Vysoká škola polytechnická Jihlava

Aplikovaná informatika

**Demonstrační nástroj Man-in-the-middle útoků na Wi-Fi sítě**

Bakalářská práce

Autor práce: Jiří Karmasin

Vedoucí práce: Mgr. Antonín Přibyl

Jihlava 2023

**Vysoká škola polytechnická Jihlava**

Tolstého 16, 586 01 Jihlava

**ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

Autor práce: **Jiří Karmasin**

Studijní program: Aplikovaná informatika

Obor: Aplikovaná informatika

Název práce: **Demonstrační nástroj Man-in-the-middle útoků na Wi-Fi sítě**

Cíl práce: Cílem práce je navrhnout a implementovat demonstrační nástroj k popisu a vysvětlení principu Man-in-the-middle (MITM) útoků na bezdrátové sítě. Tento nástroj poslouží k názorným ukázkám během školení kybernetické bezpečnosti především pro příslušníky Armády České republiky, proto musí být uživatelské rozhraní dostatečně intuitivní a přehledné. Součástí nástroje bude i popis relevantních protiopatření, které zvyšují imunitu proti MITM útokům. Nástroj bude implementován na platformě KALI Linux. Rovněž budou vytvořeny návodné a dokumentační nástroje (poster, brožura) pro cílové publikum pro zvýšení účinku školení s využitím tohoto nástroje.

Abstrakt

TODO

Klíčová slova

První klíčové slovo; druhé klíčové slovo; třetí klíčové slovo

Abstract

Vlastní text.

Keywords

První klíčové slovo; druhé klíčové slovo; třetí klíčové slovo

Prohlašuji, že předložená bakalářská práce je původní a zpracoval/a jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem v práci neporušil/a autorská práva (ve smyslu zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů, v platném znění, dále též „**AZ**“).

Byl/a jsem seznámen/a s tím, že na mou bakalářskou práci se plně vztahuje **AZ**, zejména § 60 (školní dílo).

Podle § 47b zákona o vysokých školách souhlasím se zveřejněním své práce podle směrnice prorektora pro studium č. 2/2020, a to bez ohledu na výsledek obhajoby.

Beru na vědomí, že VŠPJ má právo na uzavření licenční smlouvy o užití mé bakalářské práce a prohlašuji, že **souhlasím** s případným užitím mé bakalářské práce (prodej, zapůjčení apod.).

Jsem si vědom/a toho, že užít své bakalářské práce či poskytnout licenci k jejímu využití mohu jen se souhlasem VŠPJ, která má právo ode mě požadovat přiměřený příspěvek na úhradu nákladů, vynaložených vysokou školou na vytvoření díla (až do jejich skutečné výše), z výdělku dosaženého v souvislosti s užitím díla či poskytnutím licence.

V Jihlavě dne Klikněte nebo klepněte sem a zadejte datum.

…………………………………….

Podpis studenta

Obsah

[Seznam obrázků 7](#_Toc153705103)

[Seznam tabulek 8](#_Toc153705104)

[Seznam zkratek 9](#_Toc153705105)

[Úvod 10](#_Toc153705106)

[1 Man-in-the-middle útok 11](#_Toc153705107)

[2 Analýza síťových protokolů 13](#_Toc153705108)

[2.1 OSI model 13](#_Toc153705109)

[2.2 Inicializace připojení k Wi-Fi 14](#_Toc153705110)

[2.3 Proces autentizace 14](#_Toc153705111)

[2.4 ARP 16](#_Toc153705112)

[2.5 SSL 16](#_Toc153705113)

[3 Analýza vektorů útoku 17](#_Toc153705114)

[3.1 Připojení se k cílové zabezpečené Wi-Fi 17](#_Toc153705115)

[3.2 Přesměrování datového toku oběti 18](#_Toc153705116)

[3.3 Analýza a modifikace datového toku 19](#_Toc153705117)

[3.4 Analýza existujících nástrojů 20](#_Toc153705118)

[4 Návrh implementace 21](#_Toc153705119)

[4.1 Zdůvodnění použitého jazyka implementace 21](#_Toc153705120)

[4.2 Návrh funkcionalit konzolového back-endu 21](#_Toc153705121)

[4.3 Návrh GUI 21](#_Toc153705122)

[5 Implementace 22](#_Toc153705123)

[5.1 Implementace konzolového back-endu 22](#_Toc153705124)

[5.2 Implementace GUI 22](#_Toc153705125)

[6 Testování 23](#_Toc153705126)

[6.1 Testy funkčnosti 23](#_Toc153705127)

[6.2 Testy výkonosti 23](#_Toc153705128)

[Závěr 24](#_Toc153705129)

[Seznam použité literatury 25](#_Toc153705130)

[Přílohy 27](#_Toc153705131)

# Seznam obrázků

[Obrázek 1: Počáteční stav 11](#_Toc153721035)

[Obrázek 2: Po úspěšném MITM útoku 12](#_Toc153721036)

[Obrázek 3: ISO/OSI referenční model 13](#_Toc153721037)

[Obrázek 4 - bettercap 20](#_Toc153721038)

[Obrázek 5 – Ettercap 20](#_Toc153721039)

# Seznam tabulek

# Seznam zkratek

AČR Armáda České republiky

AES Advanced Encryption Standard

AP Access Point (Přístupový bod)

CCMP Counter Cipher Mode with Block Chaining Message Authentication Code Protocol

NATO North Atlantic Treaty Organization (Severoatlantická aliance)

NÚKIB Nádorní úřad pro kybernetickou a informační bezpečnost

MITM Man-in-the-middle

MAC Medium Access Control

MIC Message Integrity Code

IP Internet protocol

TCP Transmission Control Protocol

UDP User datagram protocol

EAPOL Extensible authentication protocol over LAN

LAN Local area network

PSK Pre-Shared Key

SSID Service Set Identifier

SSL Secure Socket Layer

URL Unified Resource Location

WEP Wired Equivalent Privacy

WPA Wi-Fi Protected Acess

WPS Wi-Fi Protected Setup

# Úvod

Dnešní svět se stále více digitalizovaný. To s sebou nese mnoho pozitivních jevů jako je zvýšení produktivity téměř všech lidských činností, rychlost komunikace mezi lidmi a možnost mít všechny myslitelné informace doslova na dlani kdykoliv si člověk přeje. Mimoto však s sebou přináší i mnoho bezpečnostních hrozeb a zranitelností, které před příchodem digitálního věku neexistovaly.

Kybernetická bezpečnost se tedy stává jedním z nejnaléhavějších problémů, kterým čelíme v této době. Jedním z hlavních způsobů, kterým zvyšovat bezpečnost v kyberprostoru, je dle vyjádření NÚKIB školení a osvěta v oblasti kybernetické bezpečnosti a chování v digitálním prostoru. Školení jsou klíčová pro posilování odolnosti organizací vůči kybernetickým útokům.

Jako specialista kybernetické bezpečnosti sloužící v Armádě České republiky je mojí povinností takováto školení provádět. Z mých letitých zkušeností je zřejmé, že teoretický výklad na povinných školeních nemá zdaleka takový dopad jako praktická ukázka probírané látky. A právě proto se moje bakalářská práce zaměřuje na návrh a implementaci demonstračního nástroje, který má za cíl poskytnout názorné ukázky útoku man-in-the-middle (MITM) a tím pomoci školitelům a studentům lépe porozumět této kybernetické hrozbě.

MITM útoky patří mezi nebezpečné techniky, při kterých útočník vstupuje mezi komunikující strany a může odposlouchávat a modifikovat datový tok. Tato zranitelnost Wi-Fi sítí je známá a často zneužívaná, což pro příslušníky resortu Ministerstva obrany představuje hrozbu při využívání Wi-Fi připojení k internetu.

Cílovým publikem bude široká škála posluchačů při školení od neznalých uživatelů koncových zařízení (smartphonů) až po IT techniky a síťové administrátory. Každému posluchači by při tom měla demonstrační ukázka přinést nový poznatek. Výsledný nástroj by tedy měl být přehledný, a přitom měl mít možnost poskytovat hlubší a komplexnější informace o fázích útoku, pokud by to o měl posluchač zájem.

Práce bude dále diskutovat o klíčových aspektech man-in-the-middle útoků, jejich důsledcích, a především ochranných opatřeních proti nim. Kromě toho popíše proces vývoje demonstračního nástroje.

Cílem práce je tedy nejen vytvořit funkční nástroj pro demonstrace, ale také přispět k zvýšení povědomí o MITM útocích a zlepšení schopností obrany proti nim. Věřím, že vypracovaný projekt má potenciál přispět k bezpečnosti digitálního prostoru a k lepší přípravě jednotlivců na výzvy, které s sebou nese kybernetická éra.

# Man-in-the-middle útok

Kybernetický útok Man-in-the-middle, který by se dal přeložit také jako Muž-uprostřed, je typ útoku na bezdrátovou komunikaci dvou zařízení. Zneužívá nepopiratelného faktu, že na rozdíl od metalického ethernetového nebo optického síťového spojení, je bezdrátová komunikace zajišťována pomocí vysílání elektromagnetických vln, které se šíří všesměrově do okolí. U například ethernetového spojení pomocí UTP kabelu je jednoznačné, odkud a kam signál drátem prochází a elektromagnetické vyzařování samotného kabelu je zanedbatelné. U bezdrátových technologií, jako je Wi-Fi, je však nejednoznačné, ze kterého zařízení přijatý signál vychází.

Útočník může pomocí zachytávání komunikace ostatních zařízení, nedokonalostem v používaných protokolech a vysíláním specifických rámců zapříčinit, že se vměstná do komunikace mezi zařízení oběti a přípojný bod. Veškerou komunikaci potom může monitorovat anebo pozměnit, čímž kompromituje integritu a důvěrnost dat, což jsou dva ze tří pilířů kybernetické bezpečnosti.

Obecná posloupnost MITM útoku na bezdrátové sítě vypadá následovně.

1. *Počáteční stav*.

Klient je připojen k AP (Access Pointu), tedy přístupovému bodu k internetu pomocí hesla a šifrovaně komunikuje.

Obsah obrázku skica, kresba, kreslené, klipart

Popis byl vytvořen automaticky

Obrázek 1: Počáteční stav

*(Zdroj: vlastní)*

1. *Fáze útoku – Rekognoskace, deautentifikace, lámání hesla*.

Útočník pomocí svého Wi-Fi adaptéru zachytává v okolí se šířící rámce a z nich extrahuje potřebné informace o AP a zařízení oběti. Jakmile má potřebné znalosti, začne vysílat deautentifikační rámce a zároveň zachytává pokus o opětovné připojení k síti odpojené oběti. Po jejich zachycení má potřebné zdroje k pokoušení se o prolomení hesla ze zašifrovaných zpráv. Lámání helsa probíhá například offlinovým slovníkovým útokem a bude popsán v pozdějších kapitolách.

1. *Fáze útoku – útočník je v síti a provádí Man-in-the-middle*.

Se získaným heslem se útočník připojuje k síti a jako další zařízení a pomocí techniky APR poisoningu zapříčiňuje, že komunikace oběti nejde napřímo k AP, ale jde přes útočníkovo zařízení. Tím se útočník stává „mužem uprostřed“.

1. *Fáze útoku – výkonná fáze.*

Dosavadní činnost útočníka žádným způsobem negativně neovlivňuje komunikaci oběti. V této fázi útoku ale nastává samotné zneužití provedeného MITM například odposloucháváním komunikace (zachytávání DNS dotazů), přesměrování na falešný web (DNS spoofing), nebo čtení obsahu samotné komunikace po prolomení SSL zabezpečení například pomocí SSLsplitu.

Obsah obrázku skica, kresba, kreslené, klipart

Popis byl vytvořen automaticky

Obrázek 2: Po úspěšném MITM útoku

*(Zdroj: vlastní)*

1. *Fáze útoku – Exfiltrace.*

Úspěšný útok je takový, o kterém oběť ani neví, že se stal. V poslední fázi tedy útočník reverzně vrazí pozměněné ARP záznamy na původní hodnoty a odpojuje se od sítě.

V současnosti existuje řada nástrojů, která celý tento proces balí do lehce použitelného balíčku. Jsou jimi například *bettercap, Gerix s ettercap* s GUI nebo *wifiphisher* (Sharma, 2017, Baloch, 2015). Všechny jsou ale pro demonstraci útoků laikům špatně použitelné, protože jsou to buďto konzolové aplikace, které jsou pro diváky nesmírně odrazující, nebo neobsahují žádné popisné doplňující informace k útoku. Jejich používání by tedy kladlo velké požadavky na znalosti přednášejícího, který by musel být schopný okomentovat celý proces sám bez pomoci. Motivací k vytvoření mého nástroje je rovnou v GUI aplikaci přehledně zobrazovat všechny potřebné informace o probíhajících částech útoku tak, aby byl přednášející schopen provést demonstraci i s jen základními znalostmi.

Celý tento postup bude v následujících částech práce podrobněji popsán a ve výsledném nástroji bude možné jej krok po kroku demonstrovat s náležitým komentářem a vysvětlivkami.

# Analýza síťových protokolů

K úspěšnému provedení MITM útoku je nutné znát alespoň základní protokoly, které se využívají při bezdrátové Wi-Fi komunikaci. Právě nedokonalosti a funkcionalita těchto protokolů umožňují útok provést.

Bezdrátovou komunikaci, jejíž napadení bude práce demonstrovat je Wi-Fi. Wi-Fi připojení koncových zařízení k internetu využívá sadu standardizovaných protokolů IEEE 802.11 (IEEE Standard for Information technology, 2016), díky kterým je vůbec možné samotnou komunikaci navázat a provozovat. V první části práce popíšu a analyzuji základní sadu protokolů. Na tuto analýzu navážu v následující kapitole, kde se zaměřím na slabá místa jednotlivých aspektů komunikace. Dobře znát princip těchto protokolů je tedy zásadní pro pozdější zneužití jejich zranitelností.

## OSI model

Útok MITM probíhá na 2. spojové (neboli linkové) vrstvě referenčního ISO/OSI modelu. Pracuje tedy s tzv. rámci. Po úspěšném přesměrování rámců na zařízení útočníka pak může probíhat následné využití MITM na vyšších vrstvách. Při SSLsplitu je to šifrování, které se řadí do 5. relační vrstvy (Wu, 2020). Při práci s DNS dotazy se pak pracuje na nejvyšší, aplikační vrstvě (Rymond, 2022).

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Obrázek 3: ISO/OSI referenční model

*(Zdroj: wikipedia.org)*

## Inicializace připojení k Wi-Fi

Samotné připojení bezdrátového zařízení k Wi-Fi probíhá v několika krocích s využitím různých rámců (frames). Zde uvádím výčet těch nejdůležitějších:

* Beacon frame – tento rámec je v pravidelných intervalech vysílán AP, pokud je Wi-Fi síť nastavená jako viditelná. Obsahuje mimo jiné SSID, tedy název sítě, a informace o zabezpečení. Účelem je oznamovat do okolí dostupnost sítě.
* Probe request – rámec vysílají koncová zařízení, když aktivně hledají požadovanou síť, která není nastavená jako veřejná. Obsahuje SSID hledané sítě.
* Probe response – odpověď AP na probe request s informacemi jako v beacon framu.
* Authentication request frame – rámec zasílající klient k přístupovému bodu, v němž ho žádá o ověření. Toto ověření může být buď otevřené (pak není spojení šifrování), nebo pomocí hesla (shared key authentication). Toto spojení je bezpečnější.
* Authentication response frame – odpověď AP směrem ke koncovému zařízení. Spojení buďto potvrzuje, nebo zamítá při otevřeném spojení. U spojení s ověřovacím klíčem si vyžaduje heslem zašifrovaný řetězec (challenge text). Při jeho úspěšném dešifrování přístupovým bodem, je klientské zařízení ověřeno a připojeno.
* Association request/response frame – klíčový prvek v procesu připojování. Po úspěšném ověření v předcházejícím kroku dochází k samotnému připojení. V association response rámci je klientovi zaslána jeho nově přiřazená IP adresa.

## Proces autentizace

Samotná autentizace z předcházející části pak může probíhat několika způsoby. Ty jsou časem přidávány s tím, jak předešlé ztrácejí na bezpečnosti kvůli jejich nedokonalostem a prolomení.

### WEP (Wired Equivalent Privacy)

WEP je původní zabezpečovací protokol pro bezdrátové sítě Wi-Fi, zavedený v rámci standardu IEEE 802.11 v roce 1997. Cílem bylo poskytnout bezdrátovým sítím zabezpečení srovnatelné s tím, které je k dispozici v drátových metalických propojeních.

Autentizace pomocí WEP se zakládá na sdíleném klíči. Tento symetrický klíč je používán jak klientem, tak přístupovým bodem. Při autentizaci je poslán autentizační požadavek, který obsahuje zašifrované informace. Pokud přístupový bod dokáže tyto informace dešifrovat pomocí stejného klíče, klient je považován za ověřeného. WEP však používá slabý algoritmus šifrování (RC4), což vede k jeho zranitelnostem.

### WPA, WPA2, WPA3

WPA (Wi-Fi Protected Access) byl zaveden jako dočasné řešení k překonání slabostí WEP. Používá TKIP (Temporal Key Integrity Protocol) pro šifrování. Autentizace u všech protokolů rodiny WPA spočívá v použití tzv. four-way handshake procesu. Tento proces ověřuje, že jak klient, tak přístupový bod mají správný klíč, a současně vytváří nové klíče pro ochranu datových rámců.

WPA2 je standardizovaný protokol, který nahradil WPA. Používá pokročilejší šifrování AES (Advanced Encryption Standard) a protokol CCMP (Counter Cipher Mode with Block Chaining Message Authentication Code Protocol).

WPA3 je nejnovější standard, který přináší zlepšení v oblasti zabezpečení a jednodušší připojení zařízení. Používá ještě silnější šifrovací metody a zlepšuje ochranu proti útokům, jako je brute-force pro opakované pokusy o přihlášení.

### Four-way handshake

Čtyřkrokový proces autentizace, známý jako "four-way handshake", je klíčovým prvkem zabezpečení Wi-Fi sítí, který se používá v protokolech WPA a WPA2. Jeho hlavním účelem je ověřit, že jak klient (zařízení, které se snaží připojit k síti), tak přístupový bod mají správný předsdílený klíč (Pre-Shared Key, PSK), a zároveň zajistit, že pro každou sezení bude vytvořen jedinečný šifrovací klíč.

Posloupnost zasílaných zpráv:

1. **Zpráva:** Přístupový bod (AP) pošle zprávu klientovi. Tato zpráva obsahuje ANonce (číslo použité jednou generované přístupovým bodem) a další informace potřebné k vytvoření klíče. Tento krok neobsahuje žádné šifrování.
2. **Zpráva:** Klient odpovídá zprávou, která obsahuje SNonce (číslo použité jednou generované klientem), informace potřebné k vytvoření klíče, a ověření, že klient zná předsdílený klíč (PSK). Tato zpráva je autentizována pomocí MIC (Message Integrity Code), který je vytvořen z předsdíleného klíče.
3. **Zpráva:** Přístupový bod nyní potvrdí, že i on zná PSK, odesláním další zprávy, která obsahuje ANonce. Tato zpráva také obsahuje šifrovací klíče použité k šifrování provozu a MIC. Klient nyní má všechny informace potřebné k vytvoření Temporal Key (TK), který bude používán pro šifrování a dešifrování dat.
4. **Zpráva:** Klient odpovídá potvrzením, že zpráva z třetího kroku byla přijata a že TK byl úspěšně vytvořen. Po této zprávě může začít šifrovaná komunikace mezi klientem a přístupovým bodem.

Celý proces zajišťuje, že obě strany (klient a přístupový bod) mají stejný šifrovací klíč a že tento klíč je jedinečný pro každou sezení. To pomáhá zabránit různým druhům útoků, včetně těch, které se snaží znovu použít staré šifrovací klíče.

### WPS (Wi-Fi Protected Setup)

WPS je navržen pro usnadnění procesu nastavení a připojení bezdrátových sítí. Jeho cílem je umožnit uživatelům připojit zařízení k síti snadno, často stisknutím tlačítka nebo zadáním PIN kódu.

WPS umožňuje uživatelům jednoduše stisknout tlačítko na routeru a na zařízení, které se připojuje, což spustí proces autentizace a automaticky nastaví síťové připojení.

WPS PIN je číslicový kód, který je potřeba zadat na zařízení pro jeho připojení k síti. Tento kód je obvykle umístěn na routeru nebo je možné ho generovat.

Je důležité poznamenat, že WPS může představovat bezpečnostní riziko, protože PIN metoda je náchylná k útokům brute-force. Proto odborníci doporučují WPS deaktivovat.

## ARP

ARP je zkratka „adress resolution protocol“. Jeho účelem je převod fyzické MAC adresy na síťovou IP adresu. Aby mohlo zařízení připojené do sítě komunikovat s okolními zařízeními v lokální síti, musí znát jejich i svoji MAC adresu. K tomu slouží dva hlavní rámce:

* ARP request – tímto rámcem se zařízení dotazuje na broadcastové adrese (FF:FF:FF:FF:FF:FF) na MAC adresu zařízení se známou IP adresou.
* ARP response – zařízení, které rozpoznalo svoji IP adresu v ARP requestu odpovídá na MAC adresu odesílatele rámec se svojí MAC adresou.

Tím je dokončen proces připojení a koncové zařízení je plně funkční ve Wi-Fi síti.

## SSL

SSL (Secure Sockets Layer) je zabezpečovací technologie, která se používá k vytvoření šifrovaného spojení mezi webovým serverem a prohlížečem. Toto spojení zajišťuje, že všechna data přenášená mezi webovým serverem a prohlížečem zůstanou soukromá a integrovaná. SSL je široce používán pro zabezpečení transakcí na internetu.

Princip fungování SSL lze stručně popsat následovně. Prvním krokem je navázání spojení mezi prohlížečem a serverem s SSL zabezpečením tím, že prohlížeč tento server požádá o jeho identitu. Server odpoví zasláním kopie svého SSL certifikátu. SSL certifikát obsahuje veřejný klíč potřebný k zašifrování dat odesílaných serveru. Prohlížeč kontroluje, zda je certifikát důvěryhodný. Pokud je certifikát platný, prohlížeč vygeneruje symetrický šifrovací klíč pro bezpečnou komunikaci se serverem. Tento šifrovací klíč je pak zašifrován pomocí veřejného klíče z certifikátu a odeslán zpět serveru. Server dešifruje tento klíč pomocí svého soukromého klíče. Jakmile má server dešifrovaný symetrický klíč, obě strany (server i prohlížeč) používají tento klíč pro šifrování a dešifrování dat, která si mezi sebou posílají. Tímto způsobem je zajištěno bezpečné spojení, které chrání integritu a důvěrnost přenášených dat. Po dokončení komunikace se spojení uzavře a klíče jsou zneplatněny.

Protokol SSL byl nahrazen modernější a bezpečnější technologií TLS (Transport Layer Security), která poskytuje lepší ochranu. Nicméně, termín "SSL" je stále běžně používán i když se ve skutečnosti často jedná o TLS.

# Analýza vektorů útoku

V této kapitole popíšu jednotlivé kroky potřebné k provedení MITM útoku a zaměřím se na zranitelnosti protokolů, které tato posloupnost zneužívá.

## Připojení se k cílové zabezpečené Wi-Fi

Prvním krokem k úspěšném provedení útoku MITM je připojení se k cílové zabezpečené Wi-Fi, ke které nezná útočník heslo. To provedu postupným odpojením připojeného klienta od sítě, zachycením přihlašovacích rámců při opětovném pokusu o připojení a nakonec dešifrováním hesla ze zachycených rámců.

### Deutaentifikace připojeného zařízení

Aireplay-ng je součástí Aircrack-ng, což je sada nástrojů pro testování zabezpečení Wi-Fi sítí. Právě tento nástroj použiji k odpojení klienta od přístupového bodu (AP) pomocí příkazu *aireplay-ng -deauthenticate*. Tento proces funguje na základě odesílání deautentifikačních rámců, což jsou speciální typy rámců definované ve standardu IEEE 802.11, určené k ukončení spojení mezi klientem a AP. Princip fungování deautentifikace lze popsat takto:

1. **Zachycení informací o síti:** Nejprve je potřeba zachytit informace o cílovém přístupovém bodu a připojených klientech. To zahrnuje MAC adresy AP a klientů.
2. **Odeslání deautentifikačních rámců:** Nástroj aireplay-ng poté odesílá deautentifikační rámců do sítě. Tyto rámce budu odesílat na adresu konkrétního klienta, což způsobí odpojení tohoto klienta od AP.
3. **Přerušení připojení:** Když je deautentifikační rámec přijat klientem standard IEEE 802.11 vyžaduje, aby příjemce rámce ukončil spojení. Toto je bezpečnostní prvek, který je v normálním provozu používán pro řádné ukončení spojení.
4. **Zneužití:** Právě tuto vlastnost zneužiji pro provedení MITM útoku a k pokusu o prolomení hesla sítě. Když jsou klienti odpojeni, často se automaticky pokusí znovu připojit, což mi umožní zachytit proces opětovného připojení (EAPOL rámce 4-way handshaku) a získat data potřebná pro prolomení hesla.

### Zachycení EAPOL rámců

EAPOL (Extensible Authentication Protocol over LAN) rámce jsou součástí protokolu EAP (Extensible Authentication Protocol), který se používá v různých autentizačních metodách. V kontextu Wi-Fi bezpečnosti jsou EAPOL rámce klíčové pro autentizační procesy, v mém případě konkrétně four-way handshake v WPA/WPA2, kde jsou používány k výměně šifrovacích klíčů a ověřovacích informací mezi klientem a přístupovým bodem.

Pro úspěšné zachycení a následnou analýzu zašifrovaného hesla v bezdrátové síti WPA/WPA2 je potřeba zachytit konkrétní sekvenci čtyř rámců "four-way handshake" mezi klientem a přístupovým bodem. Tyto rámce obsahují veškeré informace potřebné pro pokus o prolomení WPA/WPA2 hesla pomocí technik jako je brute-force nebo slovníkový útok. Tento proces může být časově náročný a složitý, zvláště pokud je heslo silné (dlouhé a obsahuje kombinaci písmen, čísel a speciálních znaků).

### Lámání WEP2 hesla

Pokud jde o prolomení zašifrovaných hesel existuje několik nástrojů, které jsou běžně používány v oblasti testování zabezpečení sítí. Těmi nejznámějšími v prostředí Kali Linuxu jsou *aircrack-ng*, *John the Ripper*, *hachcat* a například *Cowpatty*.

Pro svoji práci použiji nástroj *hashcat*, který je vysoce výkoný a podporuje mnoho různých algoritmů šifrování.

Při používání takovýchto nástrojů je důležité mít na paměti, že úspěch prolomení hesla závisí na mnoha faktorech, včetně složitosti hesla, výkonu hardwaru používaného k prolomení a dostupnosti správných dat (jako je například kompletní four-way handshake). Pro demonstrační nástroj je velice důležité, aby bylo heslo prolomeno v relativně krátkém čase; řádově několik sekund. Proto použiju lámání slovníkovým útokem. Tento slovník vytvořím například nástrojem *crunch* a správné heslo manuálně vložím na pozici, která mi zajistí prolomení hesla při demonstraci v požadovaném čase. Vytváření slovníku pro útok není součástí této práce.

Po úspěšném prolomení hesla je útočníkovi umožněno se k síti připojit jako další běžné zařízení. Tím je dokončen první krok útoku MITM.

## Přesměrování datového toku oběti

Druhou fází útoku je přesměrovat datový tok oběti k AP přes útočníkův adaptér tak jak je znázorněno na Obrázek 2: Po úspěšném MITM útoku. To provedu ARP poisoningem.

### ARP poisoning

ARP poisoning, také známý jako ARP spoofing nebo ARP cache poisoning, je fáze útoku, při kterém útočník posílá falešné ARP (Address Resolution Protocol) zprávy do síťového provozu. Tento útok je zaměřen na změnu ARP tabulky v síťovém zařízení (v mém případě v zařízení oběti a v AP), což vede k přesměrování síťového provozu skrze útočníka. ARP je protokol používaný v lokálních sítích pro nalezení MAC adresy, která odpovídá konkrétní IP adrese.

Princip fungování ARP poisoning je následující:

1. **Normální funkce ARP:** Ve standardní síťové komunikaci, když zařízení potřebuje zjistit MAC adresu jiného zařízení na stejné lokální síti, odešle *ARP request* (požadavek). Když příslušné zařízení přijme tento požadavek, odpoví s *ARP reply* (odpovědí), která obsahuje jeho MAC adresu.
2. **Zasílání falešných ARP odpovědí:** Během ARP poisoning útoku útočník zasílá falešné ARP odpovědi do sítě, i když nebyl vyslán žádný *ARP request*. Tyto falešné odpovědi mohou informovat ostatní zařízení v síti, že MAC adresa útočníka odpovídá IP adrese jiného legitimního zařízení v síti (například AP nebo telefonu).
3. **Modifikace ARP tabulky:** Když ostatní zařízení v síti přijmou tyto falešné ARP odpovědi, aktualizují své ARP tabulky. V důsledku toho bude veškerý provoz určený pro určité IP adresy přesměrován na útočníka.
4. **Přesměrování provozu:** Jakmile je ARP tabulka otrávena, útočník může zachytávat, upravovat nebo přerušovat data odesílaná mezi zařízeními. To umožňuje provádět další činnosti, jako je špehování DNS zpráv, SSLsplit a další.
5. **Transparentnost útoku:** ARP poisoning je často transparentní pro uživatele nebo zařízení v síti, což znamená, že mohou pokračovat v používání sítě bez toho, aniž by si byli vědomi, že jejich síťový provoz je manipulován nebo monitorován.

ARP poisoning je účinný v sítích, které používají ARP a nemají implementovány opatření proti ARP spoofingu, jako je například statické ARP tabulky nebo síťová bezpečnostní řešení, která detekují a blokují neobvyklý ARP provoz.

Tím je dokončena druhá část útoku, ve které se mi podařilo po připojení do sítě přesměrovat datové rámce na moji stanici, kde na ně aplikuji další část útoku.

## Analýza a modifikace datového toku

V poslední části ukázky budu demonstrovat, co všechno je možné dělat v okamžiku, kdy je útočník prostředníkem v datovém toku. Bude to především filtrace DNS dotazů, která mi umožní zjistit, na které stránky se oběť chtěla připojit a také SSLsplit, která umožní čtení samotného obsahu komunikace, jako například přihlašovací údaje.

### Filtrování DNS dotazů

DNS dotazy na překlad URL adresy na IP nejsou do této doby ve velkém šifrovány (Internet Society, 2023). Tím mohu jednoduše vyfiltrovat všechnu komunikaci na UDP port 53 a z paketů vyextrahovat požadované stránky k navštívení.

### SSLsplit

SSLsplit je nástroj používaný pro provedení poslední fáze MITM útoku. Jeho primárním účelem je zachytávat a manipulovat s HTTPS a jinými šifrovanými protokoly, jako jsou SMTPS, POP3S, IMAPS, a další. SSLsplit umožňuje útočníkovi sledovat a modifikovat síťový provoz, který je obvykle chráněn šifrováním. SSLsplit funguje následovně:

1. **Zachycení síťového provozu:** SSLsplit funguje tak, že se umístí mezi dvě komunikující strany (například mezi uživatelem a webovým serverem). Toho je obvykle dosaženo pomocí ARP poisoning nebo jiných technik pro změnu směrování síťového provozu skrze zařízení, na kterém běží SSLsplit.
2. **Rozložení šifrované komunikace:** Když projde šifrovaný provoz (například HTTPS) skrze SSLsplit, nástroj rozloží šifrované spojení. To znamená, že SSLsplit naváže dvě samostatná spojení: jedno s klientem a druhé se serverem.
3. **Předstírání identity:** SSLsplit se předstírá jako cílový server pro klienta a jako klient pro cílový server. To zahrnuje vytváření falešných SSL certifikátů v reálném čase, které jsou představovány klientovi, aby bylo možné šifrované spojení navázat.
4. **Dekódování a případná manipulace:** Po úspěšném navázání těchto spojení může SSLsplit dešifrovat a zaznamenávat přenášená data. Také je možné data modifikovat, než budou předána dál.
5. **Přeposlání dat:** SSLsplit pak předává data mezi klientem a serverem, aniž by byla narušena funkčnost komunikace, což umožňuje útočníkovi sledovat nebo modifikovat data, aniž by to bylo zřejmé pro obě komunikující strany.

## Analýza existujících nástrojů

Na systému Kali Linux existuje řada nástrojů, které se problematikou Man-in-the-middle zabývají. Buďto jsou ale konzolové, což je pro demonstrační a výukové účely krajně nevhodné, nebo se snaží o co největší efektivitu ve splnění jejich úkolu a zanedbávají demonstrační a vysvětlující aspekt uživatelského prostředí. Za všechny uvádím dva příklady.

* bettercap – tento konzolový tool dokáže provést kompletní MITM útok, ale kvůli chybějícímu uživatelskému prostředí je nevhodný pro školení z kybernetické bezpečnosti.

Obsah obrázku text, snímek obrazovky, software, Multimediální software

Popis byl vytvořen automaticky

Obrázek 4 - bettercap

*(Zdroj: vlastní)*

* Ettercap – tento nástroj má GUI, ale stále by jeho použití při školení vyžadovalo značné znalosti školitele, protože neobsahuje žádné pomocné informační texty.



Obrázek 5 – Ettercap

*(Zdroj: vlastní)*

# Návrh implementace

Vzhledem k faktu, že aplikace má sloužit pouze jako demonstrativní nástroj kybernetického útoku při školení, je nezbytně nutné, aby byla implementována s patřičně kvalitním a přehledným uživatelským rozhraním. Samotný útok Man-in-the-middle je poměrně komplexní a má několik na sebe navazujících kroků. Není mým cílem implementovat aplikaci, která by mohla sloužit i pro samostudium a která by sama o sobě poskytovala dostatek informací k pochopení problematiky nebezpečnosti bezdrátových sítí. Má sloužit pouze jako pomocný materiál, kdy je přednášející školitel seznámen s problematikou MITM útoku a zná jeho principy. DeMITMo mu poskytne názornou ukázku procedury nabourání se do cílové sítě a popřípadě obecně nastíní, které zranitelnosti a protiopatření v jednotlivých korcích ústně vysvětlit. Samotné detailní předání znalostí tak, aby je pochopilo i neznalé publikum, pak bude pouze na přednášejícím.

S tímto zadáním tedy není třeba znova implementovat to, co už bylo naprogramováno. Využiju existující nástroje a pospojuji je do jednoho na sebe navazujícího celku, kdy bude prioritní grafická přehlednost. Všechny použité nástroje jsou konzolové aplikace v systému Linux, distribuci Kali. Pro praktické používání je rozhraní v příkazové řádce naprosto pochopitelné, ale do demonstračního nástroje se nehodí a jakékoliv publikum by spolehlivě odradilo od dané problematiky.

## Použité nástroje

K získání všech informací potřebných k úspěšnému uskutečnění útoku budu využívat jak systémové nástroje samotného Linuxu, tak externí balíčky s dodatečnou funkcionalitou. Systémovými nástroji, se kterými je třeba pracovat, aby byly získány potřebné informace jsou především:

* iwconfig – pro získání dostupných bezdrátových rozhraní a
* ifconfig – pro zapnutí a vypnutí požadovaníého rozhraní)
* service – pro spuštění služby NetworkManager)
* nmcli – pro připojení se a odpojení se od sítě

Doinstalované balíčky s požadovanou funkcionalitou:

* aircrak-ng suite – jedná se o kompletní balík mnoha nástrojů sloužících k penetračnímu testování bezdrátových sítí. Budu z něj používat tyto utility:
  + airmon-ng
  + airodum-ng
  + aireplay-ng
  + aircrack-ng
* arp-spoof – provedení ARP poisoningu
* tshark – pro filtrování zachycených datových paketů
* crunch – pro vytvoření slovníku hesel dle zadaných parametrů, pokud nebude použitý žádný už vytvořený

## Zdůvodnění použitého jazyka implementace

Všechny výše uvedené nástroje budou muset být volány z jedné hlavní aplikace, která bude jejich výstup zpracovávat a bude se také starat o grafické uživatelské rozhraní. Pro vytvoření této aplikace jsem zvolil skriptovací jazyk Python ve verzi 3.

Výhodou Pythonu je jeho relativní jednoduchost, a hlavně dostupnost mnoha doinstalovatelných balíčků, které skvěle implementují všechnu mnou požadovanou funkcionalitu, jako jsou například zpracování *csv* souborů a práce a filtrování jednotlivých paketů v zachyceném datové komunikaci.

## Návrh funkcionalit konzolového back-endu

Původním záměrem bylo navrhnout aplikaci tak, aby mohla být spuštěna i s pouze textovým rozhraním v konzoli. Grafická nástavba by byla volitelná a na vnitřní funkcionalitu by neměla vliv. Nicméně během návrhu grafického rozhraní jsem ujistil, že jednoduší bude jednotlivé dílčí činnosti vykonávané během útoku navázat funkcemi přímo na ovládací prvky grafického uživatelského rozhraní. Textová podoba aplikace by stejně neměla využití, protože pro demonstraci při školení je psaní příkazů v konzoli krajně nevhodné a pro sledující nezáživné. Navíc existující aplikace jako bettercap už toto umožňují a nemá přínos programovat už jednou implementované.

Vnitřní back-endové úkony aplikace jsou tedy nedílně spojeny v ovládacími prvky uživatelského rozhraní.

## Návrh GUI

Celý proces MITM útoku je poměrně komplexní a skládá se z posloupnosti kroků. Některé tyto kroky je nutné provádět souběžně, ale většina je vykonatelná až poté, co jsou jí dostupné data a vstupy z kroků předešlých.

Rozhodl jsem se tedy pro názornost tento proces rozdělit do pěti segmentů. V každé takové části budou tři až pět spolu souvisejících činností. Těmito oblastmi budou:

* získání informací o bezdrátových zařízeních v okolí a vybrání cíle útoku
* zachycení rámců handshaku při připojení
* dešifrování přístupového hesla k síti ze zachycených rámců
* aplikace samotného MITM útoku skrze ARP poisoning a zachycení komunikace cíle
* demonstrace úspěšnosti útoku

Pro demonstraci je vhodné sledující nezahltit velkým množstvím informací. V každém okamžiku tedy bude v aplikaci zobrazen pouze jeden aktuální segment a s ním související ovládací prvky. Nejvhodnějším přístupem tedy bude pro každý segment vytvořit jednu záložku v pythonovském notebooku a postupně zobrazovat každou tuto záložku tak, jak se bude vykonávat MITM útok.

Obsah obrázku text, elektronika, snímek obrazovky, displej

Popis byl vytvořen automaticky

Obrázek 6 - Základní návrh GUI

*(Zdroj: vlastní)*

V jednotlivých záložkách budou ovládací prvky nezbytné k dokončení daného segmentu

### Vyhledání cílů

Prvním krokem v provedení útoku je vyhledat potenciální cíle útoku. K tomu je potřeba proskenovat okolní komunikaci bezdrátovým rozhraním, které je přepnuté do monitorovacího módu. Tato záložka tedy bude obsahovat prvky pro výběr rozhraní, jeho přepnutí (vypnutí) do monitorovacího módu, tlačítko pro skenování okolí a nakonec vhodný prostředek k jednoduchému vybrání cíle z dostupných zařízení. Úspěšné zvolení tohoto cíle splní všechny potřebné kroky v tomto segmentu a zpřístupní segment následující.

### Záchyt handshaku

Druhým úkolem je zachycení rámců handshaku pro jejich další zpracování. Tento rámec tedy bude umožňovat spustit zachytávání do externího souboru, vypnutí po úspěšném záchytu a tlačítko pro spuštění a zastavení rozesílání deuthentifikačních rámců. Ty jsou nezbytné pro vyvolání snahy cíle se připojit k síti a opětovně odeslat hansdhake.

### Prolomení hesla

Následujícím segmentem je sada tlačítek pro práci se souborem s handshakem. K jeho prolomení používám slovníkový útok. Bude tedy nutné uživateli umožnit vybrat takový slovník a vyhledat jen jako textový soubor na disku, poté spustit samotné prolamování hesla a po úspěšném prolomení a získání hesla k síti umožnit se k této síti jednoduše připojit stiskem tlačítka. Tím bude tento segment úspěšně završen.

Uživateli též umožním, aby si přímo v aplikaci mol navolit parametry svého slovníku hesel a nechal si jej vygenereovat

### Man-in-the-middle

Tato část je jádrem samotné aplikace. Heslo k síti se dá získat mnoha způsoby a výše popsané kroky jsou jen jedna z možností. Nejdůležitějším úkolem MITM útoku je zajistit, aby útočníkův počítač sloužil jako přemostění mezi obětí a přístupovým bodem sítě a tuto komunikaci zachytávat a dále zpracovávat. Právě v tomto segmentu bude uživateli umožněno spustit ARP poisoning na oběť i přístupový bod, spustit zachytávání komunikace do souboru a povolit přeposílání paktů na svém stroji.

### Záchyt DNS dotazů

Posledním krokem v aplikaci je názorně ukázat, že se útok úspěšně zdařil. Že od počátečního stavu, kdy útočník přichází do oblasti s bezdrátovou sítí, ke které nemá heslo, dospěje až do stavu, kdy může nahlížet do komunikace vybrané oběti. To bude demonstrováno zobrazením URL adres, ke kterým oběť přistupovala, a času tohoto přístupu.

# Implementace

Aplikaci jsem nazval DeMITMo a zjednodušení implementace jsem využil úložiště na serveru github, kde je aplikace dostupná ke stažení na adrese [*https://github.com/JKarmasin/MITM\_demo.git*](https://github.com/JKarmasin/MITM_demo.git). To mi umožnilo jednoduchý přenos mezi několika stanicemi během vývoje a testování. Je implementována v jazyce Python s doinstalovanou knihovnou cusmtomtkinter pro vykleslování grafického uživatelského rozhraní. Tato knihovna vychází z knihovny Tkinter, která bývá standarně součástí balíku instalovaného se samotným Pythonem. Customtinker umožňuje tvorbu moderního, jednoduchého rozhraní nabízí spoustu rozšiřujících funkcionalit jako je přepínání mezi světlým a tmavým vzhledem. To je pro moji aplikaci velice vhodné, protože pro práci na monitoru mnoho uživatelů preferuje tmavý vzhled kvůli únavě očí, ale pro použití na projektoru, kam je aplikace především cílen, je vhodnější světlý motiv.

Doporučovaným přístupem k implementaci aplikací v Pythonu je využívání virtuálních prostředí pro jejich běh (Romano, Krugher, 2021). Vytvoření tohoto prostředí a doinstalování do něj potřebných knihoven se provádí pomocí bashového skriptu setup.sh. Spuštění aplikace je pak možné dělat skriptem start.sh. Detailněji jsou tyto skripty vysvětleny v Příloze 1.

## Informativní splash screen

Po spuštění aplikace se zobrazí informativní splash screen. Na něm je logo aplikace, název, ale především upozornění na právní kontext používání této aplikace. Nelegální odposlech a záznam telekomunikace totiž dochází k porušení čl. 13 Listiny základních práv a svobod. Dále se může použitím této aplikace a snahou o provedení útoku na sít, které není vlastník, ani nemá písemný souhlas vlastníka, uživatel dopustit trestného činu podle § 182 a § 230 Trestního zákona (Kolouch, 2016). Na tyto fakta je uživatel upozorněn po spuštění aplikace předtím, než je možné dělat jakékoliv další činnosti.

Obsah obrázku text, snímek obrazovky, logo, Písmo

Popis byl vytvořen automaticky

Obrázek 7 - Splash screen

*(Zdroj: vlastní)*

## Implementace hlavního okna

Oproti původnímu návrhu jsem opustil myšlenku použití záložek, ale navrhnul jsem jednoduché menu na levé straně obrazovky. Tlačítky v menu se poté volí, co se bude zobrazovat v pravé, hlavní části obrazovky. Po spuštění aplikace je zvýrazněno vybrání prvního tlačítko v menu a ostatní jsou deaktivovány. Tlačítka se postupně zpřístupňují, jak uživatel dokončuje vyžadované akce v jednotlivých segmentech. Už vykonané segmenty jsou znázorněny „fajfkou“ a zeleným textem. Uživatel tedy má stále přehled, v jaké fázi demonstrace se právě nachází. To je znázorněno na následujícím obrázku.

Obsah obrázku text, snímek obrazovky, Písmo, design

Popis byl vytvořen automaticky Obsah obrázku text, snímek obrazovky, Písmo, design

Popis byl vytvořen automaticky Obsah obrázku text, snímek obrazovky, Písmo, logo

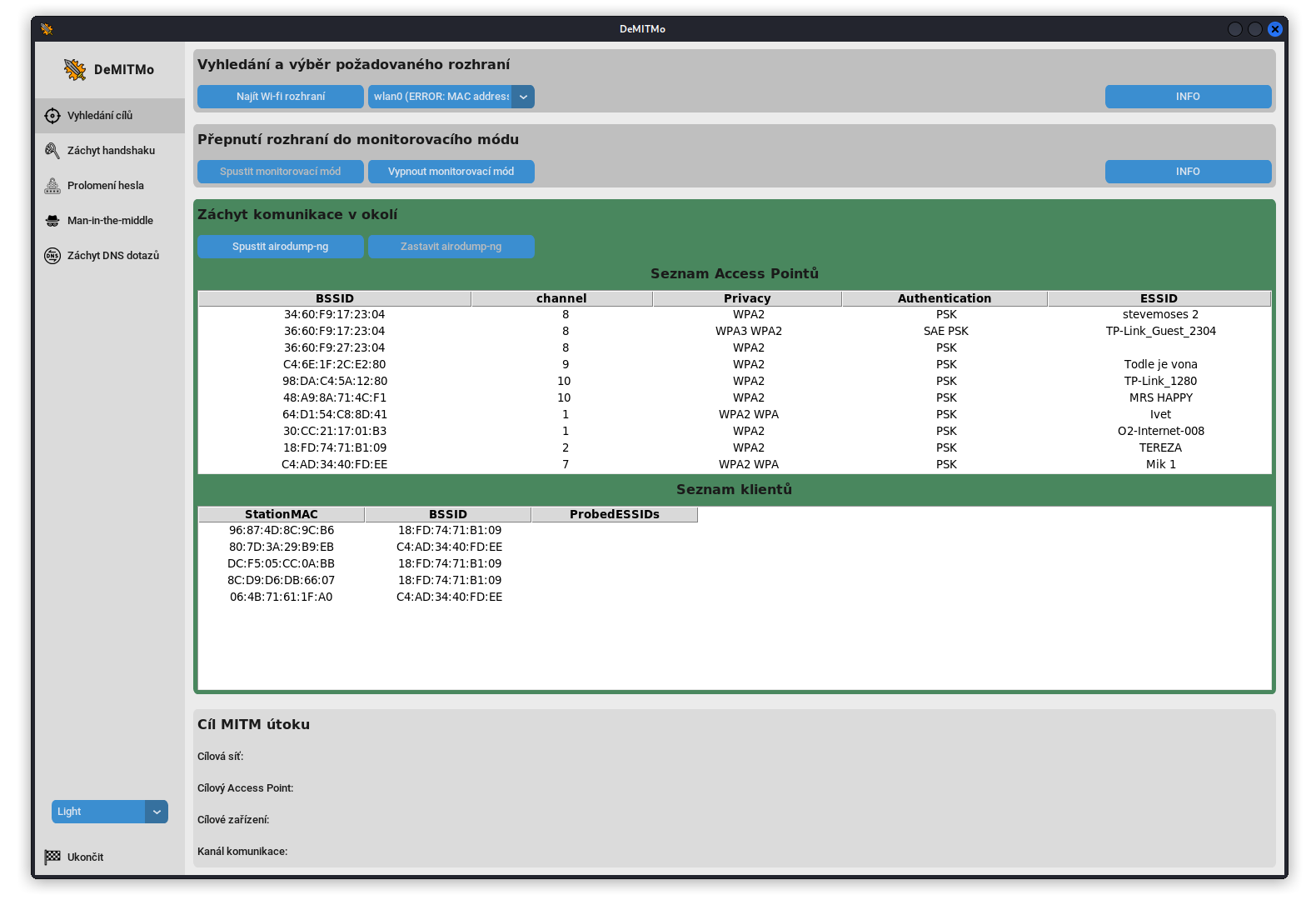
Popis byl vytvořen automaticky

Obrázek 8 - Navigační menu

*(Zdroj: vlastní)*

Na spodní straně navigačního panelu je tlačítko na přepínání světelného motivu aplikace a také tlačítko k ukončení aplikace. Tím se reverzně vrátí všechny vykonané akce do původního stavu.

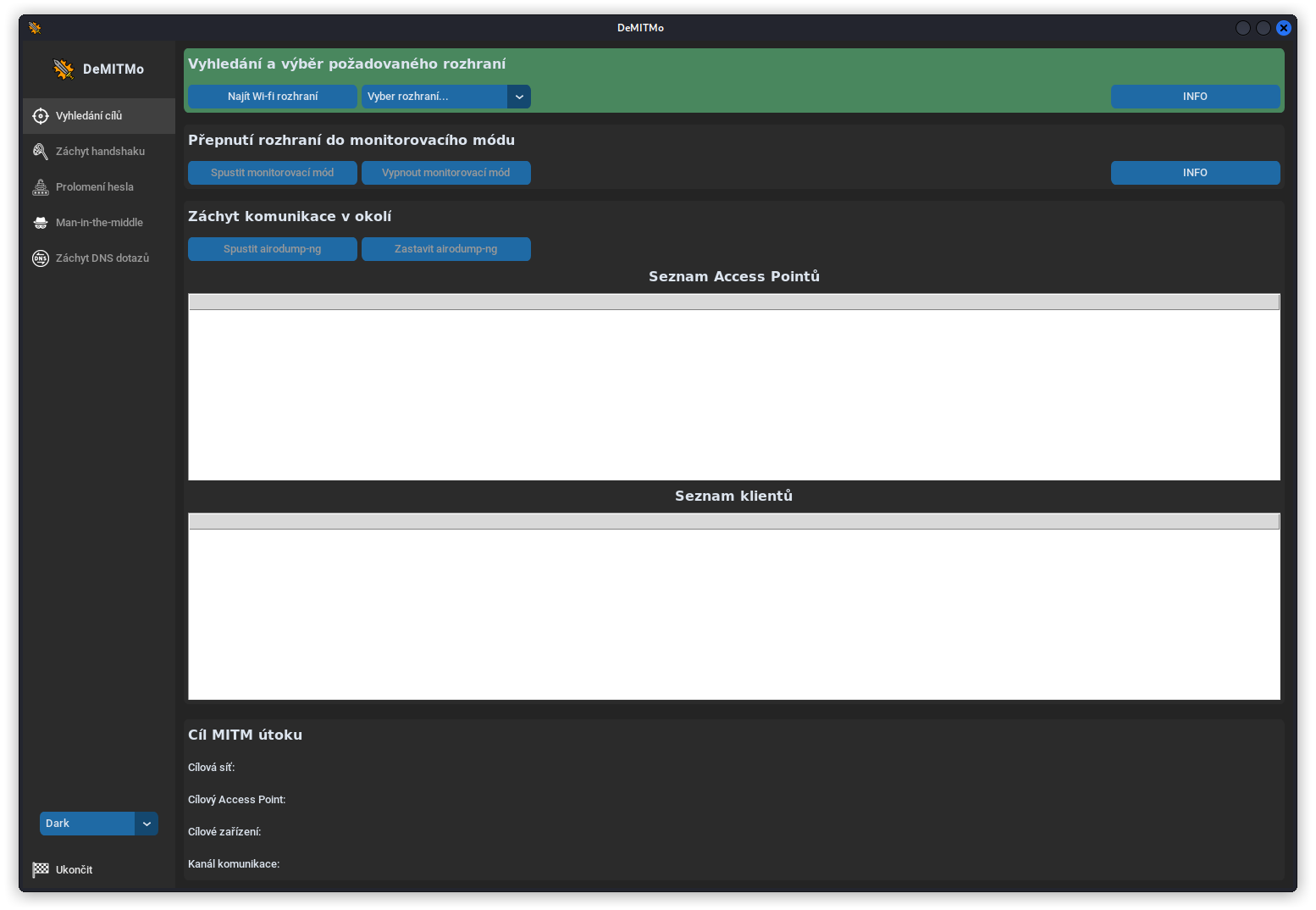
Hlavní část okna obsahuje všechny ovládací prvky k provedení komplexního man-in-the-middle útoku. V segmentu, který obsahuje více dílčích kroků k vykonání, je ten, který má být právě spuštěn zvýrazněn zelenou barvou. Všechny už vykonané akce jsou podbarveny tmavší šedou barvou. Na ukázkovém obrázku níže je zobrazeno celé okno aplikace během provádění skenování bezdrátové komunikace v okolí. Je možné vidět, že výběr použitého bezdrátového rozhraní a jeho přepnutí do monitorovacího módu už byl vykonán. V zeleném panelu uprostřed je vidět seznam všech dostupných sítí v okolí. V tabulce níže je zobrazen seznam viditelných klientů, tedy jejich MAC adres a adres přístupových bodů, ke kterým jsou připojeni. Kliknutím na kterýkoliv záznam v této tabulce se toto zařízení vybere. S ním se také vybere příslušná síť a kanál, na kterém tyto dvě zařízení spolu komunikují. Tím jsou splněny všechny požadované činnosti v prvním segmentu a odemkne se tlačítko v navigačním menu pro následující část.



Obrázek 9 - Hlavní okno aplikace (světlé)

*(Zdroj: vlastní)*

Zde je ukázka aplikace po přepnutí do tmavého motivu. Tento motiv jsem zvolil jako výchozí.



Obrázek 10 - Hlavní okno aplikace (tmavé)

*(Zdroj: vlastní)*

Zdrojové soubory aplikace jsou pro přehlednost rozděleny do několika složek. V kořenové složce aplikace se nachází instalační a spouštěcí bashové skritpy *install\_adapter.sh, setup.ch* a *start.sh*. Při vytváření virtuálůního prostředí se stahují nestandardní balíčky podle seznamu v souboru *requirements.txt*. Nachází se zde také iniciační pythonovský skript *DeMITMO.py*, který vykleslí uvítací splash screen a poté spustí skript pro hlavní okno aplikace.

Tento páteřní skript *mitm.py* se nachází ve složce src. Takždý segment v provádění MITM útoku má svůj vlastní obslužný skript, který se nachází tamtéž. Jsou zde i pomocné skripty pro vykreslování informačních oken s doplňujícími informacemi a soubor *global\_names.py* pro předávání dat mezi jednotlivými segmenty.

Všechna grafika použitá v aplikaci je ve složce *images.* Ta obsahuje dvě podsložky. Jednu pro ikony v navigačním menu a druhou pro grafiku pro pomocná okna. Jsou zde i obrázky použité pro zpřehlednění githubového úložiště. Ty ovšem nejsou pro běh aplikace nijak relevantní.

Za běhu programu je také vytvořena složka tmp, kam se ukládají pracovní soubory. Především se jedná o zachycené datové komunikace.

## Implementace jednotlivých segmentů

V této podkapitole popíšu jednotlivé segmenty a zmíním systémové příkazy, které jsou volány pro běh aplikace.

### Vyhledání cílů

V první části pro vyhledání cílů neboli rekognoskaci, je nejdříve nutné vyhledat dostupné wi-fi adaptéry. Aplikace po spuštění sama zavolá příkaz

*iwconfig*

Ten vypíše informace o bezdrátových adaptérech připojených k počítači. Výstup tohoto příkazu zpracuji a do rolovacího menu zapíšu názvy všech rozhraní i s jejich MAC adresami pro jednodušší identifikaci. Obnovení položek v rolovacím menu je možné například po připojení USB adaptéru až po spuštění programu stiskem tlačítka *Najít Wi-fi rozhraní.*

Po zvolení vhodného adaptéru se aktivuje tlačítko pro zapnutí monitorovacího módu. Ten je samozřejmě možné opět vypnout příslušným tlačítkem. S přichystaným adaptérem pro skenování okolí je možné spustit monitorování tlačítkem volajícím příkaz (pro interface wlan0)

*airmon-ng start wlan0*

Předtím je samozřejmě zavolán příkaz na vypnutí daného rozhraní a ukončení služeb, které by bránily k zapnutí monitorovacího módu. Po úspěšném spuštění toho módu je opět zapnuta služba NetworkManager, která se stará standardní připojení v bezdrátové síti ostatních rozhraní – typicky integrovaného Wi-Fi adaptéru u notebooku.

Nástroj Airmon-ng vypisuje svůj výstup do souboru airdump-01.csv. Po několika okamžicích od spuštění se začne tento soubor zpracovávat pomocí standardní pythonovské knihovny *csv* a relevantní výstupy jsou zobrazovány do tabulek. Je zde jedna tabulka pro bezdrátové sítě a jejich přístupové body (AP) a druhá pro viditelná zařízení. Ty nazývám klienti. Jakmile uživatel uvidí klienta, na kterého chce útok cílit, pomocí tlačítka zastaví monitorování okolí a tohoto klienta v tabulce vybere. Tím je segment dokončen.

### Záchyt handshaku

Cílem tohoto oddílu je zachycení všech čtyřech rámců 4-way handshaku potřebných pro ověření při připojování zařízení k síti. To provedu pouze dvěma kroky. Prvním z nich je opětovné zapnutí monitorování pomocí airmon-ng. Tentokrát ale specifikuji MAC adresu konkrétního AP a frekvenční kanál. Výstup zapíšu do souboru handshaku-01.csv,

*airmon-ng -c5 -d A1:B2:C3:D4:E5:F6 -w tmp/handshaku wlan0*

Tento záchyt nechám běžet na pozadí a souběžně spustím zasílání deautentifikačních rámců na zvolené zařízení oběti. To se provdání pomocí volání aireplay-ng. Tomuto příkazu dám jako arugemty MAC adresu připojeného zařízení, MAC adresu AP sítě, na kterém tyto dvě zařízení komunikují. Že chci zasílat deautentifikační rámce specifikuji přínakem --deauth 0

*aireplay-ng --deauth 0 -D -c A1:B2:C3:D4:E5:F6 -a F4:E5:D6:C7:B8:A9 wlan0*

Tento příkaz efektivně znemožní jakémukoliv zařízení, na které je spuštěn, se připojit k AP a zařízení, které už jsou připojeny, od sítě odpojí. Stačí jej nechat běžet několik sekund a pak jej tlačítkem zastavit. Po několika okamžicích aplikace napíše, že se jí podařilo zachytit všechny potřebné informace a segment je ukončen.

Teoreticky by šel krok aktivního vyvolání odpojení klienta od sítě vynechat a pasivně čekat, až se klient sám přihlásí. To by jistě stálo za zvážení při provádění reálného útoku, kdy je nenápadnost prováděných akcí maximálně žádoucí. V našem demonstračním případě je však vhodné utilitu aireplay-ng použít, protože školeným divákům ukáže další ze zranitelnosti bezdrátových sítí. A to tu, že každé zařízení může být kdykoliv odpojeno pouze vysíláním vhodných rámců do prostoru zařízením se zlým úmyslem (tzv. deauthentification flood attack).

Úspěch zachycení je testován pomocí příkazu tshark a hledáním požadovaného řetězce z posledního čtvrtého rámce handshaku.

*tshark -r handshake-01.cap -n -y „eapol“ | grep „Message 4 of 4“*

Pokud se nepodaří handshaku zachytit, je možné opětovně na pár sekund spustit deautentizaci. Toto lze opakovat až do úspěšného záchytu. Po získání handshaku je nutné zastavit jeho zachytávání příslušným tlačítkem.

Toto je kritická část, kdy si majitel zařízení, na které je útočeno, může všimnout, že se děje něco nestandardního. Zařízení je na okamžik odpojeno od sítě a samo se pokusí připojit zpět. Pokud uživatel aplikace nenechá běžet proces s deautentizací moc dlouho, tak se zařízení úspěšně připojí ještě dřív, než si majitel něčeho všimne. To že je Wi-Fi připojení nestabilní přece jen není nic tak neobvyklého.

### Prolomení hesla

Vstupem do této části útoku je soubor se zachyceným handshakem. Výstupem je pak připojení k bezdrátové síti pomocí hesla z tohoto handshaku. To je ale nejdříve nutné dešifrovat. Dešifrování provádím pomocí utility aircrack-ng. Ta ke svému spuštění ale potřebuje mít specifikovaný textový soubor se slovníkem hesel. Aplikace proto po spuštění tohoto segmentu uživatele vyzývá, ať buďto tlačítkem načte slovník, který je již někde na disku, nebo ať si vytvoří svůj slovník. To dělá utilita crunch, která vyžaduje jako argumenty minimální délku hesla, jeho maximální délku, vypsané znaky, které se mohou v heslech objevovat a název výsledného slovníku. Všechny tyto položky je možné zadat pomocí vstupních polí.

*crunch 4 8 abcd123456789/\*-+ -o slovnik.txt*

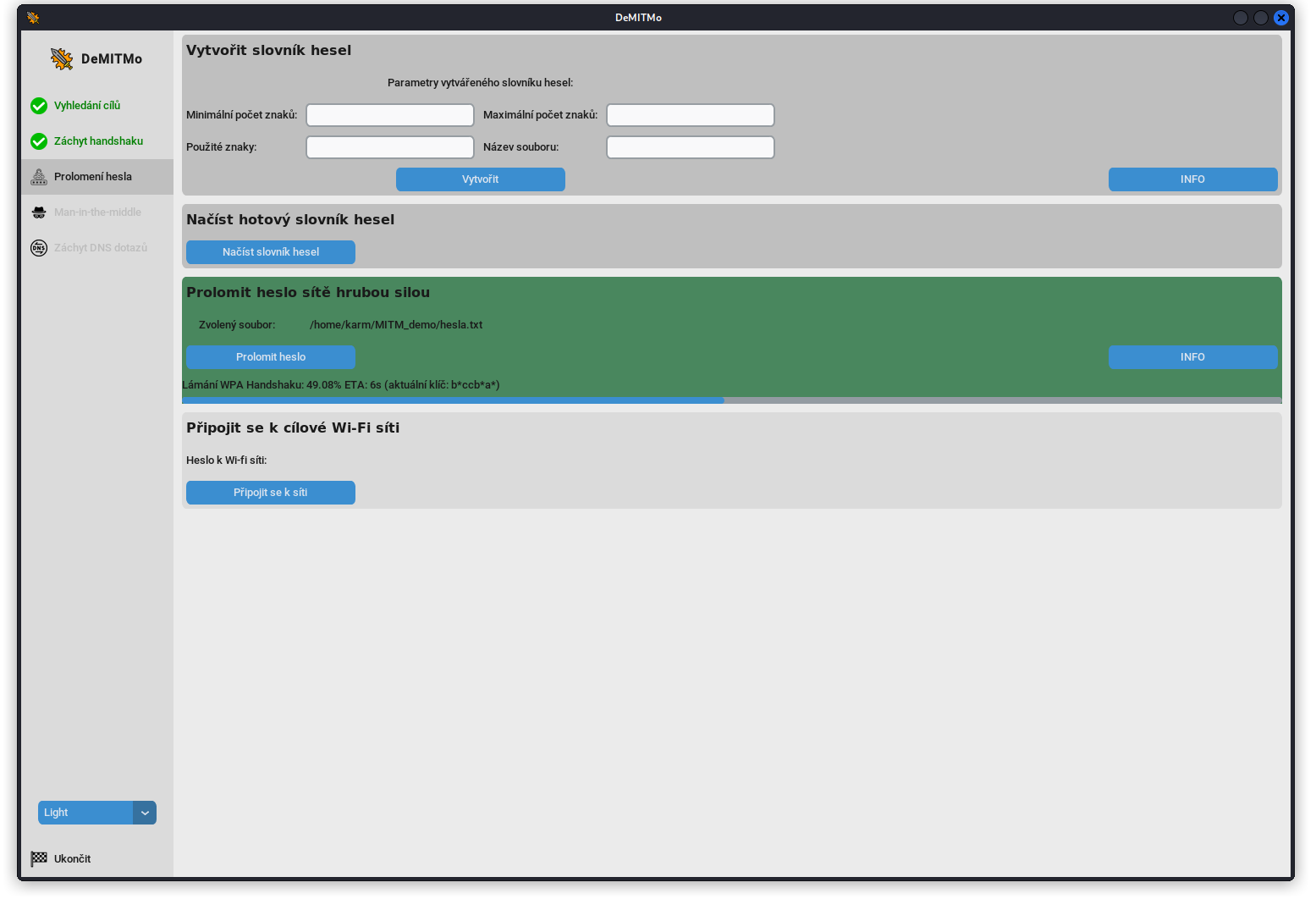
Například tento příkaz by vytvořil slovník.txt, kde by byly hesla o minimální délce čtyři, maximální délce osm znaků a obsahoval by písmena od a do z anebo číslice anebo několik vypsaných speciálních znaků.

Ve zdrojových souborech je už mnou vytvořený slovník hesel hesla.txt. Jeho velikost je nastavena tak, aby během prezentace prolamování hesla trvalo přiměřeně krátce. Je nesmyslné, aby se celé školení na 10 minut zastavilo, aby se počkalo na proces lámání hesla. Do předpřipraveného slovníku tedy prezentující kamkoliv vloží heslo od sítě, na které útok demonstruje a během prezentace jen načte a použije. Na standardním stroji lámání trvá pod 10 sekund.

Po zvolení nového, nebo dopředu vytvořeného slovníku je možné spustit samotné prolamování pomocí příkazu

*aircrack-ng tmp/handshake-01.cap -w hesla.txt -l password.txt*

Případné dešifrované heslo je potom zapsané do textového souboru password.txt. Zde je nutné zmínit, že po demonstrační účely je třeba, aby prolomení bylo úspěšné. Je tedy vhodné dopředu vytvořit slovník, kde je manuálně vepsané heslo k síti, na které útok demonstrujeme, nebo na síti nastavit přístupové heslo takové, které bude v nově vytvářeném slovníku přítomné. Jde přeci o demonstraci, nikoliv univerzální nástroj pro skutečné útoky. Na následujícím screenshotu je vidět aplikace v průběhu prolamování hesla. Je patrné, že aplikace ukazuje přibližný čas dokončení lámání hesla, aktuálně testované heslo a také postupně nabývající načítací lištu.



Obrázek 11 - Lámání hesla

*(Zdroj: vlastní)*

Po úspěšném nalezení hesla ve slovníku je toto heslo ze souboru password.txt přečteno a zobrazeno v posledním poli. To pak uživateli umožní se k síti připojit jedním tlačítkem. To provede volání systémového příkazu nmcli

*nmcli wifi connect HackMe password TajneHeslo123\* ifname wlan0*

### Man-in-the-middle

Po připojení k síti už je samotné aktivování útoku vcelku triviální. Skládá se ze tří kroků. Prvním je nastavení systému tak, aby umožňoval přeposílání paketů. To je nejlepší provádět pomocí příkazu

*sysctl net.ipv4\_forvard=1*

Lze to také provést zápisem „1“ do souboru /proc/sys/net/ipv4/ip\_forward. Volání systémového příkazu mi ale přišlo jako elegantnější řešení.

Druhým krokem je spuštění ARP poisoningu. Po stisknutí příslušného tlačítka se aplikace pokusí oskenovat síť a zjistit IP adresy AP a oběti.

*arp-scan --interface wlan0 --localnet*

To se nemusí zdařit na první pokus. Proto je někdy nutné toto tlačítko stisknout dvakrát až třikrát. Po úspěšném zjištění obou IP adres jsou volány dva procesy. Jen pravidelně posílaný AP informující ho, že MAC adresa příslušející k IP oběti je právě MAC adresa útočníkova adaptéru. Obráceně se to samé posílá i na zařízení oběti.

*arpspoof -i wlan0 -t 192.168.100.120 192.168.100.1*

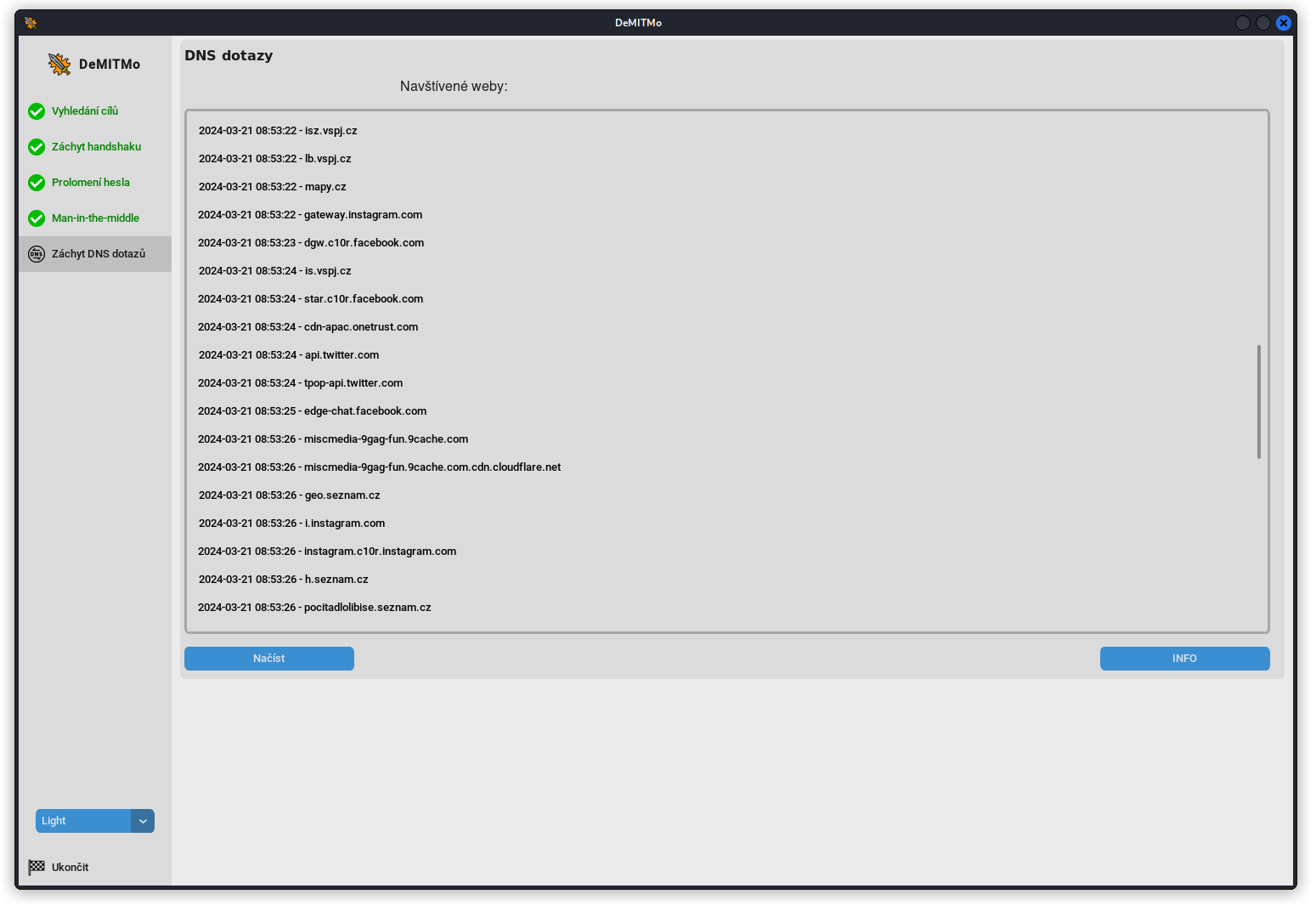
*arpspoof -i wlan0 -t 192.168.100.1 192.168.100.120*

Spuštěním těchto procesů je zapříčiněno, že všechen datový tok mezi AP a obětí je přeposílán skrz zařízení útočníka. Ten tento tok tedy může měnit nebo odposlouchávat. Pro demonstraci jsem je kompletní tok zachytáván do souboru. To provádím pomocí objektu AsyncSniffer z pythonovského balíčku scpy. Výsledný soubor je uložen ve složce tmp pod názvem full\_traffic.pcap.

### Záchyt DNS dotazů

Všechny kroky, které aplikace až doposud dělala by však neznalé obecenstvo nepřesvědčily, že byl proveden jakýkoliv kybernetický útok. Je tedy nutné zachycený datový provoz nějakým způsobem vizualizovat. Rozhodl jsem se, že jednoduše budu z DNS paketů filtrovat URL adresy webů, které se snažila oběť po dobu zachytávání komunikace navštívit. To má výhodu v tom, že dotazy na DNS jsou zpravidla nešifrované a využívají protokolu UDP. K filtraci DNS paketů ze zachyceného souboru opět využiju python knihovny pyshark. Kromě adresy webu no přehledové tabulky vypisuji i čas, kdy byl DNS dotaz na tuto adresu poprvé odeslán. Ukázka je na následujícím obrázku. Je možné z něj vyčíst, že oběť přistupovala mimo jiné na weby vspj.cz, mapy.cz, facebook.com, twitter.com a instagram.com.

Tabulka se nevypisuje automaticky, ale pouze po obnovení stiskem tlačítka Načíst.



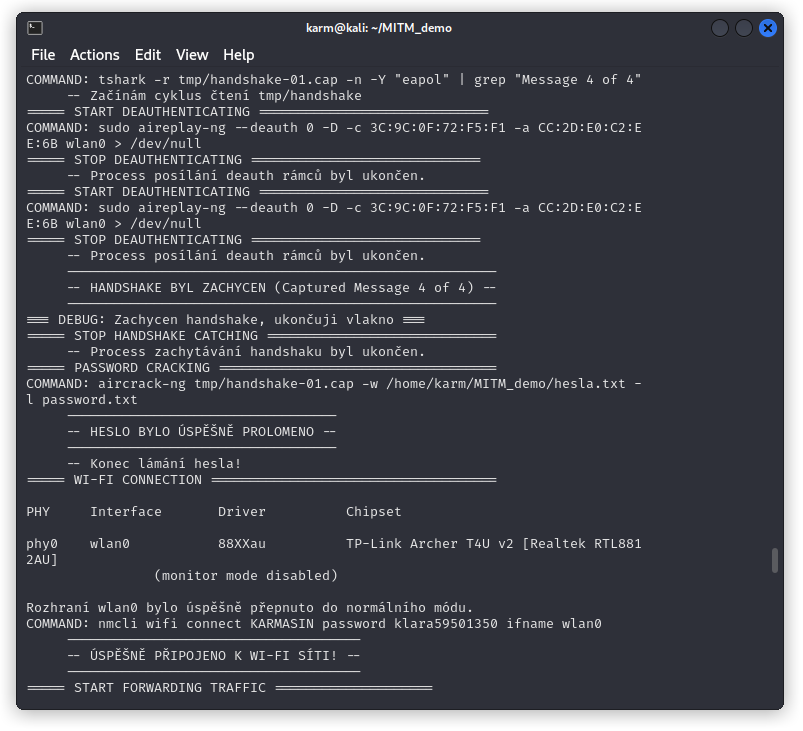
Obrázek 12 - Zachycení DNS dotazů

*(Zdroj: vlastní)*

### Konzolový výstup

Tím, že je grafické rozhraní aplikace spouštěno přes příkazovou řádku skriptem, je tato konzole po celý čas běhu programu spuštěna. Toho jsem využil nejen k výpisu ladicích řetězců, ale především k průběžnému vypisování volaných příkazů a informování přednášejícího o stavu spouštěných funkcí a k výpisu chybových hlášek. Uživatel tak v jednom okně graficky vidí, co konkrétní tlačítko dělá ve vztahu k celé proceduře MITM útoku, v druhém pak vidí, jak tato akce byla vyvolaná.

Tento výstup je ale pouze informativní a pro neznalého sledujícího bude spíše matoucím. Ideálním způsobem prezentace s využitím této aplikace tedy vidím konfiguraci, kdy je něco jiného zobrazováno na projektoru a něco jiného na obrazovce, kterou vidí pouze přednášející.

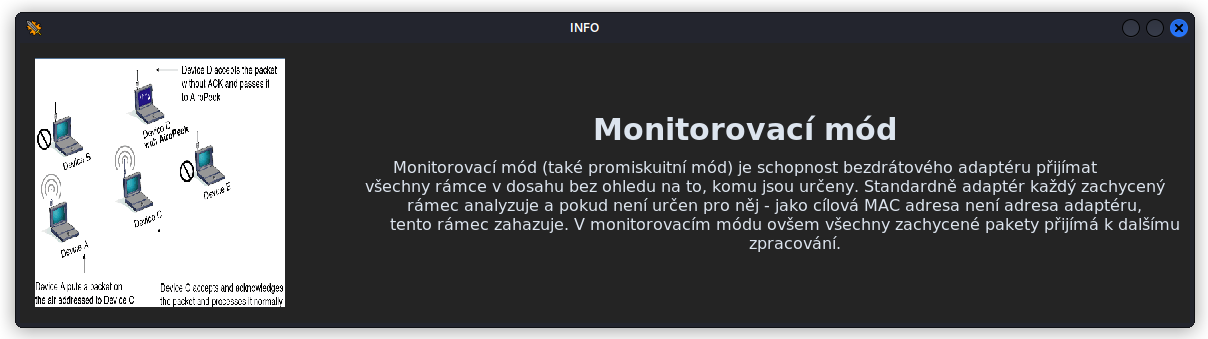


Obrázek 13 - Konzolový výstup

*(Zdroj: vlastní)*

### Informační okna

U většiny kroků na všech stránkách je v pravé straně rámce tlačítko INFO. To otevře nové okno, kde jsou k danému kroku doplňující informace a grafika. Tím je dosažena přehlednost aplikace, kdy není v jeden čas na obrazovce zbytečně velké množství informací. O doplňující informace v nově otevřeném okně se také může opřít přednášející, který není 100% znalý problematiky MITM útoků a doprovodný text mu usnadní při výkladu neopomenout důležité informace.



Obrázek 14 - Informační okno

*(Zdroj: vlastní)*

## Wi-fi adaptér s monitorovacím módem

Kriticky důležitým aspektem při používání tohoto nástroje je vlastnit Wi-Fi adaptér, který je schopný se přepnout do monitorovacího módu. Bez tohoto adaptéru není už druhý krok v celé posloupnosti proveditelný a celá aplikace by tedy byla nepoužitelná.

Monitorovací (také promiskuitní) mód je schopnost adaptéru zachytávat všechny přijaté pakety bez ohledu na to, komu jsou určeny. TODO

# Protiopatření

## ARP spoofing

## DNSsec

apod.

# Testování

Aplikace byla po implementaci otestována. Byla testována přenositelnost mezi jednotlivými systémy, funkčnost a limity implementace.

## Instalace

Aplikace byla vyvíjena na OS Kali linux nainstalovaným přímo na HW.

Testování na virtuálním stroji:

* Hyper-V nepoužitelný, protože nepodporuje přímé připojení USB do VM
* VMware Player - v pořádku

Instalace vhodných ovladačů z github.com/aircrack-ng pomocí:

./install\_driver

Vytvoření virtuálního python prostředí a stáhnutí potřebných balíků pomocí:

./setup

spuštění:

./start

ipconfig /flushdns

## Testy funkčnosti

bugy u MAC2IP

## Testy výkonosti

rychlost stahování

### Velikost slovníku hesel

Limitujícím faktorem může také být při neopatrné volbě parametrů vytvářeného slovníku jeho výsledná velikost

# Závěr

# Seznam použité literatury

BALOCH, Rafay. Ethical Hacking and P/enetration Testing Guide. USA: CRC Press, 2015. ISBN: 78-1-4822-3162-5

FREIER, A, KARLTON, P. *The Secure Sockets Layer (SSL) Protocol Version 3.0*. 2011. <https://datatracker.ietf.org/doc/html/rfc6101>

HAIDER, Zeeshan. *4-Way Handshake* [online]. 2019 [cit. 2023-12-16]. Dostupné z: <https://www.wifi-professionals.com/2019/01/4-way-handshake>

IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, in IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), 14 Dec. 2016, doi: 10.1109/IEEESTD.2016.7786995.

INTERNETSOCIETY.ORG. *Encrypted DNS Facksheet* [online]. 2023 [cit. 2023-12-17]. Dostupné z: <https://www.internetsociety.org/resources/doc/2023/fact-sheet-encrypted-dns/>

KOLOUCH, Jan. *CYBERCRIME.* Praha: CZ.NIC, 2016. ISBN: 978-80-88168-15-7

MARUŠIC, Marek. Automatizace MitM útoku pro dešifrování SSL/TLS. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Pluskal Jan.

RAYMOND, Serena. *What is the DNS Layer and How Do I Secure It?* [online]. 2022 [cit. 2023-11-22]. Dostupné z: <https://www.dnsfilter.com/blog/dns-layer-how-to-secure>

ROMANO, Fabrizio, KRUGER, Heinrich. *Leatn Python Programming. An in-depth introduction to the fundamentals of Python. Third edition*. Birmingham, UK: Packt Publishing, 2021. ISBN 978-1-80181-509-3

SHARMA, Himanshu. *Kali Linux – An Ethical Hacker’s Coockbook*. Birmingham, UK: Packt Publishing, 2017. ISBN 978-1-78712-182-9

Wu, Wentz. *What OSI Layer does TLS Operate and Why?* 2020. <https://wentzwu.com/2020/08/21/what-osi-layer-does-tls-operate-and-why/>

# Přílohy

Příloha 1 - uživatelský manuál (pro přednášejícího, jak správně nastavit všechno a na co si dát pozor… např. zjistit si MAC adresy cílených zařízení předem, ipconfig /dnsflush apod.

Příloha 2 - poster pro posluchače