s viacerými poľami, ktoré je potrebné vyplniť.



Obr. 13: Tabuľka používateľov

Do poľa *Username* zadajte prihlasovacie meno používateľa. Nižšie zadajte heslo v poliach *Password* a *Confirm password*. Môžete tiež zadať meno, priezvisko a e-mailovú adresu, ktoré sú voliteľné, ale odporúčané kvôli prehľadnosti.

Dôležité sú najmä tri zaškrŕavacie políčka:

- **Staff status:** povoľuje prístup do administrátorského rozhrania.
- **Superuser status:** povoľuje plný prístup vrátane správy používateľov (zaškrŕavajte iba ak má byť nový používateľ tiež superadminom).
- **Active:** musí byť zaškrŕnuté, aby účet mohol byť použitý.

V prípade, že účet nemá status superusera, ale má mať prístup k určitým funkcionalitám, je možné mu dodatočne priradiť oprávnenia alebo skupiny v sekcii *Groups* alebo *Permissions*. Na nasledujúcom obrázku možno vidieť, že sme do políčka *Groups* vybrali dve skupiny - *UIM Administrators* a *ALL Admins*. To znamená, že daný používateľ, bude do týchto dvoch skupín patriť. Skupina *UIM Administrators* sú správcovia siete prislúchajúcej *UIM* tenantu a skupina *ALL Admins* je skupina, do ktorej zaraďujeme všetkých administrátorov všetkých tenantov - pokrýva všeobecné oprávnenia administrátorov bez ohľadu na to, k akému tenantu patria.

User

Username* jozko-uim-admin
Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password*

Confirm password*
Enter the same password as before, for verification.

First name Jožko

Last name Mrkvička

Email address jm@example.com

Groups

Groups UIM Administrators x ALL Admins x

Status

☒ **Active**
Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

☐ **Staff status**
Designates whether the user can log into this admin site.

☐ **Superuser status**
Designates that this user has all permissions without explicitly assigning them.

Permissions

Permissions

Cancel **Create** Create & Add Another

Obr. 14: Formulár pre vytvorenie nového používateľa

7.1.3 Uloženie a aktivácia

Po vyplnení všetkých potrebných údajov kliknite na tlačidlo *Create* v dolnej časti formulára. Ak boli všetky povinné polia vyplnené správne, nový účet sa okamžite uloží a zobrazí v zozname používateľov. Odporúča sa, aby ste nového používateľa ihneď informovali o vytvorení účtu a dočasnom hesle.

7.2 Správa oprávnení - vytvorenie, priradenie a použitie

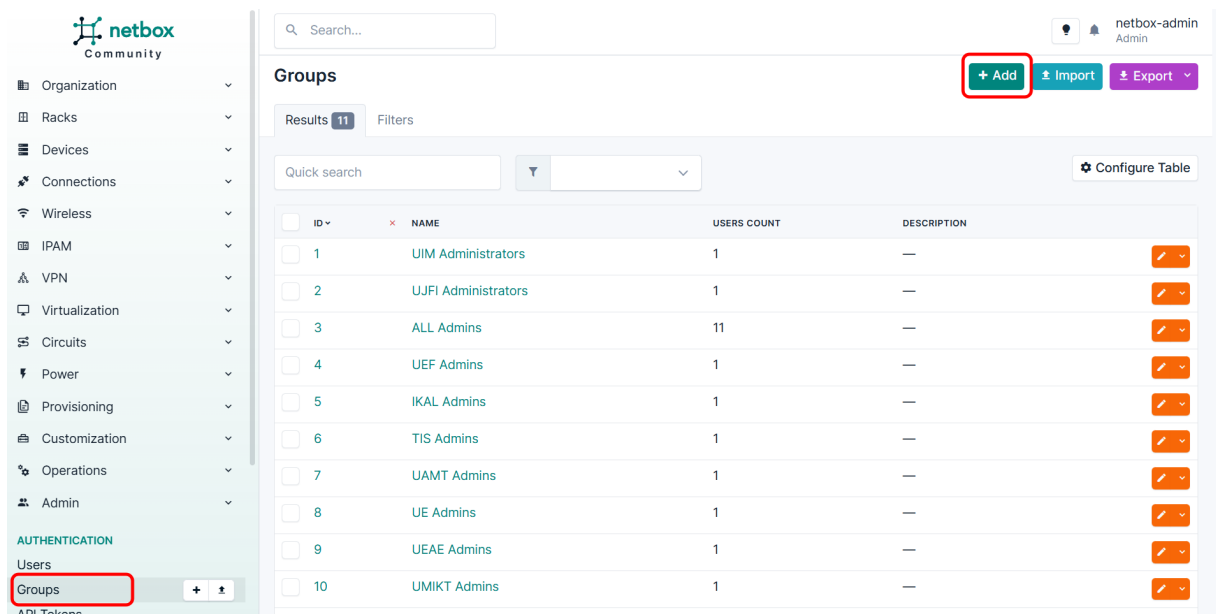
NetBox využíva systém správy oprávnení *Permissions*, ktorý je postavený na modeli používateľ – skupina – oprávnenie. Oprávnenia určujú, čo môže konkrétny používateľ

robiť – napríklad, či môže čítať, upravovať alebo odstrániť záznamy v databáze zariadení, lokalít alebo IP adries. Oprávnenia je možné priradovať priamo jednotlivým používateľom, alebo efektívnejšie – prostredníctvom skupín. V tejto časti dokumentácie si ukážeme, ako z pozície superadmina vytvoriť oprávnenie, pridať ho do skupiny a priradiť túto skupinu konkrétnemu používateľovi.

7.2.1 Vytvorenie novej skupiny

Najprv sa prihláste do NetBoxu ako superadmin. V ľavom menu prejdite do sekcie *Admin* a následne kliknite na možnosť *Groups*.

Na stránke so zoznamom existujúcich skupín kliknite v pravom hornom rohu na tlačidlo *Add*. Zobrazí sa jednoduchý formulár, kde vyplníte názov novej skupiny - napr. *Administrátori Ústavu X*. Po vyplnení kliknite na *Create*.



ID	NAME	USERS COUNT	DESCRIPTION
1	UIM Administrators	1	—
2	UJFI Administrators	1	—
3	ALL Admins	11	—
4	UEF Admins	1	—
5	IKAL Admins	1	—
6	TIS Admins	1	—
7	UAMT Admins	1	—
8	UE Admins	1	—
9	UEAE Admins	1	—
10	UMIKT Admins	1	—

Obr. 15: Tabuľka skupín

netbox Community

Search...

netbox-admin Admin

Add a new group

Create

Name* UIM

Description

Users

Users jozko-uim-admin (Jožko Mrkvička) x

Permissions

Permissions CRUD UIM Devices x CRUD UIM IPs x View UIM Tenant x

Cancel Create Create & Add Another

2025-05-16 12:18:48 UTC · csirt-asset-management · NetBox Community v4.1.7

Obr. 16: Vytvorenie novej skupiny

7.2.2 Priradenie oprávnení skupine

Po vytvorení skupiny sa automaticky presuniete na stránku s detailom skupiny. V sekcii *Permission* kliknite na možnosť *Add*.

netbox Community

Search...

netbox-admin Admin

Permissions

Results 35 Filters

Quick search

Configure Table

NAME	ENABLED	OBJECT TYPES	CAN VIEW	CAN ADD	CAN CHANGE	CAN DELETE	DESCRIPTION
ALL CRUD Devices Details	✓	Manufacturer Device Type Device Role	✓	✓	✓	✓	Permission to CRUD device details. Device Role, Device Type, Manufacturers
ALL CRUD Interface	✓	Interface	✓	✓	✓	✓	Permission to CRUD Interfaces. All admins should have this permission.
ALL CRUD Unspecified Tenant Devices	✓	Device	✓	✓	✓	✓	Permission to CRUD Device with unspecified tenant.
ALL CRUD Unspecified Tenant IPs	✓	IP Address	✓	✓	✓	✓	Permission to CRUD IP Address without specified tenant.
ALL View Organization Details	✓	Region Site Group Site Location	✓	✗	✗	✗	Permission to use organisation detail in add new entity. Sites, Site Group, Region, Location. All admin should have this permission.
CRUD IKAL Devices	✓	Device	✓	✓	✓	✓	—
CRUD IKAL IPs	✓	IP Address	✓	✓	✓	✓	—
CRUD TIS Devices	✓	Device	✓	✓	✓	✓	—
CRUD TIS IPs	✓	IP Address	✓	✓	✓	✓	—

+ Add Export

Obr. 17: Tabuľka oprávnení

Zobrazí sa formulár, kde treba vybrať:

- **Action:** typ oprávnenia - view (zobrazenie), add (pridanie), change (úprava), delete (odstránenie).
- **Object type:** typ objektu, ku ktorému sa oprávnenie vzťahuje (pre zvolenie viacerých objektov je potrebné podržať klávesu **Ctrl**) (napr. `dcim | device`, `ipam | ip address`, `tenancy | tenant`).
- **Groups:** skupiny, pre ktoré sa oprávnenie bude vzťahovať.
- **Users:** používatelia, jednotlivci pre ktorých toto oprávnenie bude platiť bez ohľadu na to, ktorej skupine konkrétny používateľ patrí.
- **Constraint** (voliteľné) špecifikácia podmienok, pre ktoré sa oprávnenie uplatní (napr. iba určité tenanti alebo iba objekty s konkrétnym štítkom), špecifikuje sa v JSON formate.

Po vyplnení kliknite na *Create*.

Name* CRUD UIM Devices

Description

☒ Enabled

Actions

- ☒ Can view
- ☒ Can add
- ☒ Can change
- ☒ Can delete

Additional actions

Actions granted in addition to those listed above

Objects

Object types*

- DCIM > Front Port Template
- DCIM > Interface
- DCIM > Interface Template
- DCIM > Inventory Item
- DCIM > Inventory Item Role
- DCIM > Inventory Item Template
- DCIM > Location

Assignment

Groups UIM Administrators x

Users

Constraints

Constraints {"tenant_id": "1"}

Cancel **Create** Create & Add Another

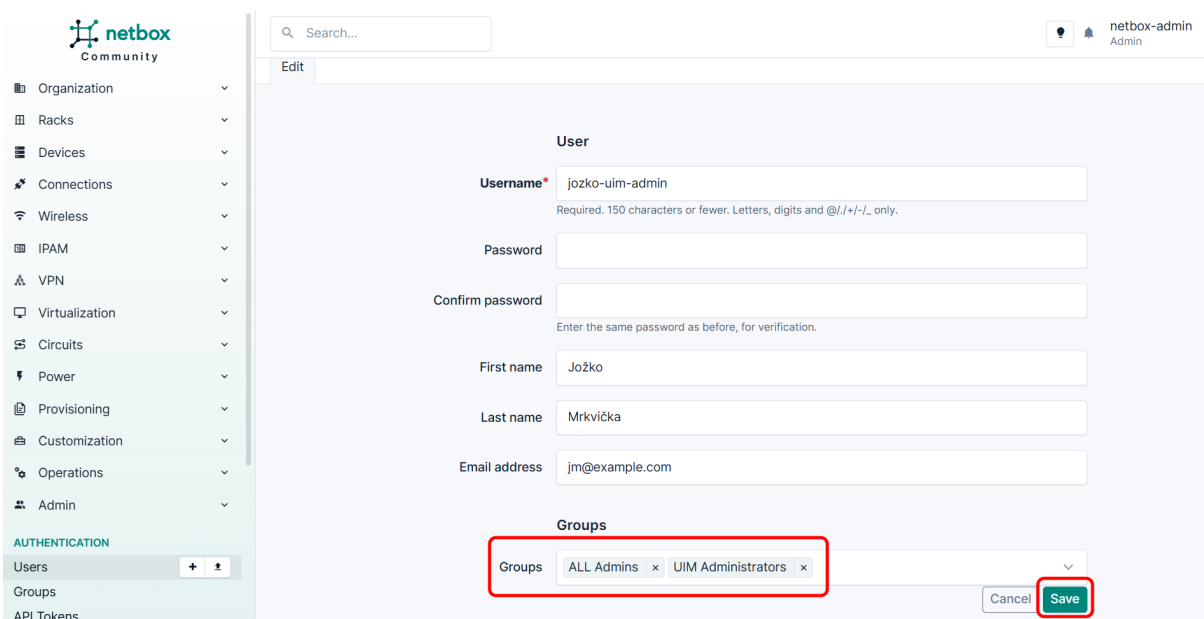
Obr. 18: Vytvorenie oprávnenia pre UIM administrátorov podľa návrhu

7.2.3 Priradenie skupiny používateľovi

Keď máte vytvorenú skupinu a definované jej oprávnenia, priradíte ju konkrétnemu používateľovi.

V ľavom menu prejdite späť do sekcie *Users*, nájdite požadovaného používateľa a kliknite na jeho meno. Na stránke detailu používateľa prejdite do časti *Groups* a zaškrtnite skupinu, ktorú ste vytvorili. Môžete priradiť viacero skupín naraz. Následne kliknite na *Save*.

Po tomto kroku bude mať daný používateľ všetky oprávnenia, ktoré sú definované v priradených skupinách.



Obr. 19: Priradenie skupín používateľovi

7.2.4 Poznámky a ďalšie možnosti

Oprávnenia je možné priradovať aj priamo používateľovi bez použitia skupiny, ale z dlhodobého hľadiska je správa cez skupiny oveľa prehľadnejšia a individuálne priradovanie pravidiel jednotlivcom by malo byť použité skôr vo výnimočných prípadoch. V prípade rozsiahlejších tímov je vhodné vytvoriť viacero skupín podľa pracovných rolí a zodpovedností.

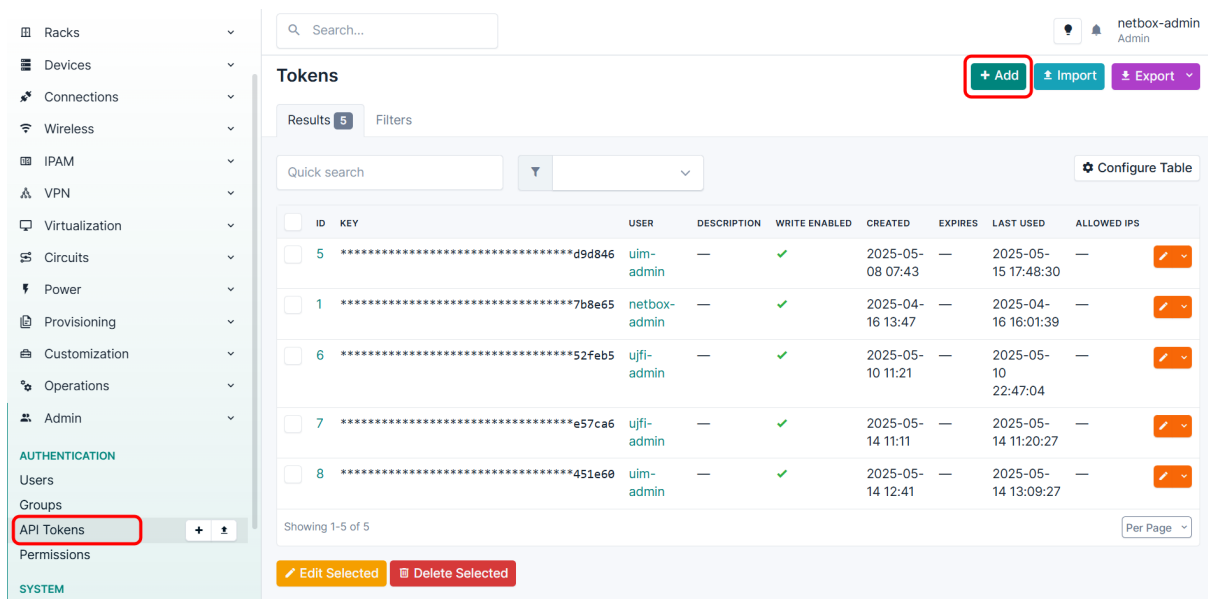
NetBox tiež umožňuje pokročilé obmedzenia pomocou tzv. object-level constraints, napr. povoliť používateľovi meniť iba záznamy s určitým *tenantom* alebo v rámci určitej lokality.

7.3 Vytvorenie a správa API tokenov

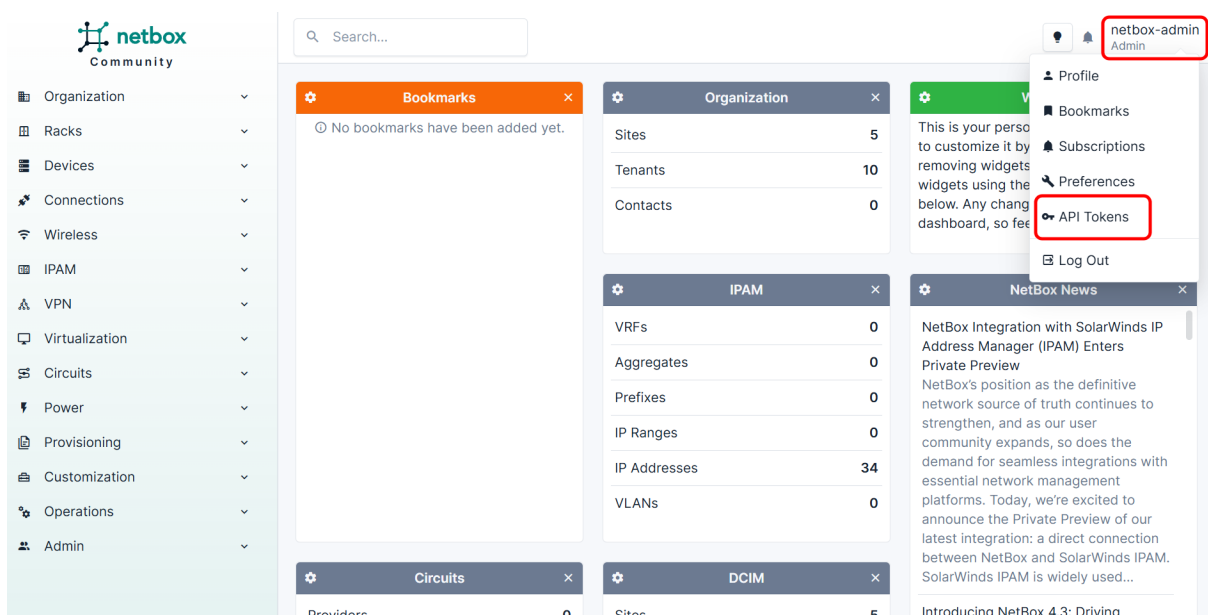
NetBox poskytuje REST API rozhranie, ktoré umožňuje programátorské alebo automatizované prístupy k údajom uloženým v systéme. Na overenie týchto požiadaviek sa využívajú API tokeny, ktoré sú generované na úrovni používateľa a fungujú ako alternatíva k bežnému prihlasovaniu heslom. Každý používateľ si môže vytvoriť vlastné API tokeny (ak má aktívny účet), no ich správa, vrátane obmedzenia ich platnosti alebo rozsahu práv je dostupná len superadministrátorom alebo cez vlastný účet daného používateľa. Táto časť návodu popisuje proces vytvorenia a používania tokenov z pohľadu superadmina aj používateľa.

7.3.1 Vytvorenie nového tokenu

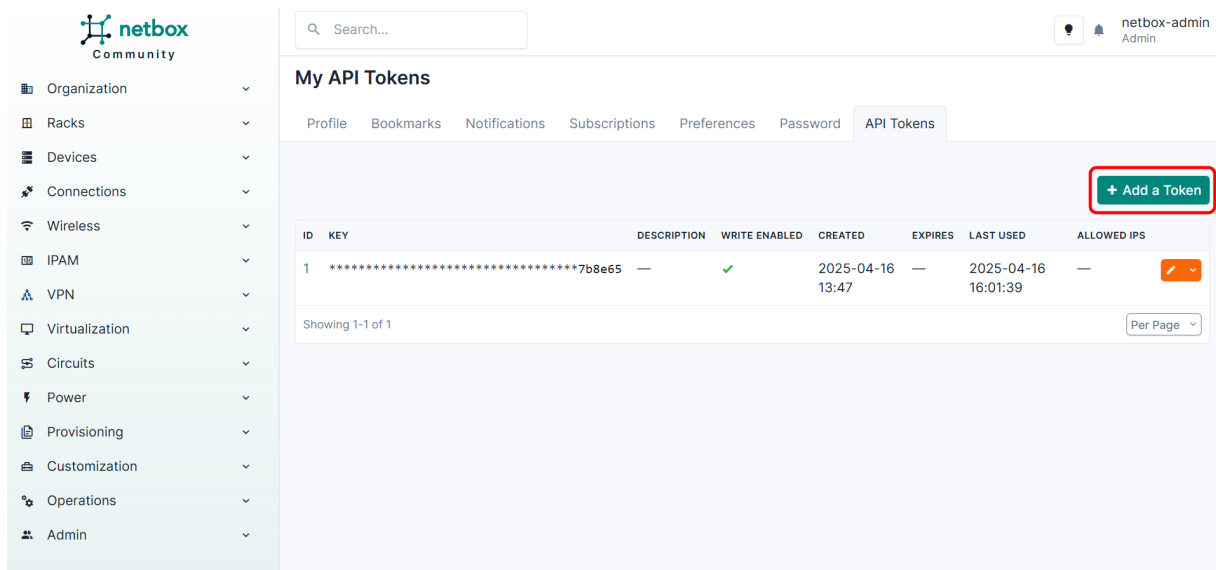
Po prihlásení do NetBoxu kliknite v pravom hornom rohu na svoje používateľské meno a z ponuky vyberte možnosť *My Profile* (Môj profil). Na stránke profilu sa v hornej časti nachádza niekoľko záložiek. Kliknite na *API Tokens*, kde uvidíte zoznam aktuálne vytvorených tokenov pre váš účet.



Obr. 20: Tabuľka tokenov



Obr. 21: Navigácia k správy vlastných tokenov



Obr. 22: Tabuľka vlastných tokenov

Ak ste superadmin a chcete vytvoriť token pre iného používateľa, najprv prejdite do *Admin* → *API Tokens*, zobrazí sa zoznam všetkých priradených tokenov a kliknete na tlačidlo *Add*.

Zobrazí sa formulár, kde treba vybrať:

- **User:** používateľ, ktorému chceme prideliť token.
- **Write enabled:** ak je toto políčko zaškrtnuté, token bude mať aj práva na zápis (tzn. umožní meniť, pridávať a mazať dáta cez API).
- **Expires:** dátum a čas, kedy token stratí platnosť. Ak toto pole necháte prázdne, token bude platný neobmedzene, čo však z bezpečnostných dôvodov neodporúčame.
- **Key:** kľúč, ktorý je potrebné si niekde uložiť, pretože po vytvorení už ho nie je možné zobraziť.

Po vyplnení požadovaných polí kliknite na *Create*.

User* jozko-uim-admin

Key* 85a461ba508c26ce46ead82ee33e47c504189249

Keys must be at least 40 characters in length. **Be sure to record your key** prior to submitting this form, as it may no longer be accessible once the token has been created.

☒ **Write enabled**
Permit create/update/delete operations using this key

Expires 2025-05-30 12:00:00

Description

Allowed IPs

Allowed IPv4/IPv6 networks from where the token can be used. Leave blank for no restrictions. Example:
10.1.1.0/24,192.168.10.16/32,2001:db8:1::/64

Cancel **Create** Create & Add Another

Obr. 23: Priradenie nového tokenu používateľovi

7.3.2 Použitie API tokenu

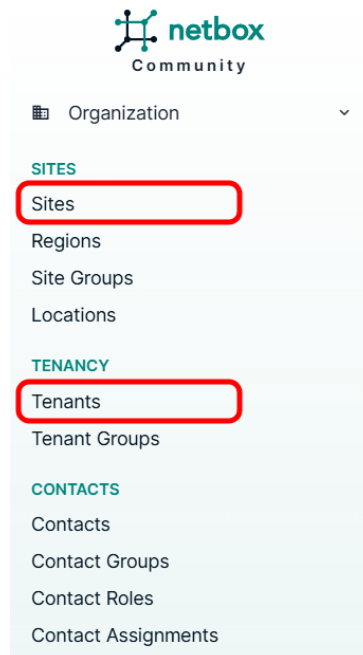
Po vytvorení tokenu sa na obrazovke zobrazí jeho hodnota – dlhý alfanumerický reťazec, ktorý je potrebné skopírovať a bezpečne uložiť. Token sa po opustení tejto stránky už znova nezobrazí.

Na použitie tokenu v API požiadavkách je potrebné poslať ho v HTTP hlavičke **Authorization** vo formáte:

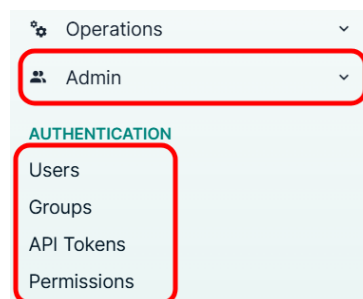
Authorization: Token <hodnota_tokenu>

Týmto spôsobom sa môžete autentifikovať voči REST API bez potreby prihlasovania pomocou používateľského mena a hesla

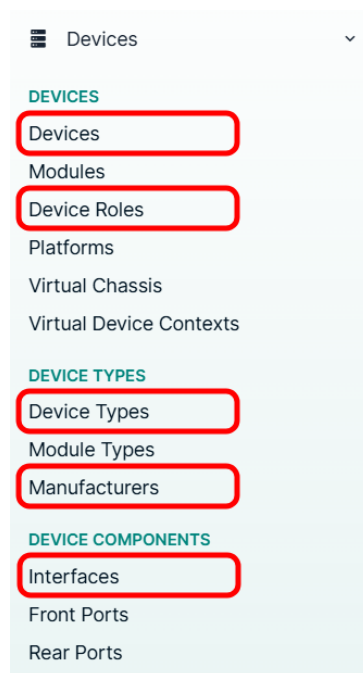
7.4 Ďalšie dôležité položky



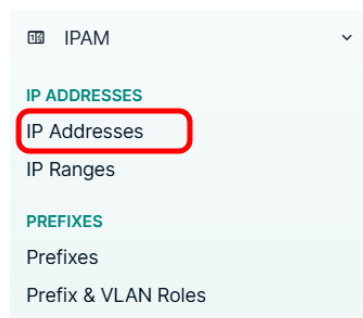
Obr. 24: Navigácia k zoznamu organizačných informácií a zoznamu ústavov (Tenants)



Obr. 25: Navigácia na adminovú stránku



Obr. 26: Navigácia k zoznamu zariadení



Obr. 27: Navigácia k zoznamu IP adries

8 Návod na obsluhu programu

Po úspešnej inštalácii a konfigurácii je možné spustiť samotný nástroj na skenovanie siete.

8.1 Základné spustenie skriptu

Skript je možné spustiť v troch rôznych formátoch. Po úspešnom naskenovaní siete sa vypíše zoznam nájdených zariadení.

```
python run.py
```

alebo s parametrom skenovanej podsiete (rozsah ip adries pomocou ipv4 CIDR formátu)
napríklad:

```
python run.py -addr 147.175.106.0/24
```

prípadne aj s názvom súboru, ktorý bude obsahovať export nájdených/importovaných zariadení:

```
python run.py -addr 147.175.106.0/24 -o results.csv
```

```
csirt@csirt-asset-management:~/Desktop/UniNetInsight$ python run.py -addr 147.175.106.0/24
Scanning subnet 147.175.106.0/24 ...
[
  {
    "id": null,
    "interface_id": null,
    "ip_addr_id": null,
    "manufacturer_id": null,
    "device_type_id": null,
    "ip_addr": "147.175.106.1",
    "mac_addr": "40:14:82:D6:AD:3D",
    "manufacturer": "Cisco Systems, Inc",
    "status": "active",
    "hostname": "_gateway"
  },
  {
    "id": null,
    "interface_id": null,
    "ip_addr_id": null,
    "manufacturer_id": null,
    "device_type_id": null,
    "ip_addr": "147.175.106.2",
    "mac_addr": "18:FD:74:FE:E5:40",
    "manufacturer": "Unspecified",
    "status": "active",
    "hostname": "uim.elf.stuba.sk"
  },
  {
    "id": null,
    "interface_id": null,
    "ip_addr_id": null,
    "manufacturer_id": null,
    "device_type_id": null,
    "ip_addr": "147.175.106.8",
    "mac_addr": "AC:1F:6B:0B:1C:66",
    "manufacturer": "Super Micro Computer",
    "status": "active",
    "hostname": null
  },
]
```

Obr. 28: Zoznam nájdených zariadení

8.2 Možnosti importovania a výber ústavu a bloku

Po dokončení skenovania program vypíše všetky naskenované zariadenia a umožní používateľovi vybrať si, či budú do **NetBoxu** importované všetky zariadenia, alebo bude selektívne vyberať medzi jednotlivými zariadeniami. Následne používateľ vyberá ústav, pod ktorý budú zariadenia patriť (používateľ patrí len pod určité ústav, spravidla jeden, preto má na výber len jemu dostupné ústavy, prípadne „no tenant“ znamená, že zariadenie importované do **NetBoxu** je viditeľné pre všetkých používateľov a nepatrí pod konkrétny ústav).

```
Devices scanned: 26

Choose how to upload scanned devices to NetBox:
0: Do not upload
1: Upload all devices
2: Select devices individually
Enter choice [0/1/2]: 1

Available tenants:
0: No tenant
1: UIM - Ústav informatiky a matematiky
Enter a number corresponding to your name: 1

Available Sites:
0: Block A
1: Block B
2: Block C
3: Block D
4: Block E
Enter a number corresponding to your name: 0
```

Obr. 29: Možnosti importu a voľba ústavu/bloku

8.3 Výber role

Používateľovi je umožnené zvoliť rolu ku každému zariadeniu. Môže tak urobiť hromadne a nastaviť rovnakú rolu pre všetky zariadenia, alebo individuálne a pri každom zariadení sa program opýta, akú rolu má nastaviť konkrétnemu zariadeniu.

```

How do you want to select Device Roles?
0: Same Device Role for all devices
1: Different Device Role per device
Enter choice [0/1]: 1

Assigning role/type for device: 147.175.106.1 | MAC: 40:14:82:D6:AD:3D | Manufacturer: Cisco Systems, Inc
Available Device Roles:
0: Access Point
1: Cable Related
2: Desktop PC
3: Firewall
4: Network storage
5: Router
6: Single Board Computers
7: Switch
8: Unspecified
9: Virtual Machine
10: Web Server
Enter a number corresponding to your name: 5

Device '_gateway' added (MAC: 40:14:82:D6:AD:3D) (Manufacturer: Cisco Systems, Inc).

```

Obr. 30: Nastavenie role zariadeniam

8.4 Informácie o úspešnom nahratí a voľba exportovania

Používateľ je informovaný o počte úspešne importovaných zariadení do **NetBoxu** a na záver mu príde výzva na exportovanie dát. Používateľ má na výber, či chce exportovať všetky naskenované zariadenia, alebo len zariadenia, ktoré boli vybrané na importovanie do **NetBoxu**, prípadne neexportovať nič.

```

Imported devices: 26

Do you want to export scan results to CSV?
0: Do not export
1: Export all scanned devices
2: Export only uploaded devices
Enter choice [0/1/2]: 1
Include NetBox-assigned IDs? [y/N]: n
Enter output filename (leave blank for default): final-export
Hosts exported to CSV: /home/csirt/Desktop/UniNetInsight/final-export.csv
csirt@csirt-asset-management:~/Desktop/UniNetInsight$ cat final-export.csv
ip_addr,mac_addr,manufacturer,status,hostname
147.175.106.1,40:14:82:D6:AD:3D,"Cisco Systems, Inc",active,_gateway
147.175.106.2,18:FD:74:FE:E5:40,Unspecified,active,uim.elf.stuba.sk
147.175.106.8,AC:1F:6B:0B:1C:66,Super Micro Computer,active,
147.175.106.10,F4:6D:04:2E:6E:F1,Asustek Computer,active,
147.175.106.22,BC:EE:7B:9E:D0:0A,Asustek Computer,active,
147.175.106.23,90:1B:0E:0A:08:E5,Fujitsu Technology Solutions GmbH,active,
147.175.106.125,90:09:D0:04:6B:4F,Unspecified,active,
147.175.106.126,00:11:32:F5:C9:CD,Synology Incorporated,active,
147.175.106.128,B6:B1:71:75:28:16,Unspecified,active,
147.175.106.145,08:BF:B8:74:C6:E2,Unspecified,active,
147.175.106.165,B8:27:EB:2E:94:2F,Raspberry Pi Foundation,active,
147.175.106.166,08:00:27:A6:E7:D2,Oracle VirtualBox virtual NIC,active,
147.175.106.171,58:11:22:B6:C4:D0,Unspecified,active,
147.175.106.172,04:7C:16:C1:9A:3D,Unspecified,active,
147.175.106.181,50:EB:F6:77:64:56,ASUSTek COMPUTER INC.,active,
147.175.106.187,04:7C:16:C1:9A:13,Unspecified,active,
147.175.106.188,1C:1B:0D:07:69:3C,Giga-byte Technology,active,
147.175.106.190,08:BF:B8:03:4E:94,Unspecified,active,
147.175.106.193,08:BF:B8:36:A6:AE,Unspecified,active,
147.175.106.194,08:BF:B8:36:A6:69,Unspecified,active,
147.175.106.195,08:BF:B8:36:A6:D9,ASUSTek COMPUTER INC.,active,
147.175.106.196,08:BF:B8:36:A2:C1,Unspecified,active,
147.175.106.197,08:BF:B8:36:A2:85,Unspecified,active,
147.175.106.230,14:DD:A9:D6:63:2C,Asustek Computer,active,
147.175.106.248,F0:79:59:DA:D6:7C,Asustek Computer,active,
147.175.106.21,,Unspecified,active,csirt-asset-management

```

Obr. 31: Výpis o úspešnom importovaní a možnosť exportovania

9 Prihlasovacie údaje a nosič

V tejto kapitole sú stručne zhrnuté potrebné prihlasovacie údaje a odkazy na relevantné (digitálne) nosiče a materiály.

9.1 NetBox server

- IP adresa: `https://147.175.106.21`
- Používateľ (superadmin):
 - Meno: `netbox-admin`
 - Heslo: `netBox963.`
 - API kľúč: `6ca44f5ebb5b1ef81ef7398106c405a5e89f7327`

9.2 Softvérové zdroje a dokumentácia

- GitHub repozitár: `github.com/Faraon40/UniNetInsight`
- Webová stránka projektu: `marcelstak111443.github.io`

10 Zápisnice

Stránka projektu: <https://marcelstotak111443.github.io/notes.html>

Zápis z 1. stretnutia TP

Dátum:	30. 9. 2024
Miesto stretnutia:	Blok C, 7. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková
Téma stretnutia:	Úvodné stretnutie
Opis stretnutia:	Na úvodnej konzultácii sme si predstavili hlavnú myšlienku projektu – správu zariadení v sieti. Dohodli sme si pravidelný termín konzultácií, zatiaľ na týždennej báze v pondelok o 11:00, a od zadávateľa projektu sme dostali základné úlohy do ďalšieho stretnutia.

Úlohy na ďalšie stretnutie:

1. Nastavenie komunikačného kanálu
2. Vytvorenie GitHub repozitáru

Zápis z 2. stretnutia TP

Dátum:	7. 10. 2024
Miesto stretnutia:	Blok C, 7. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková
Téma stretnutia:	Predstavenie komunikačného kanálu, GitHub repozitára, rozdelenie úloh v rámci tímu
Opis stretnutia:	<p>Zadávatelovi projektu sme predstavili nastavený komunikačný kanál – Discord a zriadený GitHub repozitár. Do budúcej konzultácie sme si stanovili úlohu naštudovať existujúce riešenia a rozhodnúť sa, či projekt implementujeme od začiatku („from scratch“) alebo budeme rozširovať existujúce riešenia. Zvolili sme si nasledovné roly:</p> <ul style="list-style-type: none">• Vedúci tímu – Samuel Michalčík• Zápisnice – Marcel Soták/Marek Štrba/Antonio Kiš• Webová stránka – Marcel Soták• Analýza existujúcich riešení a úvod do problematiky – Antonio Kiš a Marek Štrba

Vyhodnotenie úloh z predošlého stretnutia:

Komunikačný kanál bol vytvorený a pozvánky boli rozoslané príslušným osobám. GitHub repozitár bol taktiež úspešne vytvorený.

Úlohy na ďalšie stretnutie:

1. Naštudovať si existujúce riešenia

Zápis z 3. stretnutia TP

Dátum: 14. 10. 2024

Miesto stretnutia: Blok C, 7. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková

Téma stretnutia: Predstavenie Netboxu

Opis stretnutia: Zadávateľovi projektu sme predstavili existujúce riešenie – Netbox, ktoré sme si v tíme zvolili. Prešli sme si jeho funkcionality a požiadavky zadávateľa. Identifikovali sme, že v open-source riešení Netbox chýba (polo)automatizácia evidencie zariadení pripojených do siete, čo je jedným z hlavných cieľov nášho projektu. Zvážili sme možnosť implementácie tejto funkcionality alebo využitia existujúcich rozšírení a pluginov. Projekt by mal podporovať zber a kategorizáciu zariadení pre celú fakultu podľa jednotlivých ústavov a mal by obsahovať vhodné administrátorské rozhranie. Do ďalšieho stretnutia sme si stanovili úlohy – detailne preskúmať Netbox a jeho rozšírenia a vytvoriť webovú stránku na evidenciu zápisníc a členov tímu.

Vyhodnotenie úloh z predošlého stretnutia:

Preštudovali sme existujúce riešenia a zvolili vhodný framework pre projekt – Netbox.

Úlohy na ďalšie stretnutie:

1. Preskúmať pluginy Netboxu
2. Vytvorenie web stránky na evidenciu zápisníc
3. Vytvorenie virtuálneho stroja pre nasadenie

Zápis zo 4. stretnutia TP

Dátum:	21. 10. 2024
Miesto stretnutia:	Blok C, 7. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková
Téma stretnutia:	Predstavenie pluginov pre Netbox, prezentácia nasadeného Netboxu
Opis stretnutia:	Zadávatelov projektu sme oboznámili s dostupnými pluginmi pre Netbox. Automatizáciu (aspoň čiastočnú poloautomatizáciu) by mohla zabezpečiť kombinácia Netbox Django API a pluginu Nmap, pričom sa očakáva implementácia z našej strany. Za ďalší vhodný plugin považujeme generovanie QR kódov pre zariadenia. Ukázali sme funkčný Netbox nasadený na virtuálnom stroji, ktorý však nedetekoval zariadenia, pretože virtuálny stroj nedokáže čítať reálnu sieť. Pre testovanie sme identifikovali potrebu vytvoriť simulované prostredie. Zadávateľ nám poskytol kontakt na študentku, ktorá nám následne zaslala potrebné materiály. Zároveň sme získali kľúče od miestnosti, ktorú môžeme využiť na spoločnú prácu a konzultácie.

Vyhodnotenie úloh z predošlého stretnutia:

Preskúmali sme využiteľné pluginy pre Netbox a vytvorili sme virtuálny stroj s nasadeným Netboxom.

Úlohy na ďalšie stretnutie:

1. Konzultácia so študentkou z tímu, ktorý sa v minulosti snažil implementovať riešenie

Zápis z 5. stretnutia TP

Dátum: 28. 10. 2024

Miesto stretnutia: Blok C, 7. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková

Téma stretnutia: Upresnenie cieľov projektu

Opis stretnutia: Na konzultácii sme bližšie rozobrali konkrétne ciele projektu a špecifikovali hlavnú myšlienku – evidenciu IT komponentov v sieti a zautomatizovanie ich detekcie. Podstatným cieľom je umožniť správcovi siete vykonať sken, ktorý automaticky nájde a uloží všetky zariadenia v danej sieti. Tento sken by mal identifikovať IP adresu, MAC adresu, názov zariadenia a výrobcu. Riešili sme tiež rozdelenie používateľských práv – superadmin a administrátori jednotlivých ústavov. Do ďalšieho stretnutia sme si stanovili čas na testovanie a implementáciu základnej funkcionality.

Vyhodnotenie úloh z predošlého stretnutia:

Po konzultáciách so študentkou sme sa rozhodli implementovať vlastné riešenie.

Úlohy na ďalšie stretnutie:

1. Implementácia skenu siete s porovnaním výsledkov jednotlivých pluginov pre skenovanie

Zápis zo 6. stretnutia TP

Dátum: 11. 11. 2024

Miesto stretnutia: Blok C, 7. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková

Téma stretnutia: Predstavenie implementácie skenovania siete

Opis stretnutia: Na konzultácii sme predstavili preskúmanú databázu Netboxu a prácu s linuxovými nástrojmi Nmap a Angry IP Scanner. Upresnili sme si celý proces:

1. Spustenie skriptu
2. Sken siete
3. Uloženie údajov do formátov (CSV/XML/JSON), ktoré dokáže Netbox API prijať
4. Spracovanie údajov a import dát do Netboxu

Vyhodnotenie úloh z predošlého stretnutia:

Predstavili sme skenovanie siete a praktické porovnanie pluginov pre skenovanie.

Úlohy na ďalšie stretnutie:

1. Spracovanie oskenovaných údajov a ich importovanie do Netboxu
2. Namapovanie IP adries do tabuľky v Netboxe

Zápis zo 7. stretnutia TP

Dátum: 18. 11. 2024

Miesto stretnutia: Blok C, 7. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková

Téma stretnutia: Predstavenie importovania dát a ich mapovanie v Netboxe

Opis stretnutia: Na konzultácii sme predstavili spôsob, akým sa ukladajú dáta do Netboxu. Evidencia zariadení (devices) a IP adresy (ipaddresses) si vyžadovala mapovanie prostredníctvom tabuľky `interfaces`. V lokálnej sieti fakultného CSIRTu sme spustili nástroje Nmap a Angry IP Scanner a predviedli sme synchronizáciu a importovanie údajov cez REST API do Netboxu s nastavenými používateľskými právami.

Vyhodnotenie úloh z predošlého stretnutia:

Predstavili sme implementované importovanie dát do Netboxu a ich následné namapovanie.

Úlohy na ďalšie stretnutie:

1. Spojiť doterajšie výstupy do funkčného celku – skenovanie siete a importovanie dát

Zápis z 8. stretnutia TP

Dátum: 2. 12. 2024

Miesto stretnutia: Blok C, 7. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík, Natália Komprivňáková

Téma stretnutia: Predstavenie prvotnej implementácie praktickej časti tímového projektu

Opis stretnutia: Na konzultácii sme odprezentovali funkčné riešenie projektu. Pripravili sme prezentáciu vysvetľujúcu štruktúru databázy Netboxu, použité nástroje a proces skenovania siete. Demonštrovali sme použitie príkazu Nmap a spôsob importu údajov do Netboxu. Dohodli sme drobné úpravy a stanovili termín dokončenia prvej dokumentácie na 20. 1. 2025. Do tohto dátumu budeme pracovať na oprave chýb a príprave technickej dokumentácie.

Vyhodnotenie úloh z predošlého stretnutia:

Predstavili sme implementáciu riešenia pre skenovanie zariadení v sieti a ich následné namapovanie v Netboxe.

Úlohy na ďalšie stretnutie:

1. Dokončiť technickú dokumentáciu
2. Opraviť chyby aktuálnej implementácie

Zápis z 9. stretnutia TP

Dátum:	5. 3. 2025
Miesto stretnutia:	Blok D, 4. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík
Téma stretnutia:	Dohodnutie sa na ďalšom postupe, rozdelenie v Netboxe pre jednotlivé ústavy a ich administrátorov
Opis stretnutia:	Diskutovali sme o overení, či administrátor ústavu vie samostatne oskenovať a spravovať svoj ústav. Cieľom je umožniť koexistenciu viacerých ústavov bez možnosti vzájomného prepisovania zariadení. Vytvorili sme dve fiktívne inštitúcie, na ktorých sme tieto situácie testovali. Zároveň sme sa dohodli na vytvorení používateľského manuálu pre administrátorov s cieľom čo najrýchlejšieho zaškolenia. Administrátori by si mali vedieť stiahnuť predpripravené dokumenty a nahrať ich do Netboxu. Taktiež sme sa dohodli na vytvorení prezentácie doterajších výstupov. Na ďalšej konzultácii (19. 3. 2025) sa môže zúčastniť osoba so skúsenosťami s Netboxom, ktorá by nám mohla poskytnúť rady.

Vyhodnotenie úloh z predošlého stretnutia:

Technická dokumentácia bola vytvorená, drobné chyby boli opravené.

Úlohy na ďalšie stretnutie:

1. Pripraviť používateľský manuál pre administrátorov
2. Vytvoriť prezentáciu pre administrátora siete

Zápis z 10. stretnutia TP

Dátum:	19. 3. 2025
Miesto stretnutia:	Blok D, 4. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík
Téma stretnutia:	Prezentovanie nášho riešenia administrátorovi siete
Opis stretnutia:	Odprezentovali sme naše doterajšie riešenia administrátorovi siete. Spoločne sme prediskutovali návrhy a možný ďalší postup. Dohodli sme sa na rozšírení riešenia o možnosť filtrovania podľa miestností a zároveň podľa tenantov. Ďalšou úlohou je doplniť dokumentáciu, ktorá bude zahŕňať tieto nové funkcionality a zistenia.

Vyhodnotenie úloh z predošlého stretnutia:

Používateľský manuál zatiaľ nebol dokončený, keďže bolo potrebné vykonať ďalšie konzultácie s administrátorom siete a vedúcimi projektu. Prezentácia bola pripravená a úspešne odprezentovaná.

Úlohy na ďalšie stretnutie:

1. Rozšírenie o filtrovanie podľa miestností a zároveň tenantov
2. Doplnenie dokumentácie

Zápis z 11. stretnutia TP

Dátum: 2. 4. 2025

Miesto stretnutia: Blok D, 4. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík

Téma stretnutia: Upresnenie, kde budeme testovať našu implementáciu

Opis stretnutia: Na stretnutí sme si prešli všetky doterajšie kroky a dohodli sa na ďalšom testovaní nášho riešenia v prostredí konkrétnej školskej siete. Keďže miestnosť, v ktorej sme pôvodne pracovali, sa začala rekonštruovať, bolo potrebné zvážiť nové možnosti testovania. Dohodli sme sa, že postačuje testovanie v školskej sieti s obmedzeným počtom zariadení, čo predstavuje zmenu oproti predchádzajúcemu pracovnému prostrediu siete CSIRT. Ako alternatívu sme zvažovali aj sieť Ústavu fyziky. Určili sme termín, miesto a čas ďalšieho stretnutia s plánovaným testovaním.

Vyhodnotenie úloh z predošlého stretnutia:

Rozšírenie riešenia bolo úspešné a dokumentácia bola doplnená.

Úlohy na ďalšie stretnutie:

1. Sfunkčnenie Netboxu v novej sieti na novej lokalite

Zápis z 12. stretnutia TP

Dátum: 16. 4. 2025

Miesto stretnutia: Blok A, 2. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík

Téma stretnutia: Sfunkčnenie Netboxu na novej sieti

Opis stretnutia: Stretli sme sa so zodpovednou osobou a vykonali skúšobný sken, ktorý prebehol úspešne. Počítač, na ktorom testujeme, sa nachádza v miestnosti s obmedzeným fyzickým prístupom (na rozdiel od predchádzajúcej lokality CSIRT, kde sme mali voľný prístup), preto sme sa dohodli na vzdialenom prístupe cez SSH. Ten nám umožní realizovať ďalšie testovania bez potreby fyzickej prítomnosti.

Vyhodnotenie úloh z predošlého stretnutia:

Sfunkčnenie Netboxu prebehlo úspešne.

Úlohy na ďalšie stretnutie:

1. Testovanie Netboxu v novej sieti

Zápis z 13. stretnutia TP

Dátum: 23.04.2025 —

Miesto stretnutia: Blok A, 2. poschodie

Vedúci projektu: Ing. Štefan Balogh, PhD.

Prítomní: Bc. Antonio Kiš

Téma stretnutia: Nastavenie NetBoxu, import dát zo skenu siete, príprava entít

Opis stretnutia: V rámci stretnutia sme nastavili NetBox a vytvorili niekoľko základných entít priamo v jeho prostredí. Klonovali sme projektový repozitár a pokúsili sa spustiť sprievodný program. Pomocou nástroja Nmap sme oskenovali vnútornú sieť a výstupy zo skenu sme sa pokúsili importovať do NetBoxu. Import však vyžadoval dodatočné spracovanie výstupov, ktoré sme si naplánovali riešiť v nasledujúcich krokoch.

Úlohy na ďalšie stretnutie:

1. Naplniť NetBox potrebnými entitami (Tenants, Sites, Používateľské účty, Oprávnenia)
2. Dokončiť spracovanie dát zo skenu siete a zabezpečiť ich úspešný import do NetBoxu

Zápis z 14. stretnutia TP

Dátum:	7. 5. 2025
Miesto stretnutia:	Blok D, 4. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík
Téma stretnutia:	Pripojenie do novej siete, prezentácia výsledku
Opis stretnutia:	Na konzultácii sme prešli doteraz doimplementované riešenia, ktoré sme spoločne analyzovali a identifikovali chyby. Zamerali sme sa na návrhy vylepšení a rozšírení, ktoré by mohli zvýšiť praktickú využiteľnosť riešenia – najmä v oblasti automatizácie, detekcie výpadkov zariadení, porovnávania údajov zo skenovania so stavom v Netboxe a zobrazovania výstupov v čitateľnej forme. Dohodli sme sa na kontakte so sieťovým administrátorom z iného ústavu v rámci fakulty s cieľom otestovať systém v reálnych podmienkach a získať spätnú väzbu od prvého reálneho používateľa.

Vyhodnotenie úloh z predošlého stretnutia:

Príhlásenie a skenovanie cez vzdialený prístup prebehlo úspešne.

Úlohy na ďalšie stretnutie:

1. Vytvoriť používateľský manuál – krok za krokom
2. V manuáli popísať Nmap, jeho použitie a výstupy
3. Priložiť odkazy na inštalácie a dokumentáciu k nástrojom
4. Ošetriť vstupy skriptu – povoliť len jednoduchý názov súboru, automatická prípona .txt alebo .csv
5. Publikovať manuál a dokumentáciu na webstránku
6. Otestovať systém na reálnej sieti na fakulte
7. Zapojiť testera (napr. administrátora) – dať mu konto a asistovať pri testovaní

Zápis zo 15. stretnutia TP

Dátum:	14. 5. 2025
Miesto stretnutia:	Blok D, 4. poschodie
Vedúci projektu:	Ing. Štefan Balogh, PhD.
Prítomní:	Bc. Marek Štrba, Bc. Marcel Soták, Bc. Antonio Kiš, Bc. Samuel Michalčík
Téma stretnutia:	Pripojenie do novej siete, prezentácia výsledku
Opis stretnutia:	Po oskenovaní novej siete sme identifikovali a analyzovali pripojené zariadenia. Následne sme prezentovali konečné výsledky skenovania, zhrnuli sme všetky postupy a nastavenia v NetBoxe. V rámci záverečnej prezentácie sme odpovedali na otázky vedúceho týkajúce sa celého riešenia.

Úloha na ďalšie stretnutie:

1. Dokončiť dokumentáciu
2. Opraviť drobné nedostatky, ktoré sme objavili pri nasadení a testovaní riešenia na UIFJ

Zoznam použitej literatúry

1. KINZA YASAR, Alexander S. Gillis. What is a computer network? 2023. Dostupné tiež z: <https://www.techtarget.com/searchnetworking/definition/network>. Použité dňa 17.1.2025.
2. ———. What is Computer Networking? 2024. Dostupné tiež z: <https://www.geeksforgeeks.org/what-is-computer-networking/>. Použité dňa 17.1.2025.
3. MD SHOHEL AMIN, Showrov Rahman. An Introduction of Open System Interconnection (OSI) Model and its Architecture. 2023. Dostupné tiež z: https://www.researchgate.net/publication/371120497_An_Introduction_of_Open_System_Interconnection_OSI_Model_and_its_Architecture. Použité dňa 17.1.2025.
4. KOH, Fred. Logical vs. Physical Network Diagrams. 2020. Dostupné tiež z: <https://graphicalnetworks.com/blog-logical-vs-physical-network-diagrams/>. Použité dňa 17.1.2025.
5. ———. What Is the Internet of Things? | Oracle. 2020. Dostupné tiež z: <https://www.oracle.com/internet-of-things/>. Použité dňa 17.1.2025.
6. MITCHELL, Cory. IP Address: Definition, How It Works, and Examples. 2024. Dostupné tiež z: <https://www.investopedia.com/terms/i/ip-address.asp>. Použité dňa 17.1.2025.
7. TERRA, John. What is a MAC Address, and How Do I Find It? 2024. Dostupné tiež z: <https://www.simplilearn.com/what-is-mac-address-how-to-find-it-article>. Použité dňa 17.1.2025.
8. RUSLANOVA, Diana. Definition of Address resolution protocol. 2024.
9. GORBACHENKO, Pavel. What are Functional and Non-Functional Requirements and How to Document These. 2023. Dostupné tiež z: <https://enkonix.com/blog/functional-requirements-vs-non-functional/>. Použité dňa 12.06.2023.