VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020 Bc. Michal Řezáč

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

HONEYPOT PRO RODINU BEZDRÁTOVÝCH KOMUNIKAČNÍCH PROTOKOLŮ IEEE 802.11
HONEYPOT FOR WIRELESS COMMUNICATION PROTOCOLS OF IEEE 802.11 FAMILY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. Michal Řezáč

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Radek Fujdiak, Ph.D.

BRNO 2020

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**

Ústav telekomunikací

Student: Bc. Michal Řezáč ***ID:*** 186571

Ročník: 2 ***Akademický rok:*** 2019/20

NÁZEV TÉMATU:

Honeypot pro rodinu bezdrátových komunikačních protokolů IEEE 802.11

POKYNY PRO VYPRACOVÁNÍ:

Student provede analýzu možností, na jejímž základě vybere vhodnou platformu (např. Raspberry 3B+, vhodně zvolené softwarově definované rádio či bezdrátový směrovač s alternativním firmwarem) pro vytvoření funkčního honeypotu pro rodinu protokolů IEEE 802.11. Důležitým prvkem bude podpora všech hlavních protokolů, funkčnost min. v rámci dvou základních pásem (2,4/5 GHz), různých módů (SISO, MIMO, MU-MIMO), apod. Po zprovoznění honeypotu budou simulovány bezpečnostní incidenty a vyhodnocena efektivita navrženého řešení. Následně proběhne optimalizace a nasazení v reálném provozu. Umožněna bude volba nízké i vysoké míry interakce (a tedy nutnost dosažení vysoké bezpečnosti konečného řešení a ochrany vůči zneužití). Výsledkem diplomové práce bude finální honeypot pro rodinu protokolů IEEE 802.11 určený do reálného provozu.

DOPORUČENÁ LITERATURA:

[1] SPITZNER, Lance. Honeypots: tracking hackers. Reading: Addison-Wesley, 2003.

[2] GAST, Matthew. 802.11 wireless networks: the definitive guide. " O'Reilly Media, Inc.", 2005.

Termín zadání: 3.2.2020 ***Termín odevzdání:*** 1.6.2020

Vedoucí práce: Ing. Radek Fujdiak, Ph.D.

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRAKT

Diplomová práce se zabývá realizaci WiFi Honeypotu, provádějícího detekci síťových útoků z rádiového prostředí, využívající sadu protokolů IEEE 802.11. Byla vytvořena specifická konfigurace na základní desce formátu mITX, obsahující skripty a programovou výbavu pro sběr, zpracování a vyhodnocení dat. Na základě informací a poznatků o konkrétních síťových útocích lze identifikovat datový provoz vedoucí k anomáliím a detekci případného síťového útoku. Finální zařízení bylo otestováno v reálném prostředí pro dlouhodobý sběr dat a vyhodnocení síťové aktivity v dané lokalitě. Tím je splněn hlavní cíl této práce. Tedy realizace WiFi Honeypotu s podporou protokolů IEEE 802.11 a s možným nasazením v reálném prostředí.

KLÍČOVÁ SLOVA

Honeypot, WiFi Honeypot, IEEE 802.11

ABSTRACT

Objective of this master thesis solves possible way of WiFi Honeypot realisation, which is constructed to detecet malicious network activity and attacks in radio environment that uses a set of IEEE 802.11 protocols. A specific configuration was created on the mITX format motherboard and contains scripts and software for data collection, analysis and its evaluation. Based on information and knowledge about specific network attacks it is possible to identify data traffic leading to anomalies and detect possible network attack. The final device was tested in real use for long-term data collection and evaluation of network activity in the given location. This fulfills the main goal of this work, which is implementation of WiFi Honeypot with support for IEEE 802.11 protocols and with possible deployment for real use.

KEYWORDS

Honeypot, WiFi Honeypot, IEEE 802.11

ŘEZÁČ, Michal. *Honeypot pro rodinu protokolů IEEE 802.11*. Brno, 2020, 91 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Radek Fujdiak, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Honeypot pro rodinu protokolů IEEE 802.11“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce, panu Ing. Radku Fujdiakovi, Ph.D., za odborné, podnětné vedení a konzultace. Stejně tak i rodině a mým blízkým přátelům za jejich podporu, která nemalou měrou přispěla při tvorbě této práce.

Brno ....

podpis autora

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů

Úvod

Uživatelsky velmi oblíbené WiFi sítě, dnes rozšířené v každé domácnosti i podnikovém prostředí, jsou denně využívány k přenosu obrovského množství dat. Navíc jsou v dnešní době i tyto jednoduše kompromitovatelné sítě, založené na komunikaci s protokolem IEEE 802.11, využívány k přenosu citlivých a jednoduše odcizitelných dat. Pro detekci potencionálních útoků v rádiovém prostředí WiFi sítí a jejich případnému odklonění, bude v rámci této diplomové práce vytvořen WiFi honeypot. Diplomová práce se v několika kapitolách zabývá popisem vybraných teoretických celků, analýzou zranitelností IEEE 802.11, výběrem vhodných komponent, programového vybavení a realizací honeypotu samotného.

Nejdříve budou popsány vlastnosti IEEE 802.11, spojené se zadáním práce, respektive s požadavky na finální zařízení společně s popisem zabezpečovacích protokolů v běžných WiFi sítích. Následně bude pokryta teorie honeypotů obecně i s jejich rozdělením podle určení těchto zařízení. Posléze budou popsány možnosti samotného honeypotu fungujícího v rádiovém prostředí. Práce se bude následně zabývat analýzou možných zranitelností v rádiovém prostředí IEEE 802.11 a stanoví se možné vektory útoků a metody k jejich detekci, stejně tak i podpůrný software pro vyvolání těchto útoků v rámci penetračního testování, spolu s testováním fungování finálního zařízení.

Poté se práce zaměří na popis návrhu vlastního honeypotu, během kterého jsou porovnány mikrokontroléry, jednoúčelové WiFi směrovače a mITX/mATX základní desky. Na to naváže výběr a teoretický popis vhodných operačních systémů, spolu se softwarovou podporou a následným doplněním potřebných funkcí do navrhovaného zařízení. V následující kapitole budou experimentálně otestovány vybrané operační systémy a z nich nejvhodnější použit do stálé konfigurace. S výběrem systémů proběhne i prověření podpory a fungování jednotlivých komponent vybraného a sestaveného zařízení.

Práce následně popíše stálou konfiguraci honeypotu, zahrnující finální architekturu zařízení, s rozložením funkcionalit a způsobu realizace zabezpečení tohoto zařízení. Dále samotné skripty, které vytvářejí stěžejní funkce. Kapitola bude zakončena popisem fungování různé interakce realizovaného zařízení, spolu s optimalizací chodu honeypotu.

Závěr práce bude věnován nasazení zařízení do reálného provozu. Zde budou uvedeny průběhy zaznamenávání jednotlivých útoků, s tím spojené zatížení na výpočetní a úložné zdroje. Dále jsou zde popsány průběhy práce honeypotu, spojené s optimalizací chodu zařízení.

1 IEEE 802.11

Při realizaci honeypotu, přesněji WiFi (anglicky v celém názvu známe jako Wireless Fidelity) honeypotu, bude využita rodina protokolů IEEE 802.11., jako hlavní zprostředkovatel uživatelské datové komunikace i možných síťových útoků. Ty budou posléze zaznamenávány a rádiové prostřední, využívající ve volném frekvenčním pásmu 2,4 GHz a 5 GHz technologií WiFi, dále monitorováno.

Skupina standardů pod označením IEEE 802.11 je sadou protokolů fungující na vrstvě fyzické a podvrstvě Medium Access Control (MAC) – vrstvy linkové referenčního modelu ISO/OSI (International Standards Organization / Open Systen Interconnection). Nyní jde o nejrozšířenější protokolovou sadu využívanou k rádiové komunikaci v lokálních sítích (LAN), respektive bezdrátových sítích LAN (WLAN). Původní standard IEEE 802.11, od kterého se dále odvíjí jeho následníci, byl vydán roku 1997. Stále jsou postupně uvolňovány další verze, které umožňují i mimo jiné využití rozličných frekvencí, šířky přenosového pásma, modulace, a s tím související přenosové kapacity. Standardy a jejich základní přenosové parametry jsou uvedeny v tabulce 1.1. Pro standard IEEE 802.11ac jsou přenosové rychlosti 1,3 Gbit/s pro generaci Wave 1 (20, 40 a 80 MHz kanál) a 3,47 Gbit/s pro Wave 2 s šířkou kanálu 160 MHz, respektive 80 + 80 MHz [1].

Standard 802.11	Generace	Frekvenční pásmo [GHz]	Teoretická rychlost	Modulace až
-	-	2,4	2 Mbit/s	DQPSK
a	WiFi 1	5	54 Mbit/s	64 QAM
b	WiFi 2	2,4	11 Mbit/s	DQPSK(CCK)
g	WiFi 3	2,4	54 Mbit/s	64 QAM
n	WiFi 4	2,4/5	600 Mbit/s	64 QAM
ac	WiFi 5	5	1,3/3,47 Gbit/s	256 QAM
ad	-	60	6,76 Gbit/s	64 QAM
ax	WiFi 6	2,4/5/6	10,53 Gbit/s	1024 QAM

Tab. 1.1: Seznam se základními vlastnostmi protokol 802.11 [1], [2].

Sada protokolů IEEE 802.11 je součást nadřazené skupiny IEEE 802 zabývající se datovou komunikací a sítěmi LAN spolu se sítěmi metropolitními (MAN). Jsou mezinárodně uznávané a kompatibilní s většinovým množstvím síťových zařízení. Za tvorbou 802.11 stojí 11. pracovní skupina IEEE (Institute of Electrical and Electronics Engineers), česky Institut pro elektrotechnické a elektronické inženýrství [3].

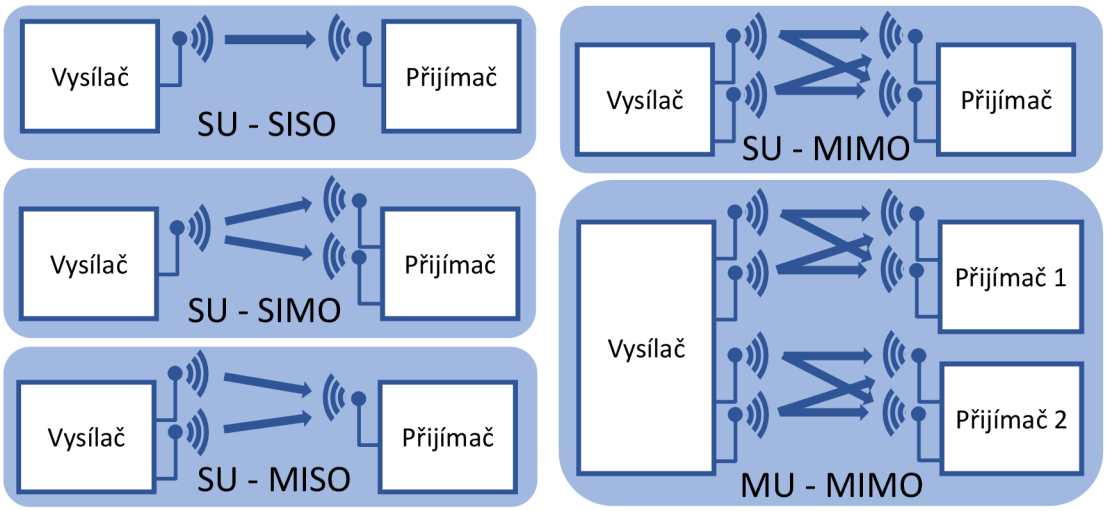
1.1 Obsluha účastníků komunikace

Během datové komunikace je využito techniky Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), technika zabraňující kolizi při komunikaci s vícenásobným přístupem na jedné nosné frekvenci. Před začátkem datového přenosu vysílající stanice naslouchá, zda je přenosové médium volné, posléze iniciuje komunikaci.

Podle hardwarové konfigurace vysílače a přijímače, respektive počtu antén vysílače a přijímače, jsou pak data přenášena, jak je zobrazeno na obrázku 1.1 pomocí anténních systémů:

- Singe User - Single Input Singe Output (SU-SISO),
- Singe User - Single Input Multiple Output (SU-SIMO),
- Singe User - Multiple Input Singe Output (SU-MISO),
- Singe User - Multiple Input Multiple Output (SU-MIMO),
- Multiple User - Multiple Input Multiple Output (MU-MIMO).

Při využití technik Multiple User - Multiple Input Multiple Output (MU-MIMO) je vysílač schopen komunikovat současně s více klientskými účastníky. Snižuje se tak doba potřebná k odbavení všech klientských zařízení a zvyšuje se datová propustnost celého systému [1].



Obr. 1.1: Rozložení antén vysílače a přijímače při různých konfiguracích přenosu dat

Faktorem se zásadním dopadem na množství transportovaných dat je použitá modulace. Podle typu modulace je volen kompromis mezi robustností datového přenosu, tedy její odolnosti proti rušení jak rádiovému i schopnosti průniku překážkami v čitelném stavu pro přijímací zařízení a přenášenými daty. Podle úrovně Recieved Signal Strength Indicator (RSSI) využívá protokol IEEE 802.11 modulaci Binárně fázového klíčování (BPSK) při nejnižším RSSI. Se zlepšujícími se parametry rádiové spojení dochází k použití Kvadraturního fázového klíčování (QPSK) a v ideálním případě některou z variant Kvadraturní amplitudové modulace (QAM) v provedení 16-QAM až po 256-QAM. V generaci WiFi 6 až teoreticky 1024 QAM. Ve zjednodušené tabulce 1.2 je znázorněna souvislost používaných modulací, modelově pro 802.11n, s parametry rádiové linky a šířkou pásma. Použitá šířka pásma může podle jednotlivého standardu nabývat 20, 40, 80, až 160 MHz.

Modulace	Kódovací poměř	Bitů na symbol	20 MHz		40 MHz	
			SNR(dBm)	RSSI	SNR(dBm)	RSSI
BPSK	1/2	1	2	-82	5	-79
QPSK	3/4	2	5	-79	8	-76
QPSK	3/4	2	9	-77	12	-74
16-QAM	1/2	4	11	-74	14	-71
16-QAM	3/4	4	15	-70	18	-67
64-QAM	2/3	6	18	-66	21	-63
64-QAM	3/4	6	20	-65	23	-62
64-QAM	5/6	6	25	-64	28	-61

Tab. 1.2: Zjednodušené modulační a kódové schéma (MCS) 802.11n [4].

1.2 Zabezpečení

Pro běžného uživatele začíná a končí zabezpečení bezdrátové sítě vytvořením přihlašovacího hesla u přístupového bodu (AP). Jde o prakticky jediné zabezpečení, které chrání uživatelská data od jejich kompromitování. Níže jsou vypsány jednotlivé generace šifrovacích protokolů spolu s jejich často překonanými bezpečnostními vlastnostmi.

Šifrovací protokoly

S postupným rozmachem a oblíbeností domácích WiFi sítí, spolu s protokoly IEEE 802.11, jsou vyvíjeny zabezpečovací protokoly. Ty cílí na poskytnutí zabezpečeného

datového přenosu rádiovým prostředím. Následující text se věnuje popisu jednotlivých šifrovacích protokolů využívaných ve WiFi sítích.

Open system authentication

Varianta známá pod českým názvem Systém otevřené autentizace, je výchozí metodou využívanou v bezdrátových sítích. Bez nutnosti zadávání hesla, či ověřovacích klíčů při připojení k WiFi síti nese úskalí, kdy se mezi sebou mohou jednotliví připojení klienti odposlouchávat, případně podvrhovat komunikaci. Proto byly postupně uvolňovány šifrovací protokoly popsané níže.

Wired Equivalent Private - WEP

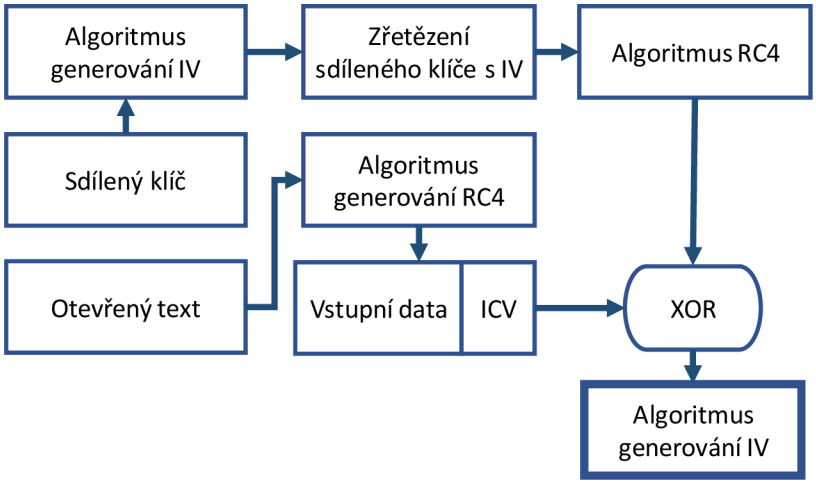
Podle překladu názvu ekvivalent soukromí po kabelu je protokol WEP symbolem prvotního zabezpečení bezdrátových sítí. Ty doposud šlo odposlouchávat kdekoliv v dosahu rádiového signálu. WEP bylo schváleno a následně uvedeno roku 1997. Za cíl kladlo poskytnutí obdobného zabezpečení a soukromí, které je u pevných, kabelových přípojek, jak avizuje název protokolu. Díky nedostatečné síle šifrování, zahrnující mimo další procesy popsané níže i logickou operací XOR, zobrazenou v rovnici 1.1 bylo WEP dále rozvinuto o nadstavby WEPplus, WEP2. Tyto implementace sice mazaly největší chyby a nedostatky WEP, stejně však byly nahrazeny protokolem WiFi Protected Access (WPA) [5], [6].

$$01011010101 \oplus 10111110000 = 11100100101 \tag{1.1}$$

Šifrovací proces WEP používá proudovou šifru 4 (RC4). Ta má za cíl zabezpečit přenášený datový tok, jehož integritu ověřuje Cyclic Redundancy Code (CRC 32) kontrolní součet. Následná výstupní hodnota CRC 32 je umístěná v Integrity Check Value (ICV). Dále je využito inicializačního vektoru IV a tajného klíče, ten může nabývat jedné ze čtyř definovaných klíčů. IV a tajný klíč je použit během procesu inicializace pole RC4. Nad získanými daty z proudové šifry RC4 a ICV je provedena logická operace XOR, tím je šifrování dokončeno. Šifrovaný text se odesílá spolu s IV. Celý proces šifrovaní WEP je znázorněn na obrázku 1.2 [7], [8].

Za největší nedostatky je považováno především [7], [9]:

- nedostatečný algoritmus RC4,
- rychle se opakující IV,
- použití stejného algoritmu pro šifrování dat i autentizaci uživatele,
- omezené varianty statického klíče,
- nedostatečná dálka šifrovacího klíče,
- použití kryptograficky nedostatečného CRC.



Obr. 1.2: Proces šifrování dat protokolem WEP [10].

WiFi Protected Access - WPA

WiFi Protected Access (WPA) je dalším šifrovacím protokolem. Po odhalení bezpčenostních mezer WEP bylo nezbytné zvýšit bezpečnost sítí 802.11. Protokol WPA, původně plánovaný pro plnění standardu IEEE 802.11i, který stále nebyl dokončen, tvořil následovníka WEP. Do značné míry omezuje jeho nedostatky. Stejně jako předchozí protokol využívá i WPA proudové šifry RC4. Došlo zde ke změně délek IV na 48 bitů a tajného klíče na délku 128 bitů [11].

Důležitá změna u WPA nastává v použití Temporal Key Integrity Protocol (TKIP). TKIP prodlužuje IV a umožňuje jim dynamickou proměnlivost. Mimo to používaný mechanismus 802.1x nabízí využití autentizačního serveru a autentizaci na portech. Při využití autentizace na portech je zamítnut datový tok na definovaných portech a povolen až po úspěšné autentizaci klienta [12].

WPA využívá techniky Message Integrity Code (MIC), převzaté z Message Authentication Code (MAC) tvořící kontrolní součet algoritmem známým jako Michael. Pracuje s nešifrovaným textem, MAC adresy zdroje, cíle a tajný klíč. Ten zpracuje jednocestným hashem na výstupní 32 bitový hash. MIC nahradilo kontrolní součet CRC 32. WPA lze shrnout v těchto bodech [13]:

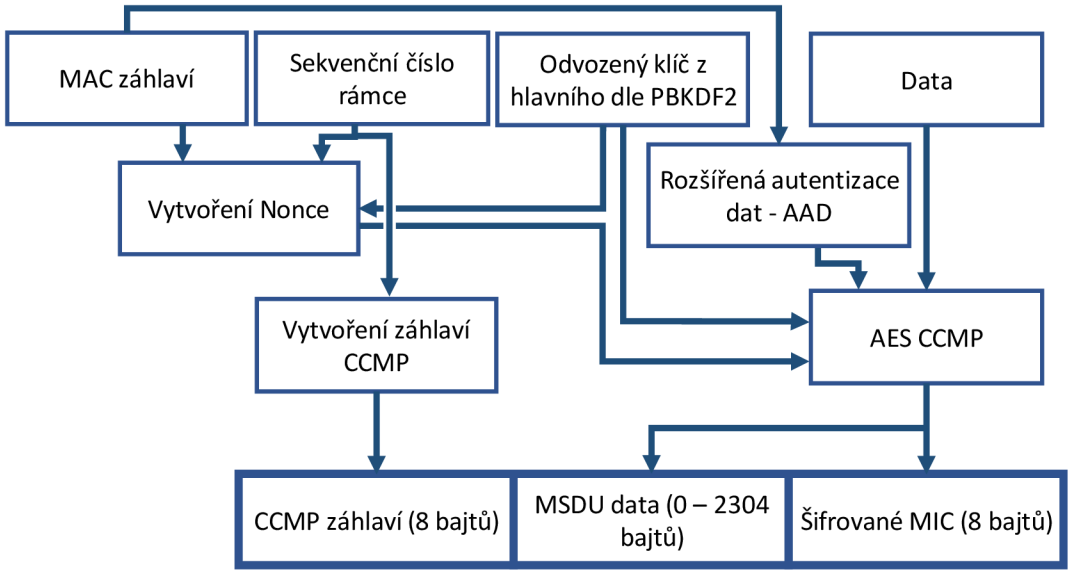
- prodloužení IV a tajného klíče oproti WEP,
- použití TKIP s dynamickými IV,
- možnost použití autentizačního serveru,
- zranitelnost TKIP injekcí paketů.

WPA2

Druhá generace WPA již splňuje kritéria bezpečnostního standardu 802.11i. Ve srovnání s předchůdcem používá několik nových mechanismů. Zahrnující 4-Way Handshake a nový šifrovací algoritmus Counter Cipher Mode with Block Chaining Message Authentication Code Protocol (CCMP) realizovaný pomocí Advanced Encryption Standard(AES) nahradil algoritmus Digital Encryption Standard (DES). CCMP využívá režimu čítače s Block Chain Message Authentication Code (CBC-MAC).

AES je symetrickou šifrou o definované délce 128 bitů. Mimo tuto délku podporuje i velikosti 192 a 256 bitů. Během 10 cyklů se provádí šifrovací operace, zahrnující bajtovou substituci, přesun řádů a sloupců a přidání rozšířeného klíče [13].

Samotný CCMP zabezpečuje hlavičku Medium Access Control Data Unit (MPDU) u 802.11. Využívá polí rozšířeného inicializačního vektoru (Ext IV), ID klíče a číslování paketu. Číslování paketů se postupně inkrementuje, společně s Ext IV a ID klíče zabírá 8 oktetů a zabezpečuje šifrovací proces MIC i datové jednotky. Funkční bloky používané při šifrovaní WPA2 jsou zobrazeny na obrázku 1.3 [14].



Obr. 1.3: Proces šifrování dat protokolem WPA2 [14].

WPA3

V průběhu roku 2018 WiFi Alliance představila právě certifikovaný protokol WPA3. Tento následovník má navázat na vcelku úspěšný WPA2. WPA3 má za cíl pomoci

běžným uživatelům co nejlépe ochránit jejich WiFi síť s vynaložením nejmenšího úsilí. Druhým podstatným bodem je poskytnutí podpory co nejširšímu spektru síťových prvků. Již nyní se přední společnosti v oblasti bezdrátových čipů a zařízení pro 802.11, jmenovitě například Cisco, Qualcomm nebo Intel připravují na přijetí tohoto protokolu. Díky robustnímu zabezpečení pro citlivá data je očekáváno, že se WPA3 stane průmyslovým standardem, právě díky očekávané kryptografické síle.

WPA3 se stejně jako předchozí generace WPA a WPA2 zaměřuje na dvě základní oblasti. První zamýšlené nasazení je běžná spotřebitelská verze WPA3 - Personal. Varianta založená na často velmi slabých uživatelských heslech používá techniku Simultaneous Authentication of Equals (SAE), ta provádí operace password authenticated key exchange (PAKE). Ta ochrání uživatele před offline slovníkovými útoky, což lze uvážit jako značné zlepšení ve srovnání WPA2-Pre Shared Key (WPA2-PSK). I technika SAE (známá i pod názvem Dragonfly) má však svá již odhalené zranitelnosti, nazývané Dragonblood.

WPA3 - Enterprise je určena do firemního prostředí. Kryptografická síla 192 bitů je považována za dostatečně bezpečnou pro využívání mezi firemními a veřejně přístupnými sítěmi [15], [16].

Shrnutí WPA3:

- SAE proti offline slovníkovým útokům,
- pokračování v trendu zvyšování robustnosti šifrování,
- jednodušší nasazení v průmyslu a Machine to Machine (M2M) komunikaci,
- certifikace pro jednotlivé modely čipů s podporou WPA3,
- již zjištěny zranitelnosti pod skupinovým označením Dragonblood.

Zabezpečení sítí 802.11 je v dnešní době nejčastěji realizováno pomocí WPA2, běžně s využitím PSK. Varianta RADIUS je využívána spíše v podnikových sítích za odbornější konfigurace a správy. WPA2 má stejně jak jeho předchůdci WEP a WPA své slabiny a nedostatky. Proto WiFi Alliance pracuje na WPA3, ta je již certifikována na prvních zařízení. Nabízí robustnější šifrování a obecně zamezení možným útokům, které zneužívaly některých slabin u jeho předchůdců. Shrnutí vybraných parametrů bezpečnostních protokolů pro rádiovou komunikaci v 802.11 je v tabulce 1.3.

Protokol	Šifrování	Integrita dat	Autentizace	Šifrovací klíč
WEP	RC4	CRC-32	Open/Shared	40/104 bitů
WPA	TKIP	MIC	PSK/Enterprise	128 bitů
WPA2	CCMP	CBC-MAC	Personal/Enterprise	128 bitů
WPA3	GCMP-256	BIP-GMAC-256	Personal/Enterprise	256 bitů

Tab. 1.3: Porovnání bezpečnostních protokolů pro 802.11 [17].

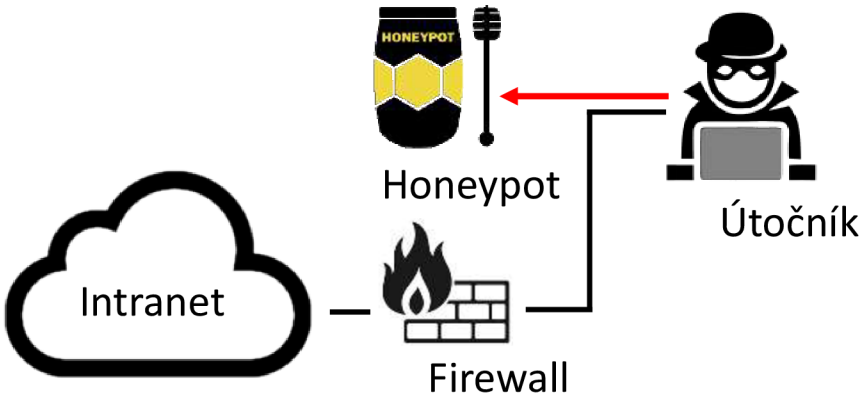
Tyto protokoly, krom WPA3 vzhledem k jeho doposud v praxi nízké rozšířenosti, budou nasazeny v části AP honeypotu. Tedy části zařízení, provádějící interakci s uživateli a možnými útočníky, kde budou bezpečnostní protokoly střídavě používány.

I přes nepřestávající vývoj těchto protokolů byla vždy nalezena v každém z nich zneužitelná zranitelnost. S odhalením útoku skrze takové systémově neošetřené nedostatky může pomoci právě WiFi honeypot, který je popsán v následující kapitole 2.

2 Honeypot

Honeypot je zařízení, navržené a realizované tak, aby svojí konfigurací vybízelo k jeho jednoduché kompromitovatelnosti skrze záměrně neošetřené bezpečnostní nedostatky. Touto metodou jsou soustředěny snahy o soustředění pokusů o infiltraci mimo skutečný, produkční informační systém. Útočník je tak pozorován při jeho počínání na zařízení, jehož konfigurační úpravou či odstavením nezpůsobí žádné omezení pro produkční síť, či jiná omezení. Díky tomu je zaznamenáno útočníkovo chování a činnosti spolu s taktikou provedeného útoku, škodlivé činnosti. Případně použité metody nebo nástroje. Následně lze proti nově zjištěným metodám útoků a použitým nástrojům ošetřit skutečnou síť nebo koncovou stanici.

Problematika honeypotů sice pochází už z 90 let, ovšem v dnešní době je toto téma stále aktuální. I po značném vývoji a výzkumu v této oblasti se objevují nové hrozby postavené na systémových trhlinách a pomyslných slabých místech v bezpečnostních systémech. To je spojené i s jednoduše dostupným penetračními nástroji a návody k možné škodlivé činnosti.



Obr. 2.1: Obecné schéma sítě s implementovaným honeypotem.

2.1 Vznik a motivace

Jak text níže napovídá, vznik a počátek honeypotů byl motivován potřebou provést na svou dobu nadstandardní zabezpečení a zaznamenáváním možných nekalých činností. Ty vedou ke zneužívání síťových služeb a odcizení dat. Vzhledem k dnešní době, stále více se orientující na používání různých datových služeb ve všech možných odvětvích lidského počínání si tyto zařízení, i přes dostupná sofistikovaná firewallová řešení a zabezpečení nynějších sítí, stále najdou využití i přízeň.

Počátek honeypotů nastal roku 1986 v knize Clifforda Stolla, Cuckoo‘s Egg. Situace systémového administrátora, kdy se sám autor Stoll snažil vystopovat chybně evidovaný poplatek na 0,75 dolarů za použití unixového systému. Stoll realizoval dva obrané mechanismy podobné honeypotům, za cílem vystopovat útočníka. Počínání útočníka při další infiltraci odchytil a zjistil, že se snažil vytěžit informace o programu jaderné obrany. Útočník byl dopaden a odhalen [18].

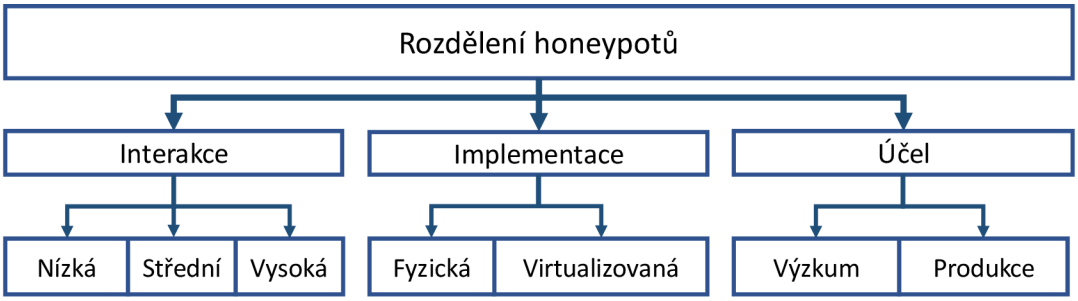
První honeypot vytvořil Bill Cheswick při práci v Bell Labs roku 1991. Systém pojmenoval „a merry chase“, s využitím typických metod honeypotů, zjistil útočníkovy techniky a zamezil mu odcizení bezpečnostních údajů [19].

Dalším milníkem pro honeypoty byl první veřejně přístupný nástroj Freda Cohena roku 1998. Ten simuloval chování systému s několika obecně známými bezpečnostními chybami. Následovala komercializace, hned pod několika variantami. V roce 2004 se dočkal první honeypot virtualizace, díky tomu bylo možné spustit na jednom zařízení větší množsví odlišných konfigurací honeypotů [20].

Definici a povědomí přináší do širší veřejnosti Lance Spitzner. Roku 1999 založil „Honeypot Project“ a publikoval „To Build a Honeypot“. Jeho definici z knihy „Honeypots, Tracking hackers“ kterou napsal zní: “A honeypot is security resource whose value lies in being probed, attacked or compromised.”, česky „honeypot je bezpečnostní zařízení, jehož hodnota vzniká, když je zkoumáno, napadeno nebo kompromitováno.“ [21].

2.2 Rozdělení

Tato zařízení pro detekci a záznam technik realizovaných útoků a škodlivé činnosti lze dělit podle míry interakce vůči útočníkům na nízkou, střední a vysokou interakci. Mezi nízkou a vysokou interakcí lze vytvořit hned několik úrovní interakce, pro účel klasifikace jsou zastoupeny střední mírou interakce. Jednotlivé rozdíly a výhody jsou popsány níže. Další rozdělení lze provést na základě implementace provedené na běžném hardwaru, případně virtualizovaně. Následně mohou být zařízení rozdělena podle účelu, jako je přímá detekce hrozby a její odvrácení od produkční sítě, či k účelům výzkumným. Toto rozdělení je shrnuto v zobrazení 2.2 [22].



Obr. 2.2: Rozdělení honeypotů.

Podle interakce

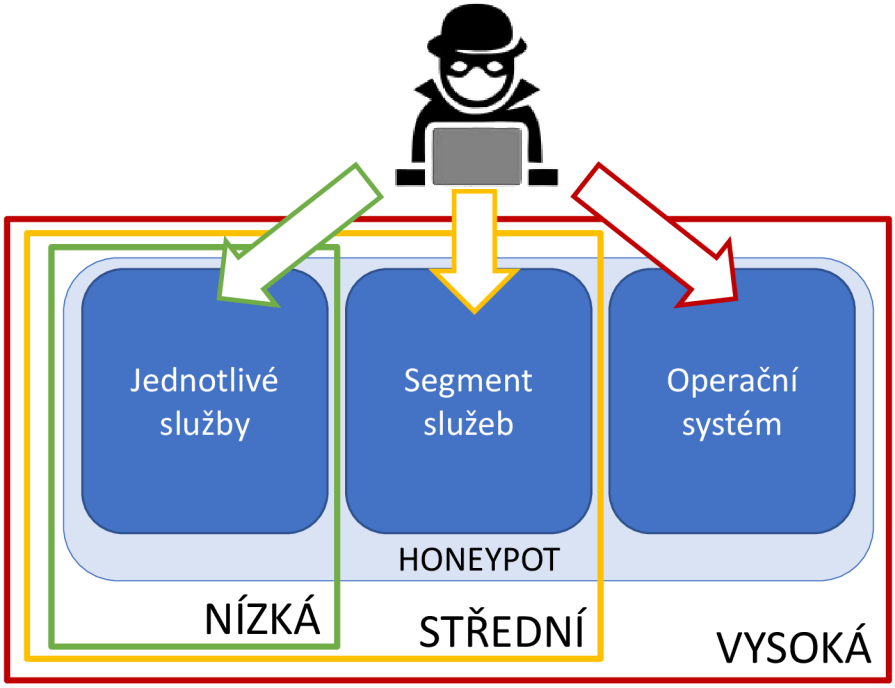
Zařízení určené k cílené interakci s útočníkem je možné rozdělit podle interakce. Podle úrovně interakce je následně určeno jaké množství síťových prostředků, případně rozhraní, portů nebo jaká část prostředí bude s případným útočníkem interagovat, jak je uvedeno na obrázku 2.3. Následující podkapitoly popisují jednotlivé interakce.

Nízká interakce

Typ nízké interakce poskytuje pro kompromitaci, interakci či manipulaci pouze úzký okruh služeb. Skrze takto poskytnuté služby může potenciální útočník realizovat své síťové útoky. Při takto nízké interakci je zařízení omezeno jen na detekci určitého, úzce stanoveného typu služeb, jako je SSH (Secure Shell).

Díky tomu lze provézt jednodušší monitorování a analýzu případných útoků na dané funkcionalitě s detekcí jejich zranitelností respektive slabin. Výhodou je i jednodušší konfigurace pro zařízení s nízkou interakcí, stejně jak jeho správa a také náročnost na výpočetní zdroje. Stejně jako nízká náchylnost na výskyt bezpečnostních rizik u takové konfigurace.

S relativní jednoduchostí daného honeypotu vzhledem k zaměření souvisejí i některé nevýhody. Jednou z hlavních nevýhod je schopnost detekovat jen relativně omezené množství různých druhů útoků. S omezeným množstvím poskytovaných služeb se pojí i riziko nezájmu o takové zařízení ze strany případného narušitele. Dále je pak zaznamenáno nižší množství dat, které lze při analýze použít, ty mohou být užitečná pro další pracování se síťovými hrozbami [22].



Obr. 2.3: Služby v honeypotu u různých úrovní interakce

Střední interakce

Kategorie střední interakce v tomto popisu zaštiťuje rozličné možnosti konfigurací spadající mezi nízkou a vysokou interakci honeypotu. Možnému útočníkovi nejsou poskytnuty pouze minimální služby, ale může jít o celý segment služeb, kterým se daná společnost, tedy produkční síť, komerčně zabývá. Nejde však o interakci vysokou, kdy je útočníkovi umožněn kompletní přístup na daný hardware se všemi službami a funkcionalitami. Zároveň je umožněn sběr dostatečného množství dat pro různé síťové hrozby a škodlivou činnost zaměřenou na určitý segment služeb [22].

Vysoká interakce

Honeypot s vysokou interakcí nabízí útočníkovi úplné prostředí. Ve většině případů jde o operační systémy (OS) obsahující obecně známé bezpečnostní nedostatky, jejichž zneužití je očekáváno. Může jít o síťové OS, případně i přímo zařízení typu směrovač, přepínač, nebo koncovou stanici.

Výhoda této konfigurace spočívá v množství zaznamenaných, obecně získaných informací. S tím je spojena i vysoká různorodost útoků a činností, které lze detekovat. Ty mohou být provedeny jak na jednotlivé službě, jako u omezenějších interakcí, tak přímo v konfiguraci honeypotu.

Na druhou stranu je u tohoto zařízení očekávána potřeba většího množství výpočetních zdrojů i kapacita datového úložiště. To vzhledem k potřebě zaznamenávat a analyzovat větší množství dat, než u nižších interakcí. Je nezbytné brát vyšší zřetel na zabezpečení pro případ převzetí honeypotu útočníkem a jeho činnosti v zařízení. Aby bylo zabráněno možné infiltraci i sítě produkční, odcizení cenných dat nebo zneužití zařízení k dalším síťovým útokům. S tím je spojená i vyšší náročnost na konfiguraci a složitost provedení celého honeypotu při této interakci [22].

Podle implementace

S ohledem na oblast nasazení honeypotu, případně žádaném typu detekovaných útoků u specifických služeb, může být provedena konfigurace několika způsoby. Přímo na fyzickém zařízení, které je nasazeno v místě působení, jako WiFi honeypot. Jinou variantou je pak nasazení virtualizované v datovém centru.

Fyzické honeypoty

Při fyzickém provedení je honeypot používán v síti ethernetové na takzvaném metalickém připojení, případně WiFi síti. Využívající Ethernet 100/1000 Base-T, nebo IEEE 802.11. Takto situovaný honeypot lze použít pro detekci specifických útoků. Může jít o nasazení konkrétně v prostředí sítí WLAN. O realizacích WiFi honeypotů, které jsou hlavním předmětem této práce, je více informací v kapitole 2.3. Fyzické honeypoty jsou nejčastěji realizovány s využitím vysoké interakce, dovolující útočníkovi kompletní "převzetí", respektive kompromitaci zařízení. Značnou nevýhodou je vyšší cena, údržba a případné přemisťování těchto zařízení po různých lokalitách [22].

Virtuální honeypoty

Varianta jednodušší na údržbu, dobře škálovatelná a umožňující běh několika na sobě nezávislých služeb, případně celých systémů, jako je honeynet . Ten je popsán v kapitole 2.2, na konci. Řešení s více systémy tak může být realizováno pomocí virtualizovaného systému v datovém centru. Běh několika na sobě nezávislých honeypotů tak může strmě zvýšit množství užitečných dat. Dále odpadá nutnost zakupovat vlastní hardware, tím je snížena vstupní investice, jelikož celá konfigurace může běžet ve zmíněném datovém centru při využití hostingu.

Nevýhodou u provedení této varianty honeypotu je omezení jen na určité útoky. Odpadá možná realizace honeypotu s podporou detekce útoků v prostředí využívající IEEE 802.11.

Podle účelu

Poslední zmíněný způsob rozdělení honeypotů je na základě jeho účelu a dělí se na zařízení určené k ochraně produkční sítě a zařízení ve výzkumné sféře. U honeypotu navrženého k ochraně produkční, podnikové sítě s cílem minimalizovat možné ohrožení bude konfigurace a robustnost řešení koncipována jinak, než zařízení určené pro výzkumné účely. U zařízení pro výzkumné účely lze očekávat i odlišnou reakční dobu, priority řešení incidentů a nižší schopnost zpracovat velké objemy dat.

Honeynet

V případě potřeby realizace robustního zabezpečení je možným řešením Honeynet. To je seskupení honeypotů simulujících komplexní datovou síť. V případě takto složité realizace, je běžně zajištěna i automatická komunikace honeypotů mezi sebou samotnými. Případný útočník musí postupně odhalovat falešné síťové prvky, čímž na sebe sám často upozorní, navíc je tento proces velmi zdlouhavý. Tím může být i odrazen v pokračování kompromitování dalších prvků a pronikání do reálné "skryté, produkční sítě"[22].

Jde sice o nákladnou metodu jak po stránce finanční - v závislosti na typu konečné realizace fyzické nebo virtualizované, tak s náročnou a vysoce komplexní konfigurací. Je ovšem možné získat vysoké množství dat k analýze, stejně jako vysoké zabezpečení produkčních, síťových zařízení a sítě samotné.

2.3 IEEE 802.11 Honeypoty

Historicky nebyla kladena příliš velká pozornost na honeypoty realizující detekci v rádiovém prostředí využívající IEEE 802.11. Podstatná část honeypotů byla zaměřena na zabezpečení, a s tím spojenou detekci síťových hrozeb odehrávajících se přes metalické, potažmo optické datové médium. Souviselo s tím hned několik historicky podstatných faktů, mezi které patřily nízké znalosti bezdrátových sítí a jejich "zranitelných" míst, tak i menší znalost a rozšířenost škodlivých nástrojů. Stejně jako potenciál odcizení hodnotných dat. V dnešní době masivního nasazení bezdrátových technologií, online bankovnictví a přenosu vysoce citlivých dat přes WiFi sítě jsou rizika síťového útoku podstatně vyšší. Nebezpečnosti přidává i obecně větší povědomí o možných způsobech, jak tyto sítě infiltrovat a jednoduchá dostupnost nejrůznějších penetračních nástrojů.

V následujících podkapitolách jsou popsány základní metody realizace WiFi honeypotu, způsobu detekce útoku, spolu s jejich základním kategorizováním.

Metody

WiFi Honeypot běžně funguje jako falešné AP v geograficky dobře přístupné síti náhodným uživatelům. Aby byl takto realizovaný honeypot pro případného útočníka zajímavý, je vhodné jej opatřit rádoby běžně se vyskytujícím jménem sítě, tedy Service Set Identifier (SSID), stejně jako zabezpečením a obecně jeho konfigurací. Vhodným doplněním je vytváření skutečných uživatelů, připojených k AP, případně s náhodnou četností se automaticky připojující zařízení generuje alespoň základní datový provoz. Při případném průzkumu útočníkem tak bude k AP připojen simulovaný klient, který při spektrální analýze, či odposlechu dat bude vytvářet datový tok [22], [23].

Logování

Pokud by nebyl datový provoz přenášený přímo přes WiFi honeypot, nebo v jeho bezprostředním prostředí zaznamenáván, jednalo by se pouze o běžné AP, bez žádné přidané hodnoty. Proto je u těchto zařízení zásadní provádět logování nejrůznějších událostí, jak skrze systémové logy, tak probíhající datové komunikace. Především monitorování rádiového prostředí, spolu s ukládáním těchto dat, je klíčovým u WiFi honeypotu.

Při logování je tedy kladeno za cíl zaznamenat co největší množství informací o činnosti potencionálního útočníka. Dá se obecně shrnout, že čím více informací je shromážděno, tím rozsáhlejší jsou možnosti následné analýzy. Aktivity, které je zásadní u WiFi honeypotu detekovat a zaznamenávat jsou [22], [23]:

- skenovatelné informace o okolních WiFi sítích,
- datová komunikace v okolním rádiovém rozhraní,
- nativní logy AP (vysoká interakce).

Informace, které lze získat skenováním okolních sítí mohou postačit k detekci možných útoků a další nekalé činnosti možných útočníků. Může jít o záplavový útok generující SSID neexistujících sítí, klonování reálných uživatelských sítí a mnohé další. Běžné útoky z prostředí WiFi sítí jsou více popsány v kapitole 3.2.

Datová komunikace odehrávající se v rádiovém prostředí poskytuje informace a potenciál detekovat nejvíce možných útoků. Z tohoto důvodu bude na tento typ sběru dat kladen nejvyšší důraz. Jak informace získané pomocí skenování okolních sítí, tak získaná monitorováním rádiového prostředí mohou být zpracovány Systémem detekce narušení, anglicky Intrusion Detection System (IDS). Dodaný v rámci komplexní aplikace, případně vytvořený vlastním skriptem, jde o nástroj pro detekci

narušení v daném síťovém prvku, konkrétně přímo v honeypotu. IDS pak často realizuje možná bezpečnostní opatření, či notifikace uživatele tohoto detekčního zařízení [23].

Během klasifikace zachycené datové komunikace je následně nezbytné zaměřit se na několik zásadních faktorů, které ovlivňují přesnost vyhodnocení a množství detekce útoků. Jde především o monitorování razantního nárůstu konkrétního typu komunikace, který může realizovat odepření služeb, anglicky Denial of Service (DoS) daného honeypotu, respektive AP, které na něm běží. Dále o specifické protokoly, používané k řízení datové komunikace, modelově Address Resolution Protocol (ARP).

Při detekování potenciálně škodlivé činnosti, nejen pro WiFi honeypot samotný, ale v obecné rovině, je vhodné uložit data permanentně. Je vhodné tyto i externě zálohovat pro možnost zpětné analýzy případným uživatelem, správcem, s co nejvíce možnými a prvotně získanými parametry. Základní potřebné informace pak jsou adresy MAC, adresy Internet Protocol (IP) a typ datové komunikace.

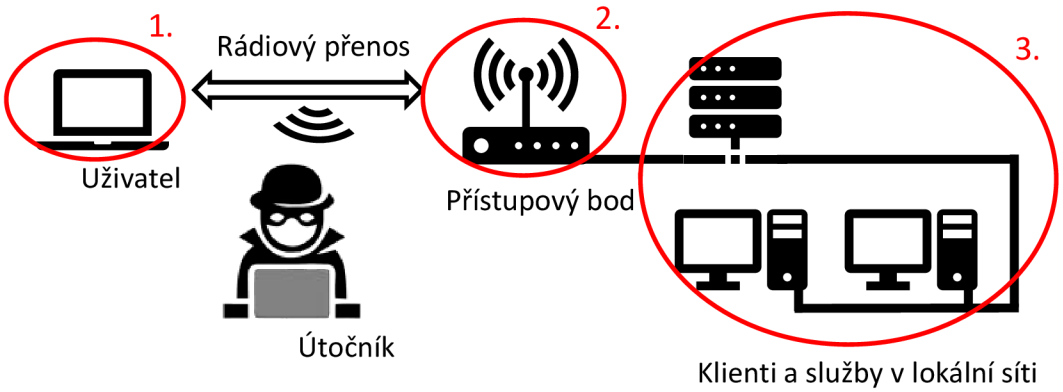
V této diplomové práci realizující WiFi honeypot lze využít všech technik zaznamenávání datové komunikace a činnosti možného útočníka, zmíněných výše. Mezi možnostmi je využití datových kanálů iptables, ze kterých lze logovat přímo do systémových logů, v případě linuxu. Další varianty zaznamenávání datové komunikace jsou s využitím nástrojů jako TCPdump, Wireshark a jiných. Možné využitelné softwarové nástroje jsou zvláště popsány v kapitole 4.3.

Útoky

Útoky probíhající v rádiovém prostředí používající IEEE 802.11 je možné rozdělit do několika základních skupin. Základní rozdělení, podle cíle útočníka, je znázorněno na obrázku 2.4 do tří kategorií:

1. na uživatele daného AP,
2. na samotné AP a jeho rádiové rozhraní,
3. na infrastrukturu za AP.

U jednotlivých cílů těchto útoků lze očekávat různé záměry a také dopady úspěšného útoku. Při útocích na uživatele AP jsou nejčastěji kladeny cíle na odcizení soukromých dat, případně odepření přístupu k službám AP. Drtivá většina z útoků na AP cílí na odepření jeho služeb spolu s převzetím iniciativy komunikace a případné řízení a filtrování uživatelských dat s cílem jejích modifikace, případně odcizení. Některé z útoků na AP mohou vést ovšem do vzdálenější části sítě. To v podobě datových úložišť a pracovních stanic obsahujících hodnotná data, pro odcizení a jejich poškození.



Obr. 2.4: Možné cíle útoků pomocí IEEE 802.11.

Aby bylo možné zvýšit šanci na zachycení konkrétních typů útoků a pokusů o infiltraci, je nezbytné přizpůsobit této činnosti provozovaný honeypot. Takové zařízení pak v závislosti na požadované míře interakce, které jsou popsány v kapitole 2.2 a poskytovaných službách využívá odlišných vrstev modelu ISO/OSI, jak je uvedeno v kapitole 4. Běžné kategorie pro typy konfigurací WiFi honeypotů [24]:

- útoky na linkovou vrstvu ISO/OSI,
- útoky mířené proti bezdrátové infrastruktuře,
- útoky na připojené klienty,
- útoky proti kabelové síti,
- kombinovaná konfigurace.

3 Analýza zranitelnosti v rádiovém prostředí IEEE 802.11

Sítě komunikující pomocí protokolů IEEE 802.11 mají hned několik bezpečnostních nedostatků, respektive slabých míst. Ty lze použít k převzetí kontroly nad některými síťovými zařízeními, odposlechu datové komunikace, případně znemožnění komunikace ostatním uživatelům připojeným k danému AP. V této kapitole jsou popsány běžně používané typy útoků a dostupné nástroje využitelné při ověření teoreticky popsaných zranitelností rádiové komunikace za pomocí IEEE 802.11. Zároveň je popsána konfigurace penetračních nástrojů a návrh pro pozdější testování detekce honeypotu.

3.1 Software pro analýzu zranitelnosti

Pomocí dostupného software pro analýzu zranitelností v prostředí IEEE 802.11 lze prověřit běžné bezpečnostní nedostatky, popsané v kapitole 3.2. V této práci bude použit OS Kali 2019.4, ze kterého lze realizovat penetrační síťové testy, se zaměřením na zabezpečení WEP, WPA a WPA2. Velká část těchto penetračních nástrojů je v této distribuci již předinstalována a určena k přímému použití. Spolu s velkým množstvím dalších dostupných penetračních nástrojů.

Kali 2019.4 je podrobněji popsán v kapitole 4.2, spolu s Ubuntu 18.04 LTS, HoneyDrive 3, Network Security Toolkit 30 a Security Tool Distribution jako varianta pro použití host OS přímo v honeypotu.

Využité nástroje pro analýzu zranitelnosti IEEE 802.11, ověření jednotlivých vektorů útoků a otestování detekce honeypotu jsou:

- Aircrack-ng,
- Airodump-ng,
- Aireplay-ng,
- mdk3,
- arpspoof,
- hping3.

Aircrack-ng nabízí monitorování a zachytávání dat, deauthentication útoky, zřízení falešeného přístupového bodu a další packet injection útoky. Mimo to lze použít na prolamování WEP a WPA i WPA2. Je určen jak pro Linuxové distribuce, tak pro Windows, OS X a další [25].

Airodump-np vhodně doplňuje další použité penetrační nástroje jako Aireplay-ng, mdk3, arpspoof a případně hping3. Jeho použití spočívá v zachycení datových

rámců. Lze ho přímo použít k zachycení IV u WEP, nebo jako v tomto případě k detekci MAC adresy možných cílů - uživatelů a AP, které budou dále podléhat penetračnímu testování. Níže zobrazený výpis 3.1 ukazuje záznam naskenovaného AP [25].

```
airmon-ng start wlan0
airodump-ng wlan0mon
airodum-ng -d <MAC adresa AP> -c <kanál> wlan0mon
aireplay-ng -0 0 -a <MAC adresa AP> -c <MAC adresa klient> wlan0mon
```

```
airodump-ng -d 64:EE:87:C6:75:D5 -c 1 wlan0mon

CH 1] [Elapsed: 1min] [2020-04-09 19:52] [fixed channel wlan0mon: 1
BSSID              PWR   RXQ Beacons #Data,#/s CH  MB ENC CIPHER AUTH ESSID

64:EE:87:C6:75:D5 -52  0   979       266    53  1   130 WPA2 CCMP PSK  wifi

BSSID                 STATION            PWR   Rate  Lost  Frames  Probe

64:EE:87:C6:75:D5  50:E0:85:87:49:76  -49    1 - 6e    18     198
```

Výpis 3.1: Zaznamenání MAC adres klienta připojeného k AP

Aireplay-ng nabízí hned několik nástrojů pro penetrační testování WLAN sítí, jeho primární účel je však doplnění palety funkcionalit Aircrack-ng pro prolamování klíčů u WEP, případně WPA šifer. Jeho běžné použití spočívá v generování datového provozu. Tím běžně může být generování deautentizačních rámců, právě za cílem zachytit datový provoz, následně se znovu připojujícího klienta k AP. Pro upřesnění je níže vypsán soupis možných typů útoků realizovaných pomocí Aireplay-ng [25]:

- parametr 0 - Deautentizační útok,
- parametr 1 - Autentizační útok,
- parametr 2 - Interaktivní přehrání paketů,
- parametr 3 - ARP požadavek opakovaného zaslání,
- parametr 4 - KoreK chopchop útok,
- parametr 5 - Cafe-latte útok,
- parametr 6 - Fragmentační útok,
- parametr 7 - WPA migrační režim,
- parametr 8 - Injekční test.

Příkaz `aireplay-ng` ukazuje základní konfiguraci použitého Aireplay-ng deautentizačního útoku při testování detekce deautentizačního útoku:

```
aireplay-ng -0 0 -a <MAC adresa AP> -c <MAC adresa klient>
```

mdk3 je nástroj určený pro účely analýzy zabezpečení WLAN sítí, využívajících IEEE 802.11. Obdobně jako Aireplay-ng i tento nástroj nabízí hned několik typů síťových útoků, kdy při testování honeypotu byly opakovaně použity autentizační DoS útoky, beacon (SSID) záplavové útoky jak slovníkové, tak s generováním náhodných SSID.

Při autentizačním DoS útoku je prováděna záplava autentizačními rámci. Při tomto konkrétním příkazu je typ útoku definován parametrem a - autentizační DoS útok. Následně parametr -a určuje cíl útoků, na který budou autentizační záplavové rámce zasílány.

```
mdk3 wlan0mon a -a <MAC adresa AP>
```

U testování beacon (SSID) záplavového útoku bylo testováno vysílaní na náhodném kanále. Parametr b definuje typ útoku - beacon/SSID záplava. V tomto případě byly generované SSID náhodně a složené z různých znaků. V případě vysílání konkrétních SSID názvu, jak zobrazuje druhý příkaz níže s parametrem -f, je odkazován na seznam vysílaných SSID. Dále je zde navíc parametr -c, kterým je vybrán kanál, na kterém jsou beacon rámce s falešnými SSID vysílány.

```
mdk3 wlan0mon b
mdk3 wlan0mon b -f <cesta-do-txt-obsahujici-ssid> -c <kanál>
```

Další terminálový nástroj ARPspoof bude použit pro ARP spoofing, tedy vysílání falešných ARP zpráv. Tento útok může indikovat počátek Man-In-The-Middle útoku, případně jinou škodlivou činnost. Při provádění tohoto testu jsou, jak ukazují příkazy níže, přeposílány dotazy mezi uživatelem a AP. Pro provedení tohoto útoku musí být povolen forwarding na zařízení provádějící tento útok.

```
arpspoof -i wlan0 -t <IP adresa klient> <IP adresa AP>
arpspoof -i wlan0 -t <IP adresa AP> <IP adresa klient>
```

Posledním použitým penetračním nástrojem je hping3. Tento generátor a analyzátor síťového provozu umí pracovat s fragmentací, umožňuje libovolně definovat velikost paketů a může být použit u přenosu celých souborů, při jejich zapouzdření. Níže je zobrazen příkaz pro generování Transmission Control Protocol (TCP) SYN DoS útoku, obsahujícího hned několik použitých parametrů.

```
hping3 -c 10000 -d 120 -w 64 -flood -rand-source <IP adresa AP>
```

- -c počet zaslání (10000x)
- -d velikost datových rámců (120 byte)
- -w velikost TCP okna (64 byte)
- – flood definice typu zasílání dat - záplavově
- –rand-source definice zdroje dat - náhodně generovaný

3.2 Vektory útoků

Jsou souborem metod, jejichž pomocí lze zneužít zranitelnosti systému. Vektorem útoku je zařízení kompromitováno za pomocí technik sociálního inženýrství, škodlivého kódu, případně kombinací obou zmíněných. V rádiovém prostředí, které využívá protokolů IEEE 802.11 jde pak o prolamování šifrovacích protokolů, odepírání služeb a odcizení citlivých dat. V této kapitole jsou popsány nejběžnější techniky, s kterými se lze setkat v uživatelsky běžném, reálném prostředí.

Deautentizační útok

Pomocí útočníkem řízené deautentizace lze odpojit uživatele od AP. Tímto útokem může dojít pouze k záměrnému odepření služeb u konkrétního AP. Ovšem běžně je této techniky využito jako prvního kroku při snaze infiltrovat danou WLAN sít. Následně může útočník proniknout do dané sítě a realizovat útok "muže uprostřed"(Man in the Middle - MITM).

Po provedení, standardně opakované deautentizace, se útočník zaměřuje na zachycení dat přenášených při procesu znovunavázání komunikace mezi uživatelem a AP, nazývaný *"handshake"*. Z těchto dat jsou extrahovány informace o přístupovém bodu, které jsou dále zneužity.

Autentizační záplavový útok

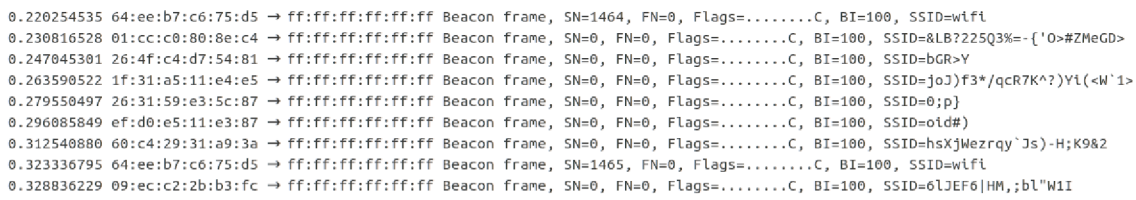
Tento záplavový útok je zaměřený na proces autentizace uživatele. Útočník zasílá velké množství autentizačních rámců na AP, čímž postupně vytíží všechny zdroje daného zařízení a dojde tak k odepření služeb DoS daného AP pro všechny uživatele [26].

Takto realizovaný útok nemusí mít žádný faktický užitek pro útočníka, který ho realizuje. S použitím dalších technik však může dojít k podvržení falešného (útočníkova) AP, na které se možní uživatelé pokusí připojit. Během procesu připojení pak mohou být odposlechnuty a zneužity přihlašovací údaje těchto klientů, to celé realizováním například Evil Twin AP útoku.

Beacon (SSID) záplavový útok

Technika, během které nedochází k přímé interakci, respektive útoku na AP nebo uživatele. Jde o záplavový proces, během kterého jsou generovány beacon rámce. Jak je na obrázku 3.1 s náhodně vygenerovanými SSID nebo případně slovníkově, v podobě textového dokumentu, stanovených konkrétních SSID. V případě na snímku je zobrezeno náhodné generování zdrojové MAC adresy útočníka, což může při detekci působit jako další faktor, stěžující detekci tohoto útoku [26].

Dopad na uživatele spočívá v zobrazení velkého množství neexistujících AP. Tím je minimálně zhoršena uživatelská zkušenost, přes znemožnění využití WiFi sítě, po odcizení přihlašovacích údajů při snaze připojit se na podvržené AP.

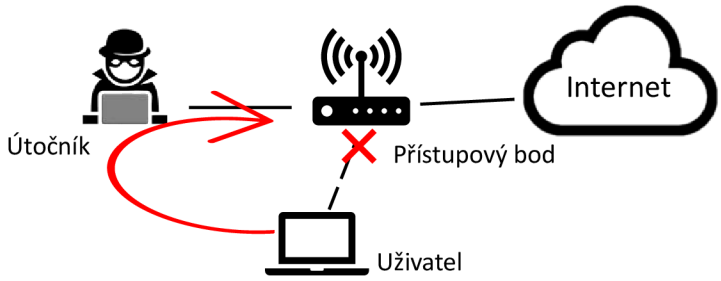


Obr. 3.1: Zachycený beacon (SSID) záplavový útok s náhodnými SSID

IP spoofing

Tento proces, během kterého dochází k vytváření datové komunikace, která cíleně manipuluje s cizí, již sestavenou datovou komunikací. Může tak dojít k vměšování se do dané datové komunikace, jak je zobrazeno na snímku 3.2. Při převzetí komunikační iniciativy ze strany útočníka a umožnění podvrhovat, odposlouchávat, či jinak poškodit data, která v takovéto podobě mohou procházet přes zařízení narušitele komunikace, tento konkrétní útok je MITM [27].

Pro realizaci tohoto útoku musí útočník proniknout do dané WLAN sítě, to je možné s využitím Evil Twin útoku, případně Deautentizačního útoku a zachycením dat potřebných k rekonstrukci hesla dané sítě.



Obr. 3.2: Příklad útoku Man in the Middle, využívajícího spoofing

Key Reinstallation Attack - KRACK

Je útok proti zabezpečení WPA2, konkrétně handshake známý u WPA2 jako 4-Way Handshake. Proces útoku se skládá z opakovaného připojování a resetování Nonce obsahujícího sdílené heslo v třetím kroku handshake. Po sběru většího množství dat je útočník schopen vzájemně porovnat zašifrované pakety a odhalit sdílené heslo [28].

3.3 Detekované útoky

Realizovaný WiFi honeypot je zaměřen na detekci útoků a škodlivé činnosti založené na vektorech útoků, popsaných v kapitole 3.2. Jednotlivé vektory útoků mohou vést k prvoplánovému odepření přípojení k AP, jako v případě Deautentizačního útoku nebo k podnícení opakovaného procesu připojení uživatele k AP. To následně může vést k infiltraci dané počítačové sítě, provedení IP spoofingu, odcizení dat, či jejich podvržení.

Pro přehlednost schopností rozlišit jednotlivé hrozby pomocí tohoto WiFi honeypotu byly možné síťové útoky popsány v tabulce 3.3 s patřičným popisem. Tam za pomoci jednotlivých vektorů útoku může být detekováno i větší množství navazující intrusivních a škodlivých aktivit.

Popis jednotlivých útoků v tabulce 3.3 zahrnuje popis vrstvy ISO/OSI, na které se daný útok odehrává, pokud tak lze specifikovat, dále s popisem možného rizika pro uživatele, konfigurace AP, jeho funkčnosti a odcizení dat. Poté se souhrnně popsaným průběhem dané hrozby, kdy je definováno, v čem daný útok spočívá. Poslední popisovanou položkou je metoda detekce, kdy je popsána technika, kterou honeypot využívá, aby aktivně vyhledával možné incidenty, případně je zaznamenával v podobě logů a statistických údajů.

Jméno útoku	ISO/OSI	Roziko	Průběh	Metoda detekce
Deautentizační útok	L2	Odepření služeb/zachycení šifrovacího klíče	Generování velkého množství deautentizačních rámců	Detekce neúměrně vysokého počtu deautentizačních rámců
Rogue access	L3, L4	Proniknutí útočníka do LAN sítě	Připojení neautorizovaných uživatelů k AP	Na základě deautentizačního útoku
Autentizační DoS útok	L2	Odepření služeb AP	Generování velkého množství autentizačních rámců	Detekce neúměrně vysokého počtu autentizačních rámců
Beacon (SSID) záplava	L2	Pokus autorizovat uživatele na falešném AP/zachycení šifry AP	Generování falešných SSID	Detekce neúměrného nárůstu SSID
ARP Spoofing	L2	Manipulace s ARP, použití k MITM	Data prochází přes bod konktrolovaný útočníkem	Detekce anomálií v ARP komunikaci
Man-in-the-Middle (MITM)	-	Manipulace s uživatelskými daty, odcizení dat	Data prochází přes bod konktrolovaný útočníkem	Na základě detekce ARP Spoofingu
TCP SYN DoS útok	L4	Odepření služeb AP	Generování velkého množství TCP SYN paketů	Detekce neúměrně vysokého počtu autentizačních rámců
Manipulace s AP	-	Manipulace a podvržení uživatelských dat, poškození konfigurace AP	Připojení se do AP, úprava konfigurace	Analýza logů Auditd, hostapd.log, kern.log, auth.log, var.log

Obr. 3.3: Detekované typy útoků

4 Návrh vlastního honeypotu

Při výběru zařízení je nutné brát v úvahu hned několik faktorů. V závislosti na účelu plánovaného honeyopotu je vybráno zařízení s dispozicemi pro běh vhodného OS spolu s dostupným, potřebným software a ovladači daného hardware. Zařízení dále musí podporovat potřebné WiFi standardy, jako IEEE 802.11. Tato kapitola v jednotlivých částech popisuje výběr zařízení, operačního systému, možný software pro WiFi honeypot a volbu jeho architektury.

Umístění v rámci ISO/OSI

Podle požadavků na funkce honeypotu může být specifikováno, na kterých vrstvách systému ISO/OSI zařízení realizuje činnost detekce a možné reakce na případné síťové útoky.

Zatímco konfigurace zaměřená na nízkou interakci může vyžadovat pro funkci datového snifferu vrstvu linkovou a síťovou. Tak konfigurace s vysokou interakcí nabízející útočníkovi kompletní prostředí zahrnující různé nástroje a služby, bude vyžadovat i vrstvu aplikační.

Umístění honeypotu v rámci ISO/OSI závisí na jeho účelu, míře interakce a softwarových nástrojích, které bude používat. V případě právě navrhovaného a vytvářeného honeypotu bude zařízení pracovat a provádět detekci na všech vrstvách. Díky střídavě nabízené vysoké a nízké interakci, kdy při vysoké interakci může případný útočník vstoupit přímo do OS, na němž může provádět konfigurační změny, nebo využívat služeb běžících na různých vrstvách ISO/OSI.

4.1 Hardware

Pro vytvoření honeypotu může být použita široká škála informačních zařízení. V případě WLAN honeypotu pak síťová zařízení, obsahující rádiové rozhraní, umožňující komunikovat pomocí protokolové sady IEEE 802.11. Vybírat hardware lze podle podporovaných periferií, škálovatelnosti i výpočetního výkonu. Tento seznam zobrazuje rozdělení na typy síťových zařízení s potenciálním využitím:

1. Mikrokontrolér,
2. WiFi směrovač,
3. mITX/mATX základná deska.

Pro honeypot s podporou protokolů IEEE 802.11 je nutné, aby použité zařízení disponovalo dostatečným rádiovým rozhraním se schopností pojmout vyšší datové

toky, které mohou odpovídat útokům DoS. Předpokládá se i podpora co nejvíce šifrovacích protokolů, popsaných v kapitole 1.2.

Zařízení bude obsahovat přístupový bod (AP), z toho důvodu se předpokládá dostatečná mobilita zařízení za účelem přenosu pro nasazení v cílové destinaci bez nutnosti složitějších příprav. Další, neméně podstatný požadavek pro cílové zařízení, musí být schopnost pracovat s potřebnými nástroji a aplikacemi, které mohou být dostupné pouze na konkrétní distribuci operačního systému. Více o možnostech jednotlivých operačních systémů a konkrétních výhodách je popsáno v kapitole 4.2, spolu s těmito požadavky musí zařízení disponovat dostatečným výpočetním výkonem. Aby navržená architektura honeypotu mohla stabilně běžet, bez možného přerušení vyvolaného nedostatečnými hardwarovými zdroji.

S ohledem na sepsaná kritéria byly z jednotlivých skupin síťových prvků vybrány reprezentativní modely, které mají díky svým parametrům potenciál pro nasazení jako WiFi honeypot.

Mikrokontroléry

Mikrokontroléry, jako Raspberry Pi a Bannana Pi jsou obecně úspornou variantou. Za nízkou investici nabízí relativně výkonné řešení. Běžně fungující na některé z Unixových platforem. Umožňují přímé využití, bez potřeby změny operačního systému. Toto může být vhodná varianta pro honeypot s nízkou nebo střední interakcí. Ve shrnutí:

+ nižší pořizovací cena ve srovnání s WiFi směrovači i mITX/mATX deskou,
+ dobře přenosné, malé zařízení,
- dokoupení dalšího příslušenství a komponent jako externí WiFi systém,
- nižší výpočetní výkon a horší škálovatelnost oproti mITX/mATX desce,
- horší provedení rádiové části.

Jednoúčelové bezdrátové směrovače

Jednoúčelové WiFi směrovače reprezentované zařízením Netgear Nighthawk XS4 AC2600 lze vyznačit často robustnější, dobře provedenou, rádiovou částí vzhledem k určené funkci zařízení. To není ovšem podmínkou. Při využití zařízení jako honeypotu je nutné vzít v úvahu vyšší výpočetní nároky oproti standardnímu provoznímu režimu. Výpočetní zdroje jsou nezbytné pro běh systému dovolujícího specifickou konfiguraci, a proto jsou tato zařízení v roli honeypotu vhodné spíše pro nízkou interakci. Obecně pak:

+ komplexní řešení, bez nutnosti rozšiřování o další příslušenství,

+ zpravidla robustní rádiová část směrovače,
+ dobrá podpora IEEE 802.11, MIMO,
- horší výpočetní výkon zařízení,
- malá operační i úložná pamět, vyplývající z jednotvárného provozního scénář.

Základní desky miniITX

Za kompromis mezi mikrokontroléry a jednoúčelovými WiFi směrovači lze považovat základní desky menších formátů jako mITX a mATX. Modelově GIGABYTE X570 I AORUS PRO WIFI. Se zaměřením na nativní WiFi perifierii a podporu IEEE 802.11 protokolů. Mimo to má tato deska další periferie jako PCI-Express (PCIe), Universal Serial Bus (USB). To zajišťuje všestrannost těchto zařízení, ovšem na úkor vyšší ceny a potřeby dokoupit nezbytné komponenty k osazení. Finálně osazené zařízení potřebnými komponenty však disponuje parametry výrazně převyšujícími ostatní zvažované varianty, tedy mikrokontroléry a jednoúčelové WiFi směrovače. Konkrétně GIGABYTE X570 I AORUS PRO WIFI je vhodná pro běh honeypotu s libovolnou mírou interakce, umožňující sandboxing, případně virtualizaci části výpočetních zdrojů. Dále je umožněna i vysoká škálovatelnost a relativně jednoduchá obměna jednotlivých komponent.

+ všestranně řešené zařízení s možnou implementovanou WiFi,
+ výborné parametry, vysoký výpočetní výkon,
+ dobrá škálovatelnost a obměna osazených komponent,
- potřeba dokoupit komponenty *(CPU, RAM, HDD/SSD, zdroj a další)*,
- riziko nekompatibilit a vzrůstající cena.

Vzhledem k výše popsaným parametrům jednotlivých skupin síťových zařízení bylo vybráno pro použití základní desky typu mITX. Vybraná kategorie hardware nabízí kompromis dostatečně zpracované rádiové části, podpory potřebných protokolů IEEE 802.11, spolu se škálovatelností a multifunkčnosti platformy mITX desky.

Vybrané řešení nabízí i podporu pro výpočetně i kapacitně náročnější OS. Tím pádem i úkony v roli honeypotu podléhající potřebě vyššího výpočetního zatížení při zpracování dat a nasazení náročnějšího software a funkcionalit, jak ve vysoké i nízké interakci honeypotu.

Zařízení, zaznamenané na obrázku 4.1, zobrazuje zakoupené a sestavené zařízení pro realizaci WiFi honeypotu. Výběr zařízení podléhal mimo další hlediska i kritériím na dostatečný výpočetní výkon, paměťovou kapacitu, podporu potřebných protokolů IEEE 802.11 a přenosových technik MU-MIMO a vzájemné podpory hardware.

Obr. 4.1: Zobrazení vybrané mITX základní desky

Na přiloženém snímku lze vidět 2 WiFi systémy, kdy první je přímo připojen do základní desky pomocí RSMA koaxiálních kabelů, do integrovaného čipu Intel WiFi 6 AX201. Ten je určen se svou širokou podporou IEEE 802.11 pro realizaci AP. Druhým připojeným WiFi systémem je USB WiFi adaptér s čipem RTL8812AU, který má realizovat monitorovací a skenovací funkce. I tento hardware splňuje v rámci typu tohoto přípojného rozhraní vysoké požadavky na podporu až IEEE 802.11ac, s možnou podporou AP režimu, mimo běžněji podporovaný režim monitorování a stanice. Tabulka 4.1 zobrazuje konečný seznam zakoupených komponent pro realizaci WiFi honeypotu.

Množství [ks]	Komponenta [-]
1	Základní deska mITX GIGABYTE X570 I AORUS PRO WIFI
1	Procesor AMD Ryzen 3 2200G
1	SSD Patriot M.2, 128 GB
2	RAM 8 GB DDR 4 Hyper Fury
1	PC skříň mTXI Cooler master
1	Napájecí zdroj Elementium E2 SI-350 W
1	USB WiFi NETIS, AC1200 Wireless

Tab. 4.1: Komponenty použité při realizaci honeypotu

4.2 Operační systém

Podle konkrétního účelu a typu honeypotu je nezbytné vybrat vhodný OS, podporující potřebné doplňkové služby, umožňující instalaci potřebného software, ovladačů pro hardware. Vybraný OS by měl splňovat i podporu napříč co nejširší škálou možného hardware pro podobné nasazení kopie vytvořeného honeypotu, stejně jako dostupnost kvalitní dokumentace a udržování ze strany vývojářů.

Vzhledem k jednodušší manipulaci s OS samotným, souvisejícími zdroji a periferiemi bude výběr operačních systému pro běh honeypotu vymezen na linuxové distribuce. Na základě vybraného hardware pro tuto realizaci v kapitole 4.1 jsou zde popsány OS s potenciálem na použití pro host OS:

- Ubuntu 18.04 LTS,
- Kali 2019.4,
- HoneyDrive 3,
- Network Security Toolkit 30 (NST 30),
- Secority Tool Distribution (STD).

Vybrané OS svým zaměřením pokrývají širokou oblast funkcí. Od běžně používaného Ubuntu s podporou velkého množství aplikací jak běžně uživatelských, tak zaměřených na síťovou bezpečnost. Kali, charakteristický pro penetrační testování komplexních systémů, i jednotlivých síťových rozhraní. Nabízí však i nativní podporu monitorování síťového provozu. HoneyDrive je pak zástupce OS stavěného přímo pro běžnou činnost honeypotu. Výběr zakončují systémy NST a STD. Ty zastupují skupinu úzce zaměřených OS, konkrétně na zabezpečení a práci s možnými bezpečnostními incidenty.

Ubuntu 18.04 LTS

První z výběru systému je Ubuntu 18.04 LTS. OS Ubuntu patří k systémům bez specifického, úzkého zaměření. Je hojně využíván a nabízí masivní podporu ovladačů s tím spojeného hardware. Stejně tak i programové výbavy. Operační systém je volně šiřitelný, bez licenčního omezení. Systém vyžaduje vyšší množství zdrojů, umožňuje ovšem realizaci běhu virtualizačního software, vizualizaci dat honeypotu například pomocí Cacti, Splunk, či jiného. Software s sebou nepřináší žádné předinstalované aplikace realizující činnost honeypotu, ani IDS. Při použití tohoto OS se předpokládá využití skriptů a menších softwarových aplikací pro realizaci dílčích činností honeypotu [29].

Kali 2019.4

Kali je linuxová, volně šiřitelná distribuce odvozená od Debianu. Systém navržený především pro penetrační testy a síťový monitoring. Předchůdcem systému je BackTrack, který byl roku 2013 editován na Kali. Nabízí podporu softwarových nástrojů přímo pro činnost honeypotu jako honeyd. Drtivá většina nativních nástrojů je však zaměřena na monitoring sítí, dešifrování dat a penetrační testy.

Ve srovnání s dalšími popisovanými OS jde, spolu s Ubuntu, o nejnáročnější distribuci, co se potřebých zdrojů týče. Distribuci lze vhodně využít i při analýze a penetračních testech WLAN sítě používající IEEE 802.11. Lze aplikovat testy zahrnující ověřování možných vektorů útoků na rádiovou část sítě, jak je zmíněno v kapitole 3.1. Jsou podporovány instalace na architekturách i386, ARM a amd64. Vzhledem k velkému množství předinstalovaných a podporovaných penetračních nástrojů bude tento OS použit minimálně při analýze zranitelností IEEE 802.11 a penetračních testech honeypotu. O jeho možném použití pro honeypot samotný rozhodne praktické otestování jeho vlastností a podpory pro konkrétní hardware [30].

HoneyDrive 3

Je jednou z předních distribucí s úzkým zaměřením přímo na funkci honeypotu. Distribuce je určena pro běh ve virtualizovaném software, konkrétně Oracle VirtualBox, nebo VMware Workstation, ESXi. Celá nadstavba honeypotu vychází z OS Xubuntu 12.04 LTS. K přímemu nasazení nabízí hned několik předinstalovaných aplikací pro funkci honeypotu, IDS, malware detekce, virtualizace vyhodnocených dat, a s tím spojenými nástroji. Patří mezi ně Kippo SSH, Dioneaea, Glastopf, Wordpot, Honezd-Viz, Kippo Graph a další.

Většina zmíněných nástrojů je však zaměřena na pracování s možnými hrozbami cílenými na konkrétní služby, jako je SSH nebo mailové daemony. Není tedy příliš vhodný pro práci s rádiovým rozhraním [31].

Network Securiry Toolkit 30

NST 30 je distribucí zaměřenou na síťové zabezpečení vycházející z distribuce Fedora. Nabízí hned několik předinstalovaných aplikací s otevřenou licencí. Mezi ně patří Wireshark, Ntopng, Arp scanning, Snort, NMap, Kismet a další. NST 30 funguje na systémech založených x86_64 a může být použit jako bootovací USB, případně i nainstalován, obdobně jako STD [32].

Security Tool Distribution

STD vycházející z linuxové Knoppix distribuce nabízí kolekci volně šiřitelných nástrojů a aplikací zaměřených na síťovou bezpečnost a zabezpečení obecně. Stejně jako NST 30 je určen především pro živý boot USB, bez nutné spotřeby úložného prostoru, pouze s využitím operační paměti. Jak autoři této distribuce sami uvádějí, je určena pro pokročilé uživatele obdobných systémů. Stejně tak jako výše zmíněné HoneyDrive 3 a NST 30 však nejsou určené pro fungování specificky pro rádiové rozhraní WLAN, IEEE 802.11 [33].

Operační systém	Ubuntu 18.04 LTS	Kali 2019.4	HoneyDrive 3	NST 30	STD
Podpora SW	vysoká	vysoká	střední	nízká	nízká
Zaměření	obecné	pen. test	honeypot	zabezpeč.	zabezpeč.
Náročnost	vysoká	vysoká	střední	nízká	nízká
min. RAM [MB]	2048	2048	1024	1024	512
min. HDD [GB]	20	20	5	5	2
Terminál/GUI	✓/✓	✓/✓	✓/✓	✓/✓	✓/✓
Dokumentace	✓	✓	✓	✓	×
Vývoj OS	✓	✓	✓	✓	×
Licence	zdarma	zdarma	zdarma	zdarma	zdarma

Tab. 4.2: Srovnání popisovaných host OS [30], [31], [32], [33], [34]

Tabulka 4.2 uvádí některé z charakteristických vlastností, popisovaných operačních systémů. Základní vlastnosti jako licence, podpora software a množství instalovatelných doplňků. V neposlední řadě i náročnost operačního systému na dané zařízení.

Aby bylo možné vytvořit honeypot s použitím virtualizovaného OS, ve kterém bude odděleně běžet AP, pro separaci prostředí možnému útočníkovi. To svou architekturou zajistí vysokou bezpečnost celého prostředí. Proto bude testování omezeno pouze na Ubuntu 18.04 LTS a Kali 2019.4. Tyto systémy jako jediné umožňují dostatečnou podporu aplikací pro virtualizaci guest OS a dalšího software pro doplňující funkce jako zachytávání a zpracování datové komunikace WLAN.

Dále jsou popsány možné OS pro realizaci guest OS, který bude fungovat za pomoci virtualizačního software. Na vybraném guest OS bude fungovat AP, které bude sbírat logy, na jejichž základě lze identifikovat některé síťové útoky a nepovolené úpravy konfigurace. Bylo vybíráno z těchto OS:

- RouterOS,
- OpenWRT,
- Lubuntu 18.04.

RouterOS

Systém vyvíjený přímo pro síťová zařízení společnosti MikroTik, běžně směrovače a WiFi směrovače. Systém obdobně jako další vybrané systémy pro guest OS nemá příliš vysoké výpočetní nároky, ty budou bez problémů pokryty. RouterOS je systém postavený na linuxovém jádru a vzhledem k jeho masivnímu nasazení je očekávána i dostačující podpora pro potřebný hardware. Samotný systém vychází s podporou několika architektur - x86, Advanced RISC Machine (ARM), PowerPC (PPC). Stejně tak ve verzi virtuálního obrazu pro běh ve irtualizovaném prostředí. S vývojem tohoto OS přímo pro hardware MikroTik je spojen i licenční poplatek za každou instalaci tohoto software.

Spravovat tento systém lze hned několika způsoby. Je možné využit webového rozhraní, připojení přes terminál, případně pomocí aplikace Winbox, s grafickým rozhraním. Pro případný běh Winboxu v linuxovém prostředí lze využít aplikační rozhraní Wine, ten umožní běh tohoto software, určeného do prostředí Windows.

Při otestování tohoto systému pro funkci AP na guest OS bude nezbytné dodat některé základní softwarové balíčky. Ty jsou s objasněním jejich funkce zobrazeny v tabulce 4.3 [35].

Balíček	Význam
routeros-x86	Kombinovaný balíček pro AMD/Intel PC - obsahuje základní balíčky, viz další uvedené.
system	Základní služby - statické routování, ip adresy, telnet, firewall
wireless	Podpora bezdrátového rozhraní
security	Secure Winbox, IPSEC a SSH
routing	Dynamické routovací protokoly - RIP, BGP, OSPF
dhcp	Dynamic Host Control Protocol, klient a server
advanced-tools	Doplňkové služby - Netwatch, ip-scan, Wake on LAN

Tab. 4.3: Základní balíčky RouterOS pro x86 [35]

OpenWRT 19.07

Platforma je dodávaná přímo pro jednoúčelové směrovače a WiFi směrovače, popisované v kapitole 4.1. Tvůrci OpenWRT doporučují splnění systémových požadavků 128 MB RAM, 16 MB flash a více. Tato podmínka, vzhledem k vybranému hardware pro běh honeypotu, bude nadmíru splněna. Systém je volně šiřitelný, bez dodatečných poplatků. [36].

OpenWRT nabízí pro ovládání a konfiguraci příkazový řádek a webové Grafické uživatelského rozhraní (GUI), takzvané LuCI. Podle zkušeností uživatele a náročnosti konfigurace tak lze využít preferovaného rozhraní. Ve srovnání s výše popisovaným RouterOS je tento systém výrazněji otevřenější a lze v něm provádět významnější zásahy v podobě vkládání různých softwarových balíčků dostupných pro jednotlivé verze tohoto systému v online repozitářích. To ovšem při neuvážené konfiguraci může ohrozit zabezpečení zařízení. Díky široké podpoře hardware a vydávání hned několika typů obrazů systémů je umožněna i potřebná virtualizace. Vzhledem k nižší výpočetní náročnosti a s ohledem na cílový hardware, podporu funkcí pro realizaci AP je i tento systém možným kandidátem pro virtualizované AP, nativně poskytující logování.

Následující seznam zobrazuje některé základní softwarové balíčky a ovladače v OpenWRT, které je nutné dodat pro realizaci konfigurace AP:

- nl80211,
- mac80211,
- hostapd,
- wpa-supplicant,
- dnsmasq.

Pomocí těchto a dalších přídavných balíčků bude prakticky testováno v kapitole 5.2 použití OpenWRT jako guest OS pro stabilní realizaci AP v honeypotu.

Lubuntu 18.04

Posledním vybraným systémem pro realizaci AP v guest OS je Lubuntu 18.04, odvozený od základů Ubuntu podobně jako Kubuntu, Xubuntu a další. Ovšem s podstatně nižšími výpočetními i úložnými nároky pro běh systému. Lubuntu podporuje běžné architektury x86, amd64. Lze jej ovšem jako i další systémy, otevřené ke konfiguračním úpravám, využít k vytvoření prostředí pro AP. Systém je bez licenčních poplatků volně k použití a editaci [34].

Značnou výhodou tohoto OS je její maximální otevřenost k úpravám a široká podpora spojená se základem systému vycházejícího z Ubuntu. Díky tomu je možné

na systém dodat potřebný software i potřebné ovladače dodatečného hardware. Systém lze ovládat standardním GUI, případně se do něj připojit a pracovat s ním pomocí příkazové řádky.

Na základě popisu vybraných guest OS, kdy RouterOS, OpenWRT 19.07 i Lubuntu 18.04 splňují teoreticky stanovené požadavky. Proto bude výběr finálně použitého guest OS pro realizaci AP a logování možné činnosti útočníků v tomto OS otestován prakticky v kapitole 5.2. Během praktického testování bude ověřena podpora potřebných funkcionalit a ovladačů na vybraném hardware, a to při jejich stabilním běhu.

4.3 Software

V závislosti na vybranou architekturu a OS je nutné doplnit toto zařízení vhodnými aplikacemi, či vlastními funkcionalitami formou skriptů. Samotně běžící OS nemůže splňovat jednotlivá kritéria potřebných funkcionalit v honeypotu vzhledem k zadání. Konkrétně funkce a aplikace realizující zachytávání datové komunikace, zpracování těchto dat, vizualizaci vybraných dat a automatizaci běžících funkcí na honeypotu. Nabízí se proto hned několik možností, jaký doplňující software do honeypotu použít. Rozdělit ho lze hned podle několika kritérií:

- Komplexnost,
- licence,
- rozšiřitelnost.

Při využití komplexního řešení lze očekávat vyšší licenční poplatek, stejně jako možná omezení daného produktu, co se jeho rozšiřitelnosti týče. Navíc by taková realizace příliš neodpovídala vlastnímu návrhu honeypotu, pouze instalace již vytvořeného IDS systému, potažmo celého systému honeypotu. I přes tyto skutečnosti jsou níže uvedeny příklady komplexních, často licencí zatížených řešení. Z jejich principu fungování může vzejít inspirace pro vlastní realizaci.

Využití méně komplexního software může na druhou stranu umožnit lepší kooperaci a vlastní zásah do otevřenějšího systému. Často i s příhodnějším licenčním zařazením. Další výhodou může být vyšší podpora potřebných formátů výstupních dat, na rozdíl od celistvějších systémů, které si mohou případná data předávat pouze v rámci svého systému.

Zástupci aplikací, z jejichž principů fungování se při této realizaci lze inspirovat, případně je v rámci jejich licence použít, jsou Cacti, Cron, HoneyD, Specter a Wireshark.

Cacti

Cacti je nástroj nabízející zpracování a vizualizaci datové komunikace, vytížení zdrojů serverů, případně uživatelem vybraných dat předávaných v různých formátech jako comma-separated values (CSV) nebo skriptem. Nástroj je volně použitelný pod OS Windows i různými Unixovými distribucemi, včetně Ubuntu. Standardní metodou sběru dat je využití Simple Network Management Protocol (SNMP). Pro dotazování a ukládání potřebných dat, které jsou následně dostupné v jeho databázi je použit přidružený nástroj Spine. Ten periodicky vysílá SNMP dotazy a ukládá data do databáze. Další variantou obdobného vizualizačního nástroje je Splunk, nabízející podobné možnosti s otevřenou licencí pro určité množství zpracovatelných dat [37].

Vizualizace základních, orientačních informací bude součástí tohoto honeypotu. Nebude ovšem využito takto komplexních nástrojů, které by neúměrně ke svému využití zatěžovaly celý systém. Namísto toho bude využito java scriptové knihovny Highcharts, pomocí níž budou vizualizovány základní informace jako počet autentizací a deautentizací v čase.

Cron

Software implementovaný v linuxových distribucích umožňuje spouštění jednotlivých uživatelských příkazů nebo celých skriptů. Nabízí definování časové periody, po které má žádané skripty automaticky spouštět. Jde o jednoduchý, ale velmi efektivní nástroj pro automatizaci nejrůznějších operací. Běžně může jít o zálohu dat, manipulace s poštou nebo stažení aktualizací systému.

V honeypotu bude hrát tento nástroj klíčovou roli při automatizování monitorování bezpečnostních incidentů, zálohy logů a hlášení bezpečnostních rizik. Výpis 4.1 níže ukazuje příkaz pro výpis cron úloh a následnou konfiguraci pro spouštění daného python skriptu každých 15 minut.

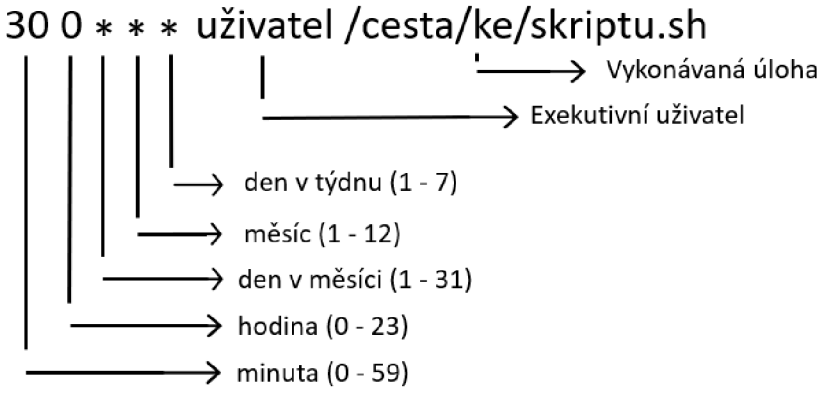
```
tester@Honeypot-AORUS-PRO-WIFI:~$ crontab -l

*/15 * * * * /usr/bin/env python3 /home/tester/wifiPripojeni.py
```

Výpis 4.1: Cron definice pro periodické vyvolávání python skriptu

Každý znak v tomto výpisu má svůj specifický význam v definování doby, kdy má dojít k provedení úlohy. Pro přehlednost je na obrázku 4.2 zobrazen popis s dalším příkladem použití cron úlohy.

Zobrazený příklad cron úlohy na obrázku 4.2, vyvolávající skript je prováděn každý den v 0:30.



Obr. 4.2: Příklad s popisem cron úkolu

Firejail

Aplikace zaměřená na sandboxing jednotlivých procesů. Tímto způsobem je obecně možné vymezit pro útočníka prostřední, ve kterém se může pohybovat a provádět případné úpravy. Zbytek systému je pro něj při správném použití takovýchto prostředků nedosažitelný. Firejail nabízí specifické, předkonfigurované profily, které obsahují různé procesy pro umístění do sandboxu. Případně lze profily nově vytvářet i upravovat. Nástroj je volně použitelný a vyvíjený komunitou. Nabízí podporu pro velké množství linuxových platforem včetně těch vycházející z debianu [38].

Právě pomocí sandboxingu lze vymezit zdroje pro útočníka a nepozorovaně provádět potřebný sběr informací. Nicméně Firejail je v tomto ohledu komplikovaně využitelný, vzhledem k jeho specifickému zaměření na konkrétní funkcionality a ne větší části prostředí. Namísto sandboxu na úrovni nabízené programem Firejail se lze posunout na izolaci útočníka pomocí virtualizovaného OS, běžně dostupného ve WiFi směrovačích. Více informací o virtualizaci je popsnáno v kapitole 4.4

Demonstrativní ukázka spuštěné aplikace v jednoduchém sandboxu s omezenými právy v systému je zobrazen ve výpisu 4.2.

```
root@tester:~$ firejail firefox
Reading profile /etc/firejail/firefox.profile
Reading profile /etc/firejail/disable-mgmt.inc
Reading profile /etc/firejail/disable-secret.inc
Reading profile /etc/firejail/whitelist-common.inc
Parent pid 4772, child pid 4773
Blacklist violations are looged to syslog

Child process initialized
parent is shutting down, bye...
```

Výpis 4.2: Sandbox aplikace Firefox

HoneyD

Multiplatformní, volně šiřitelná aplikace fungující na GNU/Linux/Windows. Nástroj pro simulování několika síťových uživatelů. Virtuální klienty lze upravit podle nároků prostředí, v kterém běží, volbou operačního systému, typu přenášených dat a dalších. Důvěryhodnosti dodává podpora příkazů ping a traceroute na virtuální klienty, HoneyD také poskytuje IDS mechanismus. Vytvořením většího množství virtuálních klientů může sloužit ke zmatení nebo odrazení útočníka. Jelikož by útočník musel analyzovat datovou komunikaci jednotlivých virtuálních klientů a hledat mezi nimi skutečné cílové zařízení. Software umožňuje využití až 65536 IP adres použitelných při simulaci virtuální sítě. Během konfigurace lze definovat ztrátovost, zpoždění a trasy virtuální topologie [39].

Celý koncept Honeyd je inspirací pro vytvoření umělé komunikace přes rádiové rozhraní, které by dodávalo na autentičnosti AP, jakožto běžně uživatelsky vytěžovaného síťového zařízení.

Specter

Nástroj podporující velkou škálu operačních systémů - Windows(98, NT, 2000, XP), různé distribuce linux, i MacOS. Je založený na činnosti IDS. Simulací nedostatečně zabezpečených koncových stanic nabízí možným útočníkům vhodný cíl. Nástroj vytváří pro jednotlivé útočníky separované profily s volbou různého stupně zabezpečení. Tím umožňuje získat více informací od každého z nich. Dále podporuje běžné elektronické služby jako Simple Network Management Protocol (SMTP), ale i služby mailové, pro transport dat a další [40].

Profil	SSID	IEEE 802.11	Zabezpečení	Frekvence [GHz]	Kanál
1	test	802.11 a, ac	WPA/WPA2	5,180	36
2	wlan	802.11 g	WEP	2,447	8
3	FreeWifi	802.11 b	otevřené	2,457	10
4	wifi	802.11 g, n	WPA/WPA2	2,412	1
5	Home	802.11 g	WEP	2,437	6

Tab. 4.4: Přehled použitých profilů AP

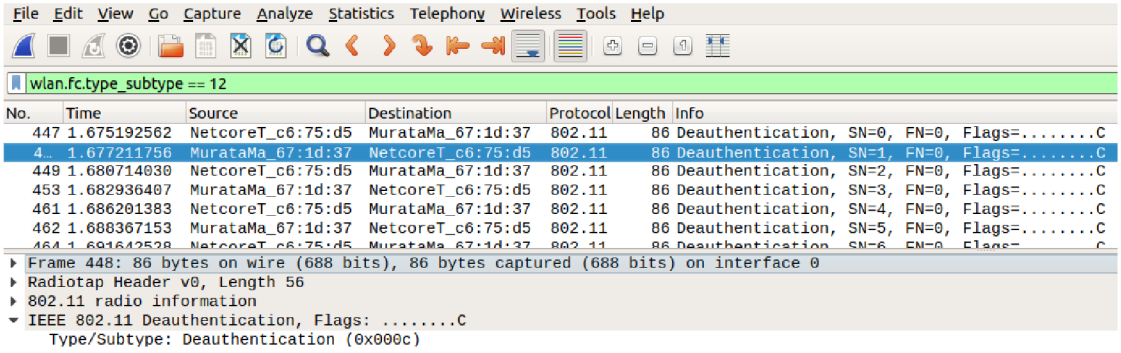
Samotný Specter nebude při vlastní realizaci využit. Nicméně upozornil na zajímavý fakt kategorizace různých zranitelností v rámci definovaných profilů. Tohoto faktu bude využito v rámci profilů AP, jak je zobrazeno v tabulce 4.4. Kdy jednotlivé

profily budou disponovat různou mírou zabezpečení, ale i podporovanými přenosovými protokoly IEEE 802.11. Využito tak bude přenosových kanálů jak v 2,4 GHz, tak 5 GHz volného frekvenčního pásma.

Wireshark

Software volně použitelný, dostupný na většině masivněji využívaných OS, umožňuje zachytávat datovou komunikaci na vybraných síťových rozhraních. Umožňuje zachytávat v promiscuous režimu, kdy ukládá všechna zachycená data (například zachycené přes rádiové rozhraní). Dále nabízí filtrační mechanismy, kdy Wireshark definovanými úložnými filtry třídí a ukládá pouze specifický typ dat nebo následně filtruje zachycená data zobrazovacími filtry a ty uživateli zobrazí [41].

Obdobou Wiresharku, znázorněného s nastaveným zobrazovacím filtrem pro deautentizační rámce IEEE 802.11, na obrázku 4.3 v terminálu je tShark. Program nabízející stejné možnosti jako Wireshark, při stejné syntaxi, ovšem ve zmíněné terminálové podobě. Při použití tSharku lze proces zachytávání a vyhodnocení dat automatizovat případnými skripty. Wireshark, respektive tShark bude v této diplomové práci dále využit pro zachytávání a analýzu dat.



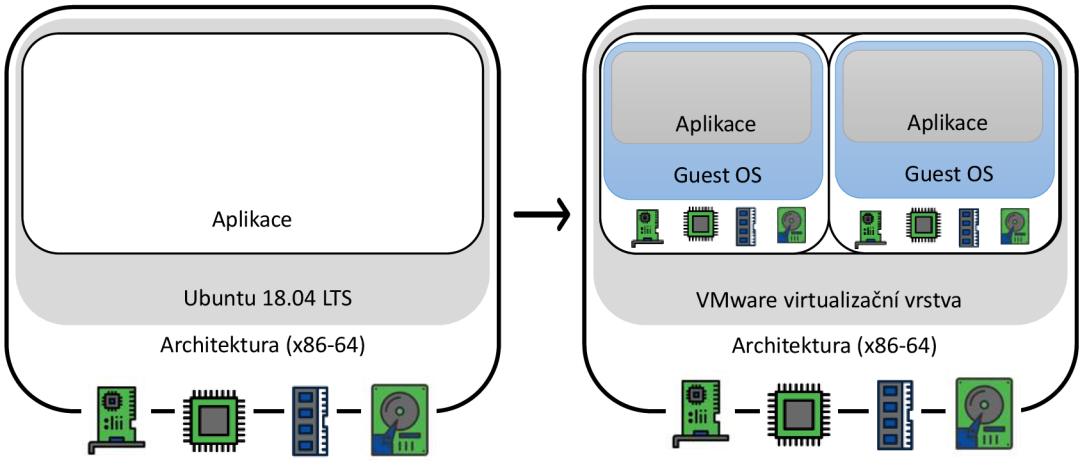
Obr. 4.3: Program Wireshark s nastaveným zobrazovacím filtrem

4.4 Virtualizace

Pomocí virtualizace se vytváří virtuální, respektive simulované prostředí, místo standardního s přímým fyzickým přístupem. V rámci virtualizace jsou počítačem rozděleny, nebo vytvořeny virtuální verze hardwaru. Jako je rozdělení úložných prostředků, operační pamětí, výpočetních kapacit a periferií. Díky tomu je jedno fyzické zařízení rozdělitelné na několik virtuálních počítačů, které podle předem definovaných mezí používají vlastní část přidělených hardwarových zdrojů. Následně každý virtuální stroj může fungovat s odlišným OS, spouštět různé aplikace, zatímco jiné

virtualizované stroje konzumují vlastní přidělené prostředky s cílem vykonávání odlišných úloh [42].

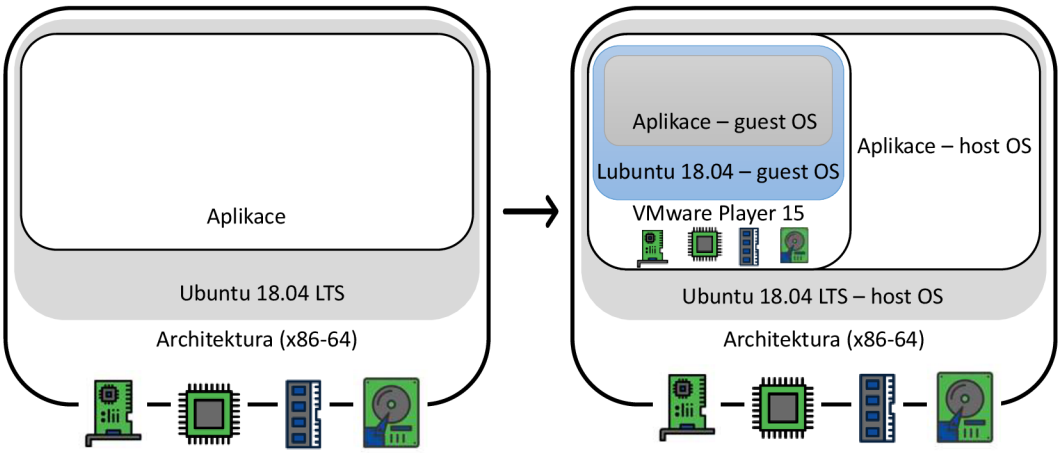
Na obrázku 4.4 je zobrazen jeden z typů virtualizace fyzického zařízení, označované jako realizaci hypervizoru typu 1. Jedná se o nativní virtualizaci, běžně využívanou při virtualizaci serverů, při potřebě vytvoření více než jednoho virtualizovaného OS.



Obr. 4.4: Nativní virtualizace

Možnosti a výhody

Virtualizace přináší hned několik výhod, které lze využít i při realizaci tohoto WiFi honeypotu. Díky virtualizování fyzických zdrojů dochází ke zvýšení škálovatelnosti a lepší schopnost zpracovávat více různých úloh. Díky tomu je snížen faktický počet separátně běžících fyzických zařízení, umožnující nejen pořizovací úsporu, ale i následné zjednodušení uskladnění a snížení spotřeby energie. Při virtualizaci OS je zaručeno i zabezpečení hostovaného OS i samotným aplikacím vykonávající činnost na tomto systému. V případě tohoto honeypotu modelově zpracování dat. Hostovaný hypervizor je zobrazen na obrázku 4.5, známý také pod označením realizace hypervizoru typu 2. Dalšími výhodami je rychlé obnovení systému z uloženého obrazu guest OS včetně veškerého nastavení, s tím i spojené klonování obrazů při rozhodnutí nasazení na více různých zařízení [42].



Obr. 4.5: Hostovaná virtualizace

Specificky pro případ honeypotu, který díky vysoké míře interakce umožňuje vstup útočníka do zařízení, jakožto guest OS, je zde vytvořeno izolované prostředí, ve kterém útočník nabude dojmu, že se nachází přímo na daném zařízení. V případě poškození virtualizovaného prostředí v podobě změn konfigurace, mazání a nahrávání software, pak lze celý obraz obnovit ze zálohy během několika málo minut. Přes různé varianty virtualizace se pro tento úkol hodí zmíněný typ virtuazace hostované, zobrazené na obrázku 4.5. Na hostovaném OS pak mohou běžet potřebné aplikace a skripty, pro útočníka neviditelné, s efektivním využitím pouze jednoho fyzického zařízení. Pro shrnutí jsou zde uvedeny hlavní výhody použití virtualizace při realizaci tohoto WiFi honeypotu v těchto bodech:

- Jednoduchá obnova,
- izolace útočníka,
- jednoduchá obnovitelnost,
- klonování,
- zabezpečení hostovaného OS.

Výběr virtualizačního software

Na základně vybraného hardware a OS lze vybírat virtualizační software, podporující hostovaný OS na bázi linuxu. S ohledem k základním požadavkům při virtualizaci nebude nutné vybírat z licencovaného, a tedy placeného software. Veškeré potřeby spojené s honeypotem pokryjí Oracle VirtualBox, případně VMware Player.

Z počátku lze konstatovat, že obojí distribuce splňují potřebné požadavky pro poskytnutí služeb virtualizace za účelem běhu linuxového OS nakonfigurovaného

jako AP. Výhodou VirtualBoxu může být v sejmutí snapshotu některého z vytvořených obrazů, který je u VMware dostupný až v placené distribuci Workstation, spolu s pokročilou konfigurací síťových rozhraní. I přes tuto nevýhodu byl VMware Player 15 vybrán pro tuto realizaci. To především pro dobrou dokumentaci VMware produktů a stabilitu tohoto nástroje, která je občas VurtualBoxu vyčítána. V tabulce 4.5 jsou vypsány základní parametry, včetně podpory obou aplikací.

Software	Oracle VirtualBox	VMware Player
Typ hypervizoru	2	2
Podporované host OS	Linux, Windows, MacOS, Solaris	Linux, Windows
Podporované gues OS	Linux, Windows, MacOS, Solaris	Linux, Windows, MacOS, Solaris
Podpora 64-bit host OS	ano	ano
Podpora 64-bit guest OS	ano	ano
Podpora AMD-V	ano	ano
Virtualizace	hardware	hardware + software
Podpora USB	ano	ano
Licence	zdarma	zdarma

Tab. 4.5: Základní parametry porovnávaného virtualizačního software [42], [43]

4.5 Skripty

Při provedené volbě využití méně komplexních programů, zaměřených na specifický úkon a nikoli funkci celého honeypotu, případně IDS, je zapotřebí doplnit honeypot vlastními skripty. Těmi je možné přímo ovládat hardware, získávat a zpracovávat naměřená data. Mimo to mají výhodu platformní nezávislosti a dají se obdobně využít i v jiném systému.

Další kritérium, které se vytvořením a nasazením skriptů splní, je automatické vyhodnocení zachycených dat v reálném čase, možnost provedení případných opatření a optimalizace konzumovaných zdrojů v podobě obsazení úložné paměti. V kombinaci s Cronem pak bude možný běh v nekonečné smyčce po požadovanou dobu.

Od skriptů použitých u této realizace jsou požadovány provedení především těchto úkolů:

- Ovládání PCIe WiFi na host OS,
- nahrávání a stahování dat z guest OS,
- zachytávání dat pomocí tShakru na host OS,
- úprava iptables, hostapd na guest OS,
- zpracování a vyhodnocení zaznamených dat,
- záloha logů a zachycených dat,
- vizualizace základních, zpracovaných dat,
- notifikace správce při detekci útoků.

Pokud by mělo řešení potřebných úkolů zůstat pouze v rámci shell skriptů, rychle by se mohly stát příliš komplikovanými. V některých případech by nedokázaly splnit potřeby vzhledem k absenci některých knihoven a komplikovanější manipulací s některými daty, které lze u běžných programovacích jazyků očekávat implementované. Vhledem k tomu nebudou skripty vymezeny pouze na shellové, ale budou vytvářené ve vyšším programovacím jazyku. Ten bude se shellovými příkazy často provázán a bude jich využito.

Python

Vhodným skriptovacím, programovacím jazykem splňující všechny požadavky je Python. Na většině hromadně známých linuxových distribucích je Python již předinstalován, a je tak možné s ním hned pracovat. Dále nabízí jednoduchou, dobře čitelnou syntaxi a obsahuje velké množství oficiálně vydaných knihoven, stejně jako tisíce knihoven třetích stran.

Při výběru verze Pythonu, tedy rozhodnutím mezi verzí 2.7 a 3.5, bylo vybráno na základě několika kritérií. Již několik let se přechází na Python 3, pro který jsou dostupné nové knihovny, stejně tak jako velká část těch ze starší verze Python 2.

Vzhledem k tomu, že se v tomto případě nenavazuje na jiný kód, který by byl již vytvořený v některé verzi a Python 3 obsahuje všechny potřebné knihovny, byl vybrán právě Python 3. Na tuto verzi navíc přechází další software s možným potenciálním použitím jako podpora Pythonu 3 v RouterOS, který je jedním z testovaných OS pro funkci AP ve virtualizovaném guest OS.

5 Experimentální konfigurace

V rámci výběru OS pro roli host a guest byla provedena experimentální konfigurace s praktickým srovnáním potřebných vlastností jednotlivých systémů. Testovány byly teoreticky popsané a vhodné OS. Hlavní pozornost u testování pro host OS byla kladena především na tyto body:

- Podpora potřebných ovladačů a hardware,
- podpora software pro zachytávání a manipulaci s daty,
- robustnost pro běh virtualizace guest OS,
- možnost vytvářet a používat skripty - otevřenost systému.

Testování je spolu s praktickým ověřením popsáno v podkapitole 5.1. Dále byla realizována instalace a základní konfigurace jednotlivých kandidátů na guest OS. Tento systém měl splňovat zcela odlišná kritéria než host systém. Shrnutí základních požadavků při experimentální konfiguraci je v těchto bodech:

- Konfigurace AP,
- podpora potřebného hardware,
- zprovoznění základní komunikace AP - uživatel,
- lokalizace a extrakce logů.

5.1 Testování a výběr hostovacího OS

Vzhledem k dosavadnímu nedodání objednaného hardware byla provedena prvotní konfigurace a výběr, zahrnující otestováním a ověřením potřebných funkcionalit host OS dvěma z několika možných způsobů:

- Virtualizace hypervizorem typu 2,
- Dual-boot.

Testování virtualizací hypervizorem typu 2

Prvotní konfiguraci bylo možné realizovat pomocí virtualizačního software. Kdy bylo testování provedeno jak s Oracle VirtualBox, tak VMware Playeru. Ovšem v obou případech došlo k problému s nekompatibilitami a špatně se mapujícími prostředky, konkrétně PCIe WiFi kartou. Ta je implementována na základní desce, jako u zařízení použitého při testovaní OS, tak i v zakoupeném hardware honeypotu. Při potřebě namapovat WiFi zařízení se zachováním funkcionalit pro vytvoření AP, včetně dodatečného ovládání, musí být WiFi karta připojena přes rozhraní USB. To

lze namapovat do virtualizovaného OS se zachováním potřebných vlastností. Tento fakt má vliv na finální konfiguraci a použitý hardware. Pro běh AP v guest OS musí být použit WiFi adaptér připojený přes USB rozhraní.

Na základě tohoto zjištění byla dokoupena pro funkci AP v guest OS WiFi karta s USB rozhraním, konkrétně Netis AC1200 Dual Band. S podporou AP režimu v běžném frekvenčním pásmu 2,4 GHz i 5 GHz pásmu a přenosových protokolů až IEEE 802.11ac. Tento USB WiFi adaptér, ve verzi WiFi 5 už ovšem neumožňuje MU-MIMO. Vzhledem k nedostatečnému počtu antén a chybějící podpoře IEEE 802.11 ax, které je dostupné až ve verzi WiFi 6.

PCIe WiFi karta, integrovaná na základní desce, sice splňuje podmínky podpory AP režimu, nejde ji však vzhledem k výše popsanému problému propagovat do guest OS. Proto bude nasazena pro monitorování a skenování v režimu klienta, umístěného v host OS.

Testování při dual-boot

Druhou zvolenou variantou srovnání hostovacího OS bylo pomocí duálního bootování. Po volbě systému v GNU GRand Unified Bootloader (GRUB) zavaděči jsou poskytnuty veškeré nástroje a prostředky použitého zařízení. Zároveň lze ověřit možné problémy s namapováním konkrétních periferií, nekompatibilitami ovladačů, či jejich úplnou absenci pro konkrétní OS.

Po odhalení komplikací s mapováním hardware do guest OS při testování na základě použití virtualizace OS a vyvození opatření pro další konfigurace honeypotu byla provedena i druhá varianta testování. Kdy byly jednotlivé distribuce porovnány a následně vybrána finální varianta.

Vybrané host OS z kapitoly 4.2 Ubuntu 18.04 LTS a Kali 2019.4 svými vlastnostmi teoreticky pokrývají všechny nezbytné oblasti zahrnující podporu funkcionalit, stabilitu systému, tak i podporu hardware a potřebných ovladačů. Tyto vlastnosti byly u těchto systému testovány a porovnány.

Na obou systémech byla testována PCIe WiFi, implementovaná na základní desce a možnosti manipulace s ní. V rámci tohoto testování bylo vytvořeno AP. Ve výpisu 5.1 je zobrazena část příkazů použitých při konfiguraci AP.

```
ifconfig wlan0 up 192.168.15.1 netmask 255.255.255.0
route add -net 192.168.15.0 netmask 255.255.255.0 gw 192.168.15.1

iptables --table nat --append POSTROUTING --out-interaface eth0 -j
MASQUERADE

iptables --append FORWARD --in-interface wlan0 -j ACCEPT

sudo apt-get install hostap
mkdir /root/pristupovyBod
sudo nano /root/pristupovyBod/hostapd.conf
```

Výpis 5.1: Část konfigurace při testování možností OS

Tato konfigurace otestovala možnosti balíčků pro vytvoření AP, nastavení síťových rozhraní a iptables. Tím byla ověřena možnost nastavit AP u linuxových, konkrétněji debianových distribucí. Díky tomu bylo možné dále testovat zachytávání provozu a manipulaci s těmito daty. Na tuto základní konfiguraci AP bude navázáno při konfiguraci Lubuntu 18.04, spuštěného ve virtualizačním nástroji.

Mimo to byly instalovány nástroje pro vytváření, úpravu a spouštění skriptů v jazyce Python, virtualizaci potřebného guest OS a zachytávání síťového provozu. Konkrétně PyCharm, VMware Player a Wireshark, respektive terminálovou verzí tShark. Oba operační systémy, se srovnatelnými nároky na hardware, disponovaly schopností pracovat se zmíněnými a případně i obdobnými nástroji. Ve výpisu 5.2 je znázorněno zachycení EAPOL rámců při testování alternativního TCPdump k Wiresharku, respektive tSharku. Zachycení shodných dat je pro srovnání zobrazen pomocí GUI Wiresharku, zobrazen na obrázku 5.1

```
root@tester:~# tcpdump -i wlan0

tcpdump: verbose output suppressed, use -v or -vv for full protocol
decode

listening on wlan0, link-type EN10MB (Ethernet), capture size
262144 bytes

09:59:57:699603 EAPOL key (3) v2 len 95
09:59:57:703727 EAPOL key (3) v1 len 117
09:59:57:704035 EAPOL key (3) v2 len 151
09:59:57:715139 EAPOL key (3) v1 len 95

09:59:57:715992 d0:df:9a:83:81:55 (oui Unknown) > Broadcast Null
Unnumbered, xid, Flags [Response], length 6: 01 00

09:59:57:754339 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP,
Request from d0:df:9a:83:81:55 (oui Unknown), lenght 300

09:59:57:754426 IP 0.0.0.0.bootpc > 255.255.255.255.bootps: BOOTP/DHCP,
 Request from d0:df:9a:83:81:55 (oui Unknown), lenght 300
```

Výpis 5.2: Zachycené připojení klient-AP programem TCPdump

	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	LiteonTe_83:81:55	Broadcast	XID	20	Basic Format; Type 1 LLC (Class I LLC); Wi
2	0.000331010	IntelCor_99:c5:cd	LiteonTe_83:81:55	EAPOL	113	Key (Message 1 of 4)
3	0.005349765	LiteonTe_83:81:55	IntelCor_99:c5:cd	EAPOL	135	Key (Message 2 of 4)
4	0.018487817	IntelCor_99:c5:cd	LiteonTe_83:81:55	EAPOL	175	Key (Message 3 of 4)
5	0.022794386	LiteonTe_83:81:55	IntelCor_99:c5:cd	EAPOL	113	Key (Message 4 of 4)
6	0.069771577	LiteonTe_83:81:55	Broadcast	ARP	42	Who has 192.168.15.1? Tell 192.168.15.39
7	0.069786661	IntelCor_99:c5:cd	LiteonTe_83:81:55	ARP	42	192.168.15.1 is at e0:94:67:99:c5:cd
8	0.070593820	LiteonTe_83:81:55	Broadcast	ARP	42	Who has 192.168.15.1? Tell 192.168.15.39
9	0.095348089	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0xa0382ab8
10	0.095359686	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0xa0382ab8
11	0.131945111	192.168.15.1	192.168.15.39	DHCP	354	DHCP ACK - Transaction ID 0xa0382ab8

Obr. 5.1: Zachycené připojení klient-AP programem Wireshark

Rozdíl, poukazující na určité nedostatky jednoho z testovaných systému, a to Kali 2019.4 byla občasná nestabilita AP. Spolu s tím byly spojené i komplikace s podporou WiFi karty při testování na náhradním hardware. Proto bylo následně testování opakováno na daném hardware WiFi honeypotu. V případě Kali se nepodařilo stabilně zprovoznit jak s doporučenými ovladači WiFi karty, stejně jako dalšími, obecněji zaměřenými ovladači. Nestabilita, občasné výpadky, WiFi zařízení bylo pozorováno i při monitorování datového provozu, nejen při běhu AP. To byl jeden z klíčových faktorů, proč Kali 2019.4 zůstal pouze jako penetrační nástroj pro

testování detekce útoků a zatěžování honeypotu. Níže zobrazená tabulka 5.1 obecně shrnuje testované body u obou systémů.

Operační systém	Ubuntu 18.04 LTS	Kali 2019.4
Podpora HW	✓	✓
Ovladače a balíčky	✓	✕
Python 3	✓	✓
VMware Player	✓	✓
Wireshark	✓	✓

Tab. 5.1: Prakticky testované okruhy u Kali 2019.4 a Ubuntu 18.04 LTS

5.2 Testování a výběr guest OS

Dále proběhlo testování možných guest OS pro realizaci AP, běžící virtualizovaně ve VMware Playeru. Požadavky, které vybraný guest OS musel splňovat byly podpora dlouhodobě a stabilně běžícího AP, s potřebnými službami Dynamic Host Configuration Protocol (DHCP) a Domain Name System (DNS). Při podpoře zvoleného hardware, s dobře fungujícími ovladači, dostupnými logy ze zařízení a schopností jednoduše překonfigurovat vlastnosti AP.

RouterOS

Po instalaci RouterOS, spolu s potřebnými základními balíčky, proběhla základní konfigurace pro zprovoznění AP. Konkrétně zajištění přidělování IP adres spolu s informacemi o výchozí bráně pomocí DHCP serveru a překlad doménových jmen přes DNS. Po zprovoznění těchto služeb bylo zapotřebí integrovat do RouterOS samotné rádiové rozhraní pro AP.

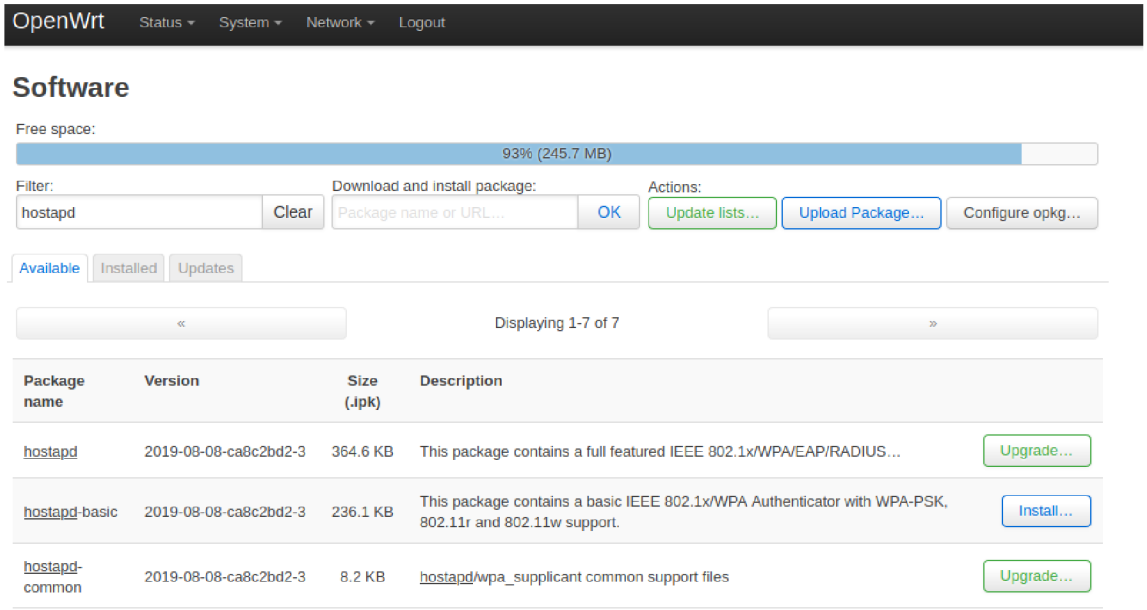
Navzdory faktu, že tento systém vychází z linuxovém základu, je velmi uzavřený a neumožňuje přílišné zásahy, jako doinstalování ovladačů třetích stran, dalších aplikací. Tím pádem chyběla potřebná podpora USB WiFi karty, která s čipem RTL8812AU a potřebným stejnojmenným ovladačem nebyla kompatibilní. Obdobně dopadla i integrovaná PCIe WiFi karta na základní desce honeypotu. Ta se do RouterOS sice namapovala, ale pouze jako ethernetové, nikoliv rádiové rozhraní.

Při nezbytné potřebě použít RouterOS jako guest OS s USB WiFi kartou, by bylo zapotřebí použít starší versi RouterOS, konkrétně nižší než 6.0. Kde byly externí karty využitelné, a to pouze se specifickými, podporovanými čipy Atheros, např AR9580, AR9380. V dnešní době, bez nadsázky, už zastaralým hardware. Ten by

nesplňoval nároky na AP, fungujícím v dvou základních pásmech s dostatečnou podporou přenosových protokolů IEEE 802.11. Mimo to je RouterOS licencován, tedy by musel být zakoupen a při každém přemazávání obrazu ve VMware Playeru (například při poškození útočníkem) by musel být manuálně vkládán licenční kód pro aktivaci daného software ID nahraného obrazu systému.

OpenWRT

Jakmile byl OpenWRT nainstalován, proběhla jeho základní konfigurace pomocí terminálu dostupného přes spuštěný obraz ve VMWware Playeru. Tedy konfigurace IP adresy a výchozí brány na ethernetovém rozhraní, respektive rozhraní vmnet8. Následně je na definované IP adrese dostupné webové rozhraní LuCI. Pomocí tohoto rozhraní byly jednodušeji a přehledněji hledány, stahovány a instalovány softwarové balíčky a ovladače hardware, jak je ukázáno na obrázku5.2.



Obr. 5.2: Webové rozhraní LuCI u OS OpenWRT 19.07

Podobně jako v případě RouterOS, i u OpenWRT nastal problém s podporou hardware, konkrétně WiFi. Na základní desce integrovaná PCIe WiFi karta nebyla v tomto systému vůbec detekována, respektive namapována. Pro USB WiFi kartu byly nalezeny hned 3 možné ovladače. Jeden z nich, ovladač - RTL8812AU, kartu úspěšně namapoval. Poté byla vytvořena jednoduchá konfigurace AP, i klienta této USB WiFi karty.

Jediný ovladač, za pomocí kterého byla karta namapována byla zároveň uvedena do vysoce nestabilního stavu. USB WiFi karta se krátce po vložení konfigurace odpojovala a zařízení nebylo schopné kontinuálně provozovat AP. V případě testování monitorovacího režimu zařízení naskenovalo jen některé okolní sítě, a to s neúměrně dlouhou časovou prodlevou.

Bylo otestováno hned několik verzí OpenWRT, jak starší 15.05 i naopak v době testování nejnovější 19.07. Při nasazení různých verzí se však problémy spojené s namapováním USB, ani PCIe WiFi nevyřešily. Proto bylo od tohoto systému, jako možnosti guest OS odstoupeno.

Lubuntu

Poslední experimentálně otestovanou variantou byl OS Lubuntu. Po instalaci byly staženy a nainstalovány ovladače pro PCIe WiFi, RTL8812AU. Po úspěšném namapování hardware PCIe WiFi byl stažen a nakonfigurován hostapd, isc-dhcp-server, iptables a další. Tím bylo dosaženo stabilně běžícího AP. Vzhledem k dobře fungující konfiguraci, která bude použita i ve stabilní verzi honeypotu, je níže v několika bodech popsán základní proces instalace a konfigurace balíčků software, pro zprovoznění AP:

- Hostapd,
- isc-dhcp-server,
- persistent-iptables.

Prvním krokem bylo ověření dostupnosti a stažení vybraných softwarových balíčků, potřebných pro konfiguraci AP.

```
sudo apt-get update
sudo apt-get install hostapd
sudo apt-get install isc-dhcp-server
sudo apt-get install iptables-persistent
```

Pro nastavení hostapd je nezbytná úprava konfiguračního souboru, umístěného v `/etc/hostapd/hostapd.conf`. Základní položky potřebné nastavit jsou zobrazeny v tabulce 5.2. Následně byla na tento soubor poukázána cesta úpravou položky `DAEMON_CONF="/etc/hostapd/hostapd.conf"`v souboru `/etc/default/hostapd`.

Poté proběhlo nastavení isc-dhcp-serveru, upravením položek v souboru `/etc/default/isc-dhcp-server`, definujících rozhraní, které mají poskytovat DHCP služby. Následně byl vytvořen IP rozsah 192.168.10.024, určen pro přidělování IP, pomocí DHCP serveru a celkové fungování AP. Spolu s nastavení DHCP serveru byla nastavena i služba DNS.

interface	Rádiové rozhraní sítě
ssid	Název dané WiFi sítě
hw_mode	Přenosový standard IEEE 802.11
channel	Rádiový kanál ISM/(UNII u 5GHz)
wpa=2	Generace WPA protokolu
wpa_passphrase	Uživatelské heslo k WiFi síti
wpa_key_mgmt	Typ ověření
wpa_pairwise	Šifrovací algorimus párovacího klíče

Tab. 5.2: Základní konfigurační parametry obsažené v hostapd.conf

```
subnet 192.168.10.0 netmask 255.255.255.0
{
range 192.168.10.10 192.168.10.99;
option domain-name-servers 8.8.8.8, 8.8.4.4;
option routers 192.168.10.1;
}
```

Následovně byla upravena konfigurace iptables a uložena pomocí iptables-persistent, která umožní její zachování i po restartu zařízení.

```
sudo iptables -t nat -A POSTROUTING --out-interface eth0 -j MASQUERADE
sudo iptables -A FORWARD --in-interface wlan0 -j ACCEPT

sudo service netfilter-persistent start
sudo su iptables-save > /etc/iptables/rules.v4
```

Realizace prvotní, experimentální konfigurace, během které byl vybírán host a guest OS, potvrdila možnost použití některých z navržených variant teoreticky popsaných a vybraných systémů. Jako nejvhodnější varianta pro host OS po všech srovnáních a testování byl vybrán Ubuntu 18.04 LTS. Tento OS splňoval podporu pro veškerý potřebný hardware, stejně jako ovladače i potřebný software.

Vybraný guest OS byl Lubuntu 18.04, používanými příkazy podobný, v mnoha případech identický s Ubuntu 18.04, pouze v méně na zdroje náročné verzi. Lubuntu 18.04 splňoval všechny požadavky na AP a zajistil jeho stabilní běh. Tedy podporu hardware s vhodnými ovladači, dostupný software a nativní podporou logování. Ta je dostupná přímo pro hostapd, ale i kernel a autentizaci.

6 Stálá konfigurace

V rámci experimentální konfigurace proběhl výběr pro použití Ubuntu 18.04 LTS jako host OS. Na něm operuje mimo další i VMware Player 15.5 umožňující virtualizaci Lubuntu 18.04 pro guest OS s konfigurací AP. To při ověřené dostupnosti všech potřebných funkcionalit a software pro host i guest OS. Následně přešla měnící se, experimentální konfigurace do stavu konfigurace stálé, kdy byly přidány funkce, o kterých tato kapitola pojednává.

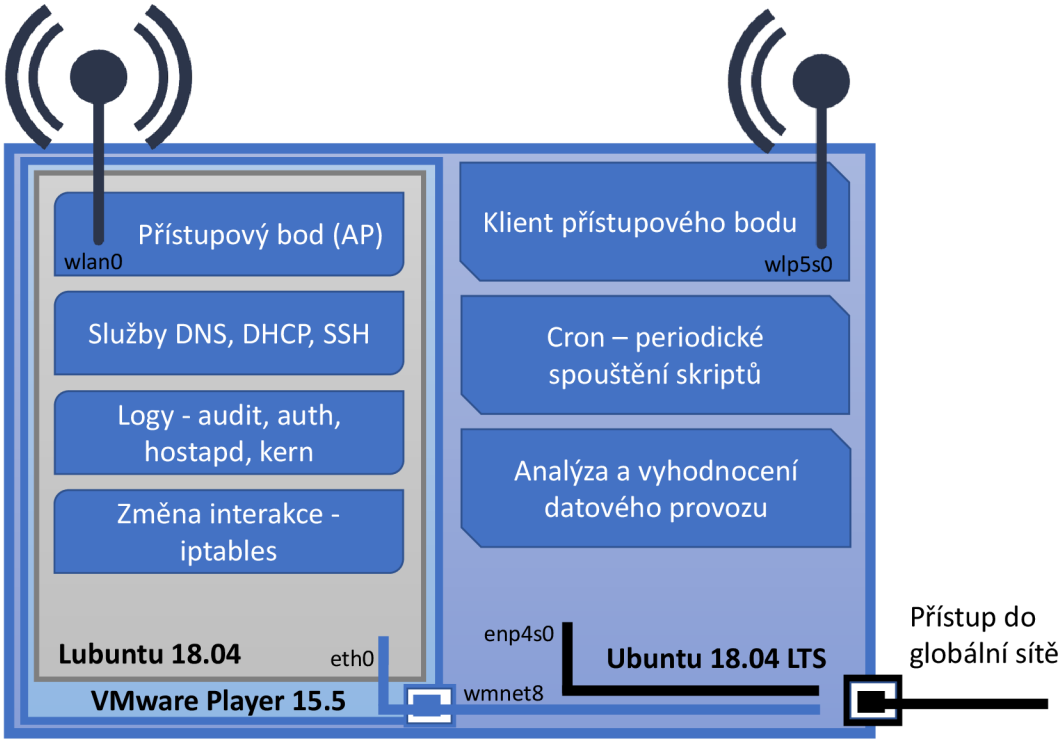
Počínaje definováním nastavení pro zabezpečení zařízení z pohledu možného proniknutí útočníka z rádiového rozhraní do AP, a i dále do honeypotu. Následné vytvoření skriptů, pro monitorování rádiového rozhraní, zpracování a analýzu těchto dat. To i s jejich vyhodnocením, přes testování detekce a zatížení jednotlivými detekovanými útoky po optimalizaci chodu celého zařízení. Mimo to byla vytvořena možnost volby interakce WiFi honeypotu, respektive AP, spolu se změnou konfiguračních profilů AP, prováděné periodicky přes Cron.

6.1 Architektura

Finální architektura honeypotu musela splňovat hned několik kritérií a předpokladů, díky kterým má zařízení plnit očekávané úkony. Při výběru řešení se separací OS, tedy pomocí virtualizace, který běží na AP a figuruje jako WiFi směrovač je dosaženo vzhledu reálného, samostatného prostředí. To vše bez detekování služeb, hardware a skriptů vykonávajících vyhodnocení chování uživatelů AP.

Dále je možné jednoduché obnovení útočníkem poškozené konfigurace AP, včetně její zálohy pro pozdější vyhodnocení kroků útočníka. S tím je spojena izolace tohoto prostředí od zbytku systému, které je útočníkovi nepřístupné a nemůže dojít k poškození zálohovaných dat, jakožto konfigurace host OS a skriptů, které jdou v podobně vybrané linuxové distribuce také vcelku jednoduše zálohovat. Více o docílení zabezpečení na host OS, respektive honeypotu je popsáno v kapitole 6.2.

Na obrázku 6.1 je zobrazeno rozložení služeb mezi guest a host OS. Tak aby byly splněny výše popsané body. Zajištění autentického prostředí AP, které svou izolací od zbytku systému jasně vytyčí prostor, v kterém je útočníkovi umožněno se pohybovat.



Obr. 6.1: Architektura finálního honeypotu

V host OS operuje PCIe WiFi karta, využívána skripty, jak popisuje kapitola 6.3. Ty jsou periodicky spouštěny nástrojem Cron, díky tomu může zařízení běžet nepřetržitě, bez ukončení smyčky a přetečení případného čítače.

Součástí periodicky volaných skriptů jsou i analýza a vyhodnocení datového provozu, který je monitorován. Vzhledem k architektuře je realizováno i připojování se mezi host a guest OS, za účelem úpravy konfigurace AP, stažení logů na guest OS, či pro změnu interakce.

6.2 Zabezpečení

Aby bylo zajištěno vysoké zabezpečení finálního řešení honeypotu i v situaci různé interakce, byla vybrána architektura popsaná v kapitole 6.1, oddělující prostředí. Na prostředí dostupném útočníkovi, od části systému, v němž probíhá zaznamenání, zpracování a vyhodnocení dat. Při vysoké interakci může útočník volně vstoupit z rádiového rozhraní wlan0 na AP, respektive do guest OS.

I přes izolované prostředí, které poskytuje samotný hypervizor VMware Player, je pravidelně probíhající interakce mezi host a guest OS možným bezpečnostním rizikem. Proto byla nástrojem `iptables` v host OS omezena komunikace přes rozhraní

vmnet8, což je rozhraní interagujícího mezi guest a host OS.

Je dosaženo poskytování průchodu datové komunikace mezi AP a veřejnou sítí, povolením logického kanálu forward. Připojení se z guest OS na host OS je zamezeno v logickém kanále input. Sestavovat spojení mezi host a guest OS, pro ovládání AP a kopírování logů z guest OS však není dotčeno. Jako další, preventivní opatření je SSH daemon, pomocí kterého interakce mezi systémy probíhá zapnut v host OS pouze ve chvíli, kdy je realizován přenos dat. V guest OS je SSH daemon zapnut trvale, vždy při startu systému guest OS.

6.3 Skripty

Jsou hojně využity skripty spojující potřebné linuxové a Python3 příkazy. Linuxové pro manipulaci s hardware honeypotu, jako PCIe WiFi karta, nebo realizují kopírování souborů a připojování se mezi guest a host OS. Python3 skripty data získané pomocí linuxových příkazů zpracovávají a vyhodnocují. Hlavní část fungování honeypotu je realizována pomocí skriptů a je zobrazena na obrázku 6.2.

Obr. 6.2: Zjednodušené znázornění činnosti WiFi honeypotu realizované skripty

Monitorování

Honeypot potřebuje k vyhodnocování možných síťových útoků data. Ta podrobuje analýze a na základě jejich obsahu jsou pak vytvářena další akce. Data jsou obstarávána během opakujícího se cyklu, který periodicky vyvolává nástroj Cron. Zobecněné znázornění 6.3 ukazuje která data jsou pomocí PCIe WiFi modulu ovládaného z host OS zaznamenávána a zpracovávána.

Obr. 6.3: Základní činnost PCIe WiFi v host OS

Skenování probíhá vždy na počátku 15 minut trvajícího cyklu. Je realizováno pomocí nástroje `nmcli`, ovládaného z příkazové řádky. Na základě naskenovaných informací v celém vysílacím spektru, určeném pro bezlicenční WiFi sítě, jsou zpracovány statistické informace o počtu naskenovaných WiFi sítí v okolí, nejběžnějším zabezpečení, ale i detekce možného beacon (SSID) útoku. Příkladem kombinovaného využití linuxového a pythonového kódu je zpracování dat linuxovými příkazy, které ovládají samotný hardware a jsou spouštěny jako součást pythonového skriptu. Získávají a předávají data pro následovné zpracování samotným pythononem. To je zobrazeno ve výpisu 6.1.

```
naListWifi = 'nmcli -f SSID,SIGNAL,SECURITY device wifi'
wifiVysledek = os.popen(naListWifi)
wifiList = list(wifiVysledek)

if len(wifiList) > 1:
```

Výpis 6.1: Příklad zpracování naskenovaných dat

Monitorování, pomocí kterého je detekováno největší množství síťových útoků tvoří ve stabilní verzi honeypotu převážnou dobu vytížení PCIe WiFi karty. Monitorování probíhá pouze na konkrétním kanálu, na němž aktuálně funguje AP na guest OS - kanál je určen vždy podle aktuálně používaného profilu AP. Před a po každém monitorování, které je realizováno terminálově řízeným nástrojem tShark, probíhá přepínání PCIe WiFi karty do a z monitorovacího režimu. Ten umožňuje zachytávání dat v promiskuitním režimu. Ve výpisu 6.2 je zobrazena konfigurace zachytávání dat pomocí zmíněného terminálového nástroje tShark a změna režimu PCIe WiFi karty.

```
os.system('echo␣{}|␣sudo␣-S␣-k␣ifconfig␣{}␣down'
  .format(self.sudoPass, wifiKlientRozhrani))

os.system('echo␣{}|␣sudo␣-S␣-k␣iwconfig␣{}␣mode␣monitor'
  .format(self.sudoPass, wifiKlientRozhrani))

os.system('echo␣{}|␣sudo␣-S␣-k␣ifconfig␣{}␣up'
  .format(self.sudoPass, wifiKlientRozhrani))

os.system('echo␣{}|␣sudo␣-S␣-k␣iwconfig␣{}␣channel␣{}'
  .format(self.sudoPass, wifiKlientRozhrani, aktivniKanal))

os.system('echo␣{}|␣sudo␣-S␣-k␣tshark␣-i␣{}␣-w␣{}␣-a␣duration:{}'
  .format(self.sudoPass, wifiRozhrani, jmenoZaznamu, delkazaznamu))
```

Výpis 6.2: Příklad přípravy a zachytávání dat pomocí nástroje tShark

Analýza dat

Jakmile jsou data zachycena a předána ke zpracování, jsou v několika cyklech za pomocí tSkarku a zobrazovacích filtrů rozdělena. Ukládají se k dalšímu přepočítání a zpracování do textových souborů vždy podle požadovaného typu dat, jako jsou pouze beacon rámce, nebo deautentizační, či autentizační rámce.

V případě zálohy dat jsou zálohovány i tyto textové soubory obsahující výpis časové známky, zdrojové MAC adresy, zdrojové IP adresy, cílové MAC adresy a cílové IP adresy, spolu s informačním polem. Pod jednotlivými záznamy je seznam kombinací zdrojových a cílových adres, respektive informativní seznam udávající množství, kolikrát byly tyto rámce z konkrétních síťových adres zasílány.

Detekce hrozby

Při analýze dat dochází k vyhodnocení, zda tyto data mají charakter možného útoku. U každé monitorované kategorie útoku se hledí na prahové hodnoty jednotlivých rámců v čase, které jsou kategorizovány jako běžný provoz, a pak následně jako data, která lze považovat za intruzivní, či zdroj možného síťového útoku. Níže uvedený seznam znázorňuje aktuálně nastavené prahové hodnoty pro vyhodnocení analyzovaných dat, jako obsahující možný útok:

- Deautentizační rámce - více než 200 při 120 vteřin dlouhém záznamu,
- autentizační rámce - více než 200 při 120 vteřin dlouhém záznamu,
- nárůst monitorovaných SSID o 5 vůči průměru monitorovaných SSID,
- nárůst naskenovaných SSID o 10, vůči průměru skenovaných SSID,
- ARP duplikátní zprávy 1 a více při 120 vteřin dlouhém záznamu,
- poměr TCP SYN a TCP SYN, ACK 4:1 a vyšší pro TCP SYN s počtem 500 a více TCP SYN při 120 vteřin dlouhém záznamu.

Na základě opakovaného testování a optimalizace byla stanovena hraniční hodnota zachycených deautentizačních i autentizačních rámců na hodnotu 200 a více při 120 vteřin dlouhém cyklu. S ohledem na možné opakování vysílání této zprávy je tato hodnota dostatečně vysoká, aby nedocházelo k detekování falešných útoků. Zároveň není příliš vysoká a již při několika jednotek vteřin deautentizačním, respektive autentizačním útoku - v závislosti na konfiguraci penetračního nástroje dojde k detekci útoku.

Monitorované SSID, jsou SSID odposlechnuté z vysílaných beacon rámců na monitorovaném kanálu, jenž odpovídá kanálu používaném i AP běžícím v rámci honeypotu. Ty jsou průměrovány po 10 měření a musejí nárazově zaznamenat navýšení o 5 nových SSID, aby byla situace vyhodnocena jako beacon (SSID) záplava na základě monitorování. Druhou variantou je vyhodnocení beacon (SSID) záplavy na základě skenovaných SSID, ty jsou naskenovány vždy na začátku monitorovacího cyklu na všech kanálech - nikoli jeden konkrétní jako v případě monitorování. Pokud je naskenováno o 10 více SSID, než je průměrná hodnota posledních 10 skenování. Pak je situace vyhodnocena jako beacon (SSID) záplava na základě skenování.

Pro detekci ARP spoofingu, který je možné detekovat zachycenou ARP duplicitou, byla stanovena hodnota detekce možného ARP spoofingu při hodnotě 1 a více při 120 vteřin trvajícím cyklu. Tyto zprávy nejsou běžně vysílány a lze proto stanovit prahovou hodnotu takto nízkou.

TCP SYN záplava je detekována ve chvíli, kdy je celkový počet TCP SYN rámců je roven, nebo převyšuje množství 500 při 120 vteřin dlouhém cyklu. Dále pak těchto zpráv musí být v poměru k TCP SYN, ACK 4:1 a více. Tento vysoký nepoměr jasně

indikuje záplavu TCP SYN rámců. Při testování tohoto útoku se poměr pohyboval 9:1, takže stanovená hodnota 8:2 umožňuje širší záběr při detekování tohoto útoku.

Záloha dat

Aby bylo možné zachovat data, obsahující záznam probíhajícího útoku, případně logy z AP, které mohou být dále zpracovány pomocí nástroje Auditd, nebo v případě .pcap logů z monitorování rádiového prostředí zpětně prozkoumány například pomocí Wiresharku probíhá jejich záloha. Ta je rozdělena na dvě kategorie.

První kategorie, automatické zálohování dat, které vytváří ve složce logy-zaloha novou podsložku obsahující v názvu typ útoku spolu s časovou známkou pro jednoduchou orientaci. Tato složka pak obsahuje .pcap zachycená data a vyfiltrované data, uložené v textovém dokumentu, obsahující například při deautentizačním útoku pouze informace spojené s deautentizačními rámci. To znamená počet těchto rámců v daném časovém úseku a zdroj s cílem těch rámců.

Druhou kategorií, uživatelské zálohy dat jsou myšleny zálohy prováděné správcem při práci se zařízením. Těmi může být standardně stažení aktuálních logů z honeypotu, které se jinak periodicky stahují se změnou AP profilu, ve 12:00 a 0:00 každý den. Pro okamžité vyhodnocení však může uživatel honeypotu vyžadovat aktuální data. Ty lze stáhnout pomocí menu ve skriptu `start.py`, možností 10]. Následně vyhodnotit pomocí nástroje Auditd, nebo do nich přímo nahlédnout, například pomocí příkazu `cat nazev-souboru`.

Alarm správci

Aby byla umožněna co nejrychlejší notifikace správce, či administrátora tohoto honeypotu o možném, aktuálně probíhajícím síťovém útoku, byl vytvořen skript zasílající e-mail s detaily právě probíhajícího útoku. Součástí takové zprávy je časová známka, vypsaná délka, během které byla konkrétní data zachycena, množství rámců, včetně jejich cílových a zdrojových adres, které tuto notifikaci vyvolaly. Příklad možné notifikace s detekovaným útokem je zachycen na obrázku 6.4. Aby tento skript fungoval, bylo dále využito veřejného SMTP, e-mailové adresy zřízené pro zasílání notifikací a balíčku Postfix v Ubuntu.

Obr. 6.4: Modelová zpráva při notifikaci deautentizačního útoku

To, zda budou při detekci síťového útoku zasílány notifikace, lze nastavit v `prahoveHodnoty.txt` nebo pomocí skriptu `start.py` v jeho menu. Pokud uživatel nevyplní pole určené pro zapsání e-mailové adresy, notifikace nebudou zasílány. Stejně jako omezení zasílání notifikace pak změnou e-mailové adresy lze určit i příjemce těchto zpráv.

6.4 Interakce honeypotu

Administrátor může pomocí menu dostupného ve skriptu `start.py`, vyvolatelného přes terminál `python3 start.py` a zvolit mezi dvěma úrovněmi interakce, které jsou popsány v kapitole 2.2. Na výběr pak má z vysoké a nízké interakce.

Vysoká interakce nedefinuje žádná omezení z žádného síťového rozhraní v rámci guest OS v podobě firewallu ani iptables. Iptables definuje omezení přístupu do zařízení v případě nízké interakce, jak je zobrazeno ve výpisu 6.3.

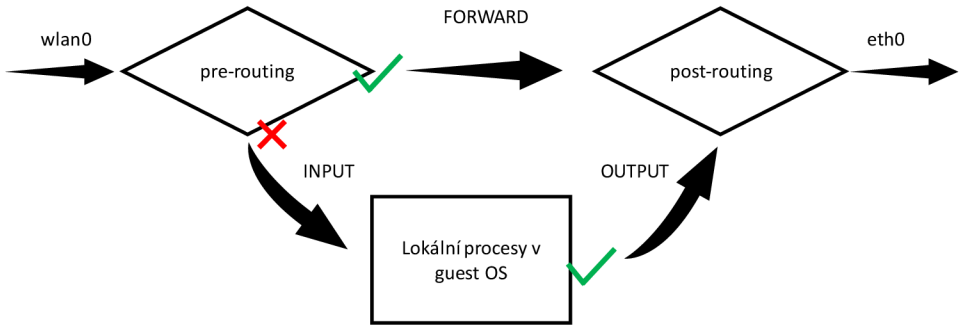
```
-A INPUT -i eth0 -j ACCEPT
-A INPUT -i wlan0 -j LOG --log-prefix "nizkaInteraceLOGGING:"
--log-level 1
-A INPUT -i wlan0 -j DROP

-A FORWARD -i wlan0 -j ACCEPT
-A OUTPUT -j ACCEPT

-A POSTROUTING -o eth0 -j MASQUERADE
```

Výpis 6.3: Konfigurace iptables na guest OS při nízké interakci

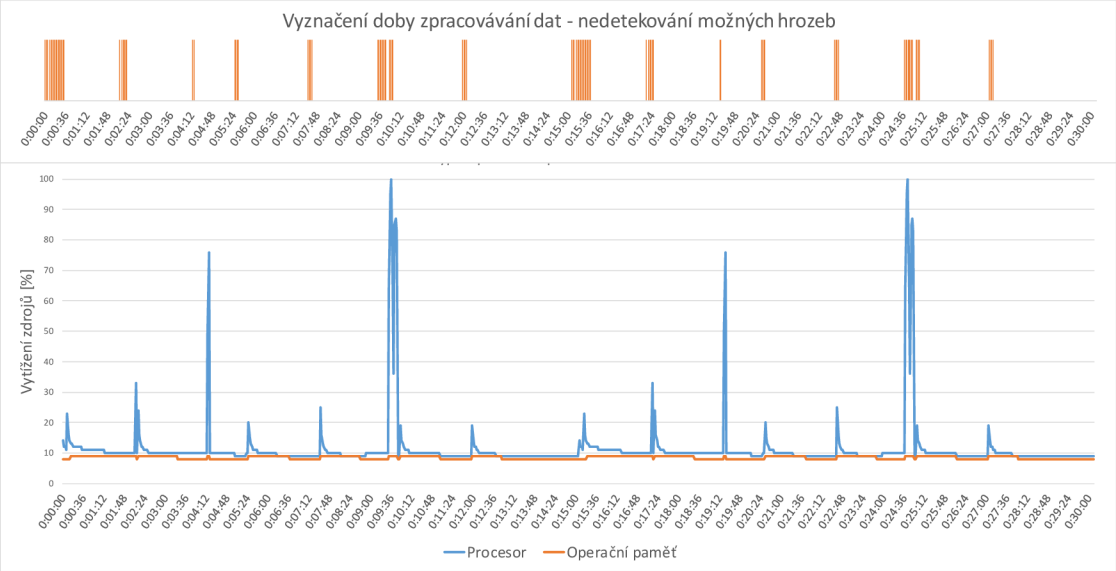
V nízké interakci je zamezeno přístupu do guest OS, respektive do AP z rádiového rozhraní wlan0. Dále je zachována funkcionalita AP, ve smyslu obsluhy uživatelů a výměny datové komunikace, procházející přes přístupový bod v řetězci forward, jak je znázorněno na obrázku 6.5. Veškeré pokusy o přístup do zařízení z rozhraní wlan0 v tomto režimu interakce jsou logovány do `kern-auth.log` a následně zamítnuty.



Obr. 6.5: Omezení logického datového řetězce INPUT při nízké interakci

6.5 Optimalizace

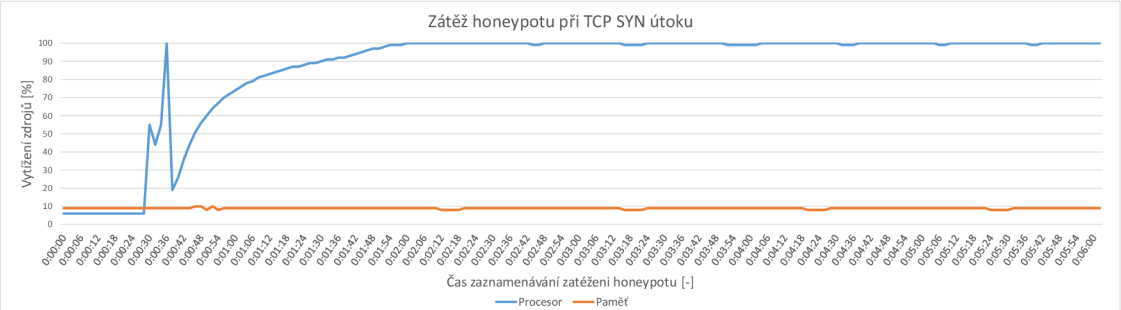
Po vytvoření stabilně fungujícího AP, spolu se schopností detekování simulovaných síťových útoků, proběhlo zátěžové testování. S použitím různých délek monitorování, a tím různě velkými objemy zachycených dat, které byly následně analyzovány a vyhodnocovány. Pomocí tohoto testování byly odhaleny nedostatky ve zpracování a analyzování zachycených dat při monitorování pomocí PCIe WiFi rozhraní.



Obr. 6.6: Vyznačení doby zpracovávání dat a nedetekování možných hrozeb

Na obrázku 6.6 je znázorněno neúměrně dlouhé vytížení procesoru a operační paměti honeypotu po dobu analyzování dat zachycených během monitorování. Během této doby je monitorování i veškeré další činnosti honeypotu běžně vedoucí k detekci možné hrozby nečinné. Během této doby může probíhat některý z útoků, který tak nebude zachycen.

Proto byla zkrácena doba, po kterou jsou data zachytávána na 120 vteřin. Tato časová úsečka může být administrátorem jednoduše změněna, ale empiricky se osvědčila během všech typů útoků jako vhodná pro dostatečně rychle zpracovatelnou dobu a potenciálně zaznamenané množství zachycených dat. Jako příklad déle probíhajícího zachytávání dat při záplavovém TCP SYN útoku, náročném na analýzu z pohledu výpočetních zdrojů - trvajícím 4 minut (240 vteřin). Při takto déletrvajícím monitorování a zachytávání dat dochází následně k jejich až několika desítek minut trvajícímu analyzování a vyhodnocení, jak je zobrazeno na obrázku 6.7.



Obr. 6.7: Příklad přetížení výpočetních zdrojů honeypotu

Aby byla co nejvíce omezena doba, kdy honeypot není schopen detekovat žádnou hrozbu, jsou data analyzovány a vyhodnocovány paralelně. Během jejich zpracování jsou již zachytávána data nová, která jsou po specifikovaném čase předána k analýze.

Paralelní zpracování zachycených dat pomocí monitorování rádiového prostředí bylo zajištěno vytvořením služby `analyzaMon.service`. Tato služba je vyvolána vždy po dokončení monitorovacího cyklu. Je pro ni vytvořen nezávislý proces, na procesu hlavní smyčky skriptu a nijak se díky tomu neovlivňují.

7 Nasazení při reálném provozu

Aby bylo možné zařízení otestovat před dlouhodobým nasazením, bylo provedeno testování v reálném provozu. Záměrem tohoto testování bylo provést kontrolovanou interakci s náhodnými uživateli a potencionálními útočníky. S tímto nasazením byla spojena myšlenka a snaha provést další, dodatečnou optimalizaci případných nedostatků, které by byly odhaleny.

Běh v reálném prostředí byl proveden s ohledem na omezení a restrikce spojené s koronavirovou karanténní situací jara, 2020. Proto bylo zařízení umístěno v domácím prostředí - v husté zástavbě, s vysokým pohybem procházejících, potencionálních uživatelů i útočníku. Toto provedení je pouze kompenzací vzhledem k již zmíněné globální koronavirové krizi. Testování zařízení v reálném prostředí zahrnovalo:

- Zkoumání reálné interakce s náhodnými uživateli,
- ověření možného výpočetního zatížení systému a jeho stability,
- vyhodnocení datové náročnosti na úložiště.

7.1 Zatížení výpočetních zdrojů systému

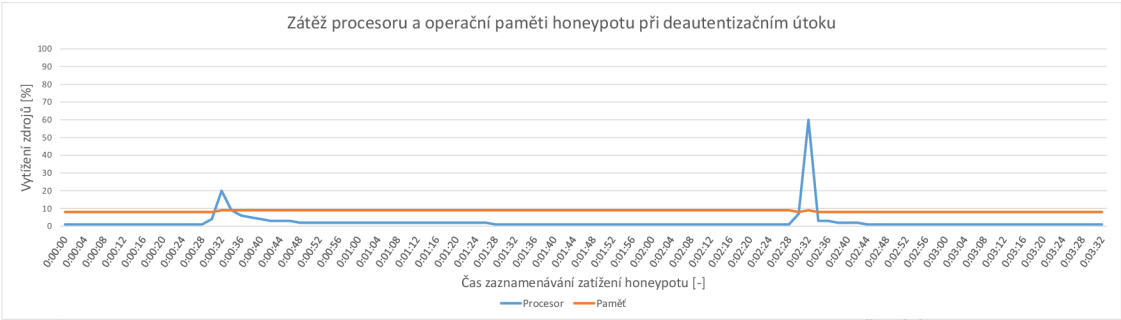
Během nasazení při reálném provozu se nepodařilo vzhledem k omezenému pohybu potenciálních uživatelů i útočníků zaznamenat validní data, prokazující detekci útoků náhodným útočníkem ,a tedy vyhodnotit zpracování těchto dat z pohledu zatížení výpočetní zdrojů a operační paměti zařízení. Proto byly v tomto prostředí provedeny vlastní penetrační testy.

Během monitorování jednotlivých síťových útoků bylo zaznamenáno zatížení výpočetních zdrojů a operační paměti honeypotu. Pro přehlednost a čitelnost níže zobrazených grafů jsou tato data znázorněna v kratším časovém úseku odpovídajícím 210 vteřin. I přes kratší dobu zobrazeného záznamu jsou však zachyceny všechny podstatné body jednoho průběhu monitorování i jeho následného vyhodnocení počínající ve všech grafech v čase 0:30 a následně 2:30. Poté je zachycen nárůst vytížení výpočetních zdrojů spojený s analýzou dat. Konkrétní detekované a monitorované útoky jsou:

- Deautentizační útok,
- Autentizační záplava,
- Beacon (SSID) záplava,
- TCP SYN záplava,
- ARP spoofing.

Deautentizační útok

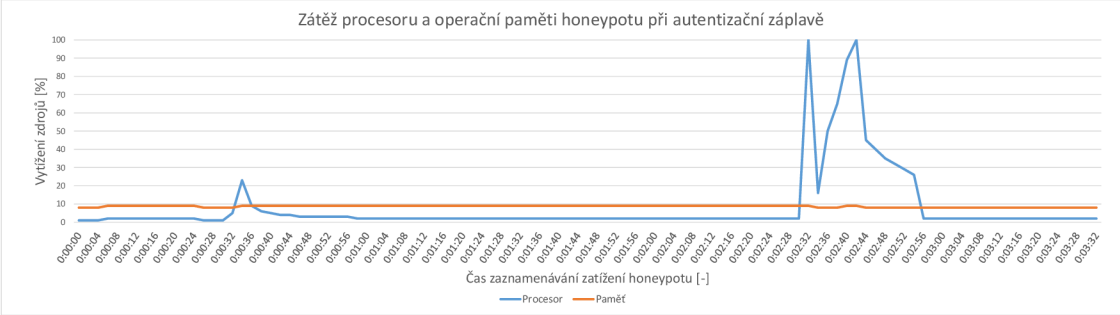
Během probíhajícího deautentizačního útoku vedeného proti konkrétnímu, připojenému uživateli AP dojde ke znemožnění dalšího využívání služeb daného AP. Pouze pro dotčeného uživatele. Ostatní uživatelé jsou při tomto útoku schopni nadále využívat služeb AP. Na základě intenzity a délky probíhajícího deautentizačního útoku je zachyceno větší množství dat na rádiovém rozhraní. Nejde ovšem o mimořádný nárůst a honeypot je schopen tyto data standardně zpracovat. Detail optimalizovaného cyklu zachycení a zpracování těchto dat je na obrázku 7.1. Při testování deautentizačním útoku nástrojem `aireplay-ng` odpovídal nárůst zachycených dat o 2,1 MB na 60 vteřin záznamu.



Obr. 7.1: Zatížení procesoru a operační paměti honeypotu při deautentizačním útoku

Autentizační útok

U autentizačního, záplavového útoku, který je vedený přímo proti AP bylo následkem úplné odepření všech uživatelských služeb, pro všechny uživatele po dobu jeho trvání. Obdobně, jako ve výše popsané situaci, kdy byl testován deautentizační útok, je možné provést toto testování nástrojem `aireplay-ng`. Zaznamenaný datový nárůst odpovídající tomuto útoku při jeho zachycení a testování činil 4,3 MB na 60 vteřin záznamu.



Obr. 7.2: Zatížení procesoru a operační paměti honeypotu při autentizační DoS záplavě

Beacon (SSID) záplavový útok

Při zachycení Beacon (SSID) záplavy nedošlo k odepření služeb jednotlivých uživatelů, ani omezení fungování samotného AP. Během testovaného záplavového Beacon útoku, který byl realizován několika způsoby došlo k zachycení a zaznamenání abnormálního nárůstu nově naskenovaných SSID. Podle konfigurace penetračního nástroje `mdk3`, pomocí kterého byl útok realizován. To bylo provedeno v následujících konfiguracích:

- Náhodně vygenerovaný rádiový kanál a názvy SSID,
- specifický rádiový kanál a náhodně vygenerované názvy SSID,
- specifický rádiový kanál a specifické názvy SSID.

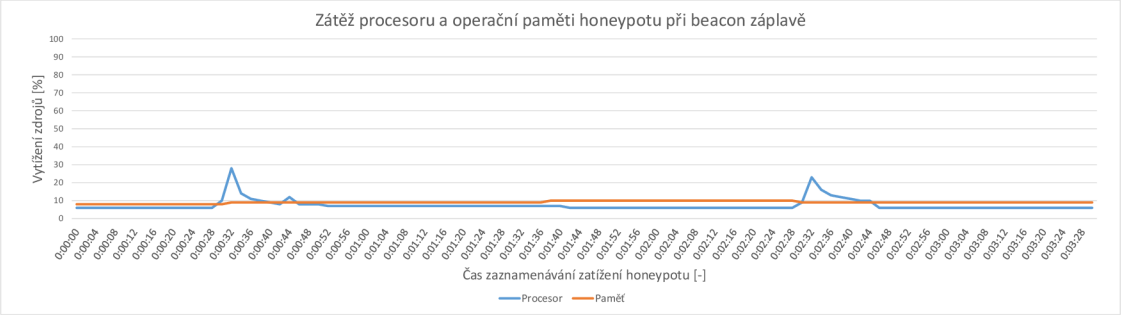
```
0.127944660 64:ee:b7:c6:75:d5 → ff:ff:ff:ff:ff:ff Beacon frame, SN=46, FN=0, Flags=........C, BI=100, SSID=wifi
0.242968621 f6:05:94:01:be:b4 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi1test
0.259311444 bc:44:78:fa:49:69 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi2test
0.275518380 e6:23:d0:1a:da:69 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi3test
0.291714375 6a:7e:4c:7e:51:25 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi4test
0.307712215 b3:48:84:53:3a:94 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi5test
0.324025282 fb:31:99:90:32:57 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi6test
0.230355355 64:ee:b7:c6:75:d5 → ff:ff:ff:ff:ff:ff Beacon frame, SN=47, FN=0, Flags=........C, BI=100, SSID=wifi
0.129628696 4a:c4:30:f6:20:23 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi7test
0.145943015 85:6c:fb:b2:07:04 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi8test
0.162005317 f4:ec:0b:b9:20:ba → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifi9test
0.178345655 86:c3:3e:05:f1:ec → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifiAtest
0.194394211 d9:67:33:b7:99:50 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifiBtest
0.210593082 a3:e3:14:d3:d9:34 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifiCtest
0.226928862 f7:5e:a0:f2:10:a8 → ff:ff:ff:ff:ff:ff Beacon frame, SN=0, FN=0, Flags=........C, BI=100, SSID=wifiDtest
0.332756693 64:ee:b7:c6:75:d5 → ff:ff:ff:ff:ff:ff Beacon frame, SN=48, FN=0, Flags=........C, BI=100, SSID=wifi
```

Obr. 7.3: Zachycený beacon (SSID) záplavový útok

Detekce tohoto typu útoku, jak je popsána v kapitole 6.3, je připravena na všechny zmíněné varianty konfigurace mdk3. Na obrázku 7.3 je zobrazený snímek vyfiltrovaných datových záznamů vztahujících se k útoku v konfiguraci - specifický

rádiový kanál a specifické názvy (slovník) SSID. Zaznamenané pomocí tSharku, zálohovaná do textového souboru pro další možnou manipulaci.

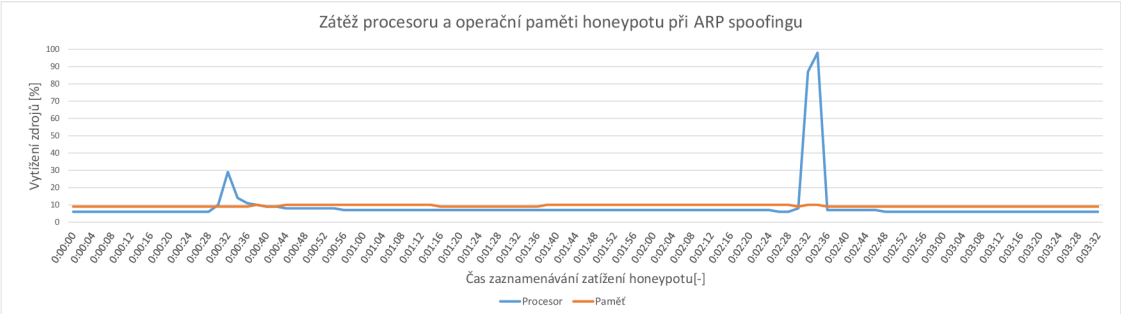
Dodatečná datová zátěž, způsobená tímto záplavovým útokem odpovídala 1,1 MB na 60 vteřin záznamu. Obrázek 7.4 znázorňuje vytížení zdrojů honeypotu při tomto typu útoku.



Obr. 7.4: Zatížení procesoru a operační paměti honeypotu při beacon (SSID) záplavě

ARP spoofing

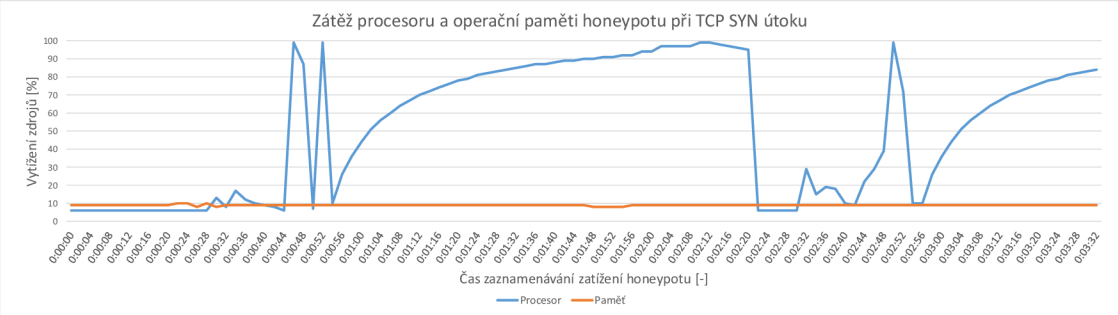
Simulovaný ARP spoofing neměl dopad na služby poskytované AP, ani jejich odepření některému z uživatelů. Tento typ útoku byl testován pomocí nástroje `mdk3`. Během zachytávání dat tohoto bezpečnostního incidentu dochází k minimálnímu navýšení datového přenosu, odpovídající 0,1 MB na 60 vteřin záznamu. Na obrázku je znázorněno vytížení zdrojů honeypotu při analyzování zachycených dat během tohoto útoku.



Obr. 7.5: Zatížení procesoru a operační paměti honeypotu při ARP spoofingu

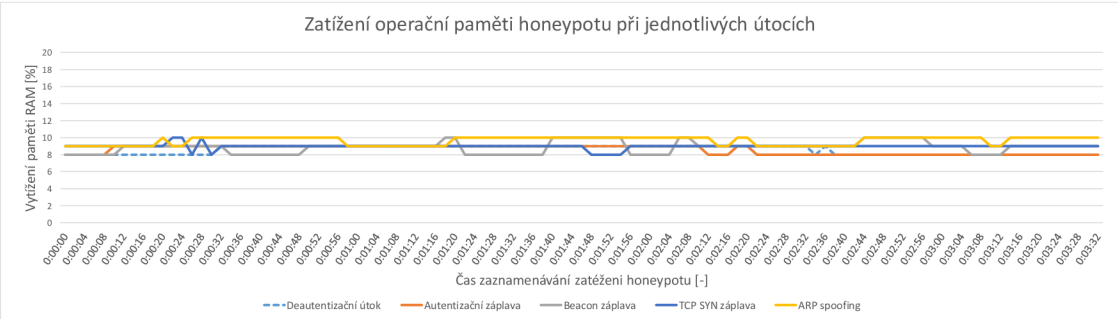
TCP SYN záplavový útok

Během probíhajícího TCP SYN záplavového útoku dochází k plnému vytížení zdrojů AP. Tím pádem k odepření jím poskytovaných služeb všem uživatelům. Záplavový útok TCP SYN byl realizován za pomocí software `mdk3`. Na obrázku 7.6 je zobrazeno zatížení výpočetních zdrojů honeypotu, které je nejvyšší ze všech testovaných útoků. V tomto případě bylo nejvyšší i zatížení datové. Dodatečná datová zátěž totiž dosahovala 54,7 MB na 60 vteřin záznamu.



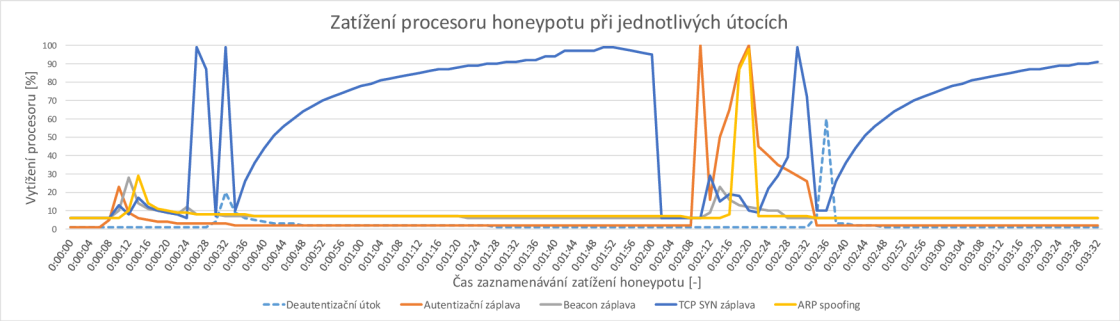
Obr. 7.6: Zatížení procesoru a operační paměti honeypotu při TCP SYN DoS záplavovém útoku

Pro porovnání náročnosti jak na vytížení výpočetních zdrojů i operační paměti zařízení jsou níže zobrazeny grafy se záznamem zatížení při všech detekovaných útocích. V grafu 7.7 je zatížení operační paměti, která bez větších výchylek vystačuje se spotřebou mezi 8 až 10 %. To ukazuje na minimální zatížení systému po stránce potřeby výpočetní paměti.



Obr. 7.7: Zatížení operační paměti honeypotu při jednotlivých útocích

V grafu 7.8 je zobrazeno zatížení výpočetních zdrojů, které je daleko razantnější než v případě operační paměti. Nabývá maximálního zatížení při zpracování monitorovaných dat, jejichž proces byl optimalizován, aby nedocházelo k přetěžování systému a odepření služeb AP a detekce síťových hrozeb.



Obr. 7.8: Zatížení operační paměti honeypotu při jednotlivých útocích

Tabulka 7.1 shrnuje množství dat, které generují jednotlivé útoky při konfiguraci penetračních nástrojů, uvedených v kapitole 3.1. To spolu s informací, zda je zachována dostupnost a služby AP pro uživatele. V případě deautentizačního útoku je AP s jeho službami dostupné všem uživatelům až na konkrétního uživatele, na nějž je veden útok.

Typ útoku	množství dat /60 vteřin [MB]	dostupnost služeb AP
Deautentizační	2,1	dostupné
Autentizační záplava	4,3	nedostupné
Beacon SSID záplava	1,1	dostupné
TCP SYN záplava	54,7	nedostupné
ARP spoofing	0,01	dostupné

Tab. 7.1: Nárůst datového zatížení při zaznamenávání jednotlivých útoků

Nejvýznamnější množství dat je generováno při záplavovém útoku TCP SYN. Právě tento typ útoku vedl k podmětu optimalizace paralelního zpracování, analýzy a případné zálohy zachycených dat na síťovém rozhraní wlan0, popsaného v kapitole 6.5.

7.2 Zatížení kapacit datového úložiště

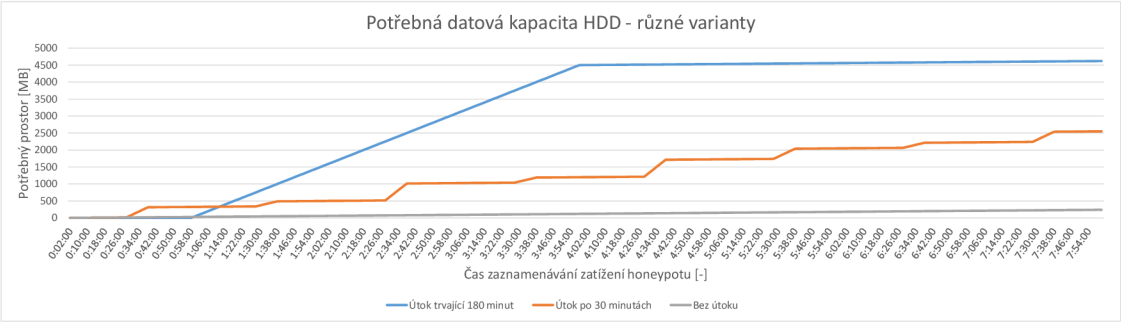
V závislosti na různě náročnou spotřebu datového úložiště honeypotu při běhu zařízení v reálném prostředí je nutné při nasazení honeypotu brát zřetel na kapacitu datového úložiště zařízení. Běžný provoz pak lze rozdělit do několika charakteristických kategorií, právě podle zaznamenané náročnosti na spotřebu datového úložiště:

- Uživatelský datový provoz,
- uživatelský datový provoz s ojedinělými útoky,
- uživatelský datový provoz s intenzivními útoky.

Jak znázorňuje tabulka 7.1 v předchozí kapitole 7.1, ne každý útok je definovaný schopností zachovaní služeb AP a zároveň také velkého množství dodatečně zpracovaných a zálohovaných dat.

Během standardního užívání AP dochází k minimálnímu vytížení úložných zdrojů honeypotu. Průběžně jsou ukládána pouze data vykreslující do grafů orientační informace o počtu naskenovaných, monitorovaných SSID a zachyceném množství beacon rámců. Ty jsou ukládány během každého cyklu monitorování, jehož délku si může administrátor navolit pomocí menu ve skriptu `start.py`, nebo přímo v konfiguračním dokumentu `prahoveHodnoty.txt`.

Při detekování ojedinělých, nebo krátkodobých útoků, ke kterým v této situaci dochází, s ohledem k typu útoků k mírnému a středně velkému příbytku spotřeby úložné kapacity honeypotu. Na níže zobrazeném obrázku 7.9 je graf znázorňující nárůst datové spotřeby úložiště honeypotu při opakovaně zachycených útocích se střídající se intenzitou zatížení 10 MB, 50 MB a 100 MB na 5 minut zálohovaného záznamu. V tomto případě je nezbytné počítat se spotřebou datového úložiště na 7,5 GB 24 hodin.



Obr. 7.9: Spotřeba úložné paměti honeypotu při záznamu různých síťových útoků

U situace se zaznamenanými intenzivními útoky dochází k mimořádné spotřebě kapacity datového úložiště honeypotu. Stejně jak v předchozích modelových situacích s ohledem na typ konkrétního útoku. V této situaci je na obrázku 7.9 zaznamenána spotřeba dat odpovídající 48,1 MB na 5 minut zálohovaného záznamu. Při nasazení honeypotu a braní ohledu na tuto situaci je nezbytné počítat se spotřebou datového úložiště 13,8 GB na 24 hodin.

Ze zpracovaných dat, získaných při nasazení v reálném prostředí, lze na základě různých scénářů očekávat odlišně dlouhou dobu, po kterou lze zařízení nechat operovat. V případě "nezájmu"o WiFi honeypot ze strany útočníků a nezachytávání žádných útoků může zařízení vystačit s 50 GB úložné kapacity pro logy po dobu delší než 50 dní. V případě druhé situace - při občasných útocích pak úložiště o kapacitě 50 GB odpovídá možné době běhu honeypotu přibližně 6 dní. Pří déletrvajících, případně na množství zálohovaných dat náročnějších útocích bude očekávaná doba výdrže 50 GB velkého datového úložiště postačovat na přibližně 3 dny.

8 Závěr

V úvodu této diplomové práce byly popsány teoretické oblasti, spojené se zadáním práce a nároky na její řešení. Konkrétně komunikační protokoly IEEE 802.11 a jejich zabezpečení. Následně byly popsány honeypoty obecně i specifické vlastnosti pro WiFi honeypoty.

Dále byla provedena analýza zranitelností v rádiovém prostředí, využívající protokolů IEEE 802.11, kde byly stanoveny nejběžnější vektory síťových útoků v rádiovém prostředí. Na základě popisu těchto útoků a možných programů pro vyvolání penetračních testů, byly zjištěné informace a metody využity k potvrzení těchto zranitelností, stanovení detekovatelných útoků a souvisejícího testování finálního zařízení. Práce pokračuje výběrem vhodného zařízení pro vytvoření vlastního WiFi honeypotu. Výběr proběhl z několika kategorií zařízení, konkrétně mikrokontrolérů, jednoúčelových WiFi směrovačů a mini ITX/mini ATX základních desek. Byla vybrána základní deska formátu mini ITX. Vzhledem k nejlepším možnostem tohoto hardwaru pro konfiguraci podporující vysoké zabezpečení i virtualizaci umožňující izolaci činnosti útočníka a výpočetně náročnějšího zpracování dat přímo na tomto zařízení, provádějícího detekci hrozeb. Pro podporu potřebných funkcí proběhl výběr programů a rozšíření doplňujících potřebné vlastnosti na vytvářeném WiFi honeypotu. S ohledem na zvolené zařízení proběhl výběr možných variant host operačních systémů, konkrétně Ubuntu 18.04 LTS, Kali 2019.4, HoneyDrive 3, Network Security Toolkit 30 a Network Security Tool. Ty měly teoreticky splňovat funkce monitorování rádiového okolí, skriptů vyhodnocujících datový provoz a běh virtualizovaného přístupového bodu. S ohledem na potřebu běhu virtualizovaného přístupového bodu byly pro tuto činnost vybrány RouterOS, OpenWRT a Lubuntu jako teoreticky vhodné systémy.

V další části práce byly operační systémy určené pro vytváření host systému, tak i systémy pro vytváření přístupového bodu, otestovány experimentální konfigurací. Během této konfigurace proběhl výběr nejvhodnějšího systému s ohledem na podporu potřebných funkcionalit, programových balíčků a ovladačů hardware zajišťující stabilní běh testované konfigurace. Na základě zvolené architektury a požadovaném fungování systému byl vybrán jako nejvhodnější hostovací systém Ubuntu 18.04 LTS. Nejlépe splňoval podporu potřebných aplikací, stabilitu a podporu ovladačů pro veškerý hardware. U výběru systému pro funkci přístupového bodu byl vybrán Lubuntu, jediný umožňoval stabilní běh přístupového bodu na požadovaných komponentách, s podporou logování potřebných událostí i možností vytvoření firewallu pro nastavení různé interakce přístupového bodu. Po praktickém ověření a výběru operačních systémů byla vytvořena pomocí python skriptů stálá konfigurace pro monitorování rádiového okolí, zahrnující analyzování okolních WiFi sítí a datového provozu. Tyto

data jsou přímo na zařízení zpracována a podle stanovených prahových hodnot, u předem vybraných vektorů útoků, detekovány síťové hrozby. S možnou notifikací správce zařízení o daném incidentu, včetně zálohování dat, která byla zaznamenána v inkriminovaném čase. Během vytváření a testování těchto funkcí podléhalo zařízení a jeho konfigurace optimalizaci. Především v nasazení paralelního zpracování dat zachycených v rádiovém prostředí a stanovením vhodně dlouhého cyklu monitorování rádiového rozhraní pro úměrně dlouhý čas zpracování těchto dat.

Poslední část práce pojednává o nasazení zařízení do reálného datového provozu. Lokalita byla vzhledem k jarní koronavirové situaci omezena pouze na jednu lokalitu, na níž nebyl zaznamenán přílišný zájem ze strany útočníků ani náhodných uživatelů o přístupový bod, ani v případě otevřené konfigurace nevyžadující znalost přístupových údajů. Proto bylo zatížení systému a výpočetních zdrojů testováno vlastními penetračními útoky, které potvrdily správné fungování zařízení a detekci konkrétních síťových útoků, zahrnující deautentizační útok, záplavový SSID beacon útok, ARP spoofing, autentizační a TCP SYN záplavový útok pro odepření služeb. Následně jsou popsány volby délek sběru a zpracování dat, které jsou spojené s maximální dobou nasazení v cílové lokalitě s ohledem na spotřebu úložiště zařízení při zálohování dat v době detekovaných útoků.

Práce tak splnila požadavky, které definovalo zadání. Vytvořený WiFi Honeypot podporující širokou škálu protokolů IEEE 802.11 od IEEE 802.11b až IEEE 802.11ac pro přenos dat ve volném frekvenčním spektru 2,4 GHz a 5 GHz. Díky architektuře zařízení oddělující výpočetní zdroje pro útočníka ve virtualizovaném přístupovém bodu s doplněním firewallu je docíleno vysoké bezpečnosti. WiFi honeypot nabízí vysokou i nízkou interakci a je otestován pro běh v reálném provozu. Další pokračování této práce lze uvážit v rozšíření palety detekovatelných útoků a nekalé činnosti pocházející z rádiového prostředí WiFi. Případně další testování, které po úplném rozvolnění karanténních opatření nabídne nové poznatky a zajímavá data ke zpracování a analýze.

Literatura

[1] Lochan Verma a další. Wifi on steroids: 802.11 ac and 802.11 ad. *IEEE Wireless Communications*, pages 30–35, 2013.

[2] Vitosinschi Alexandr. 802.11ac wireless throughput testing and validation guide. Dostupné z <https://www.cisco.com/c/en/us/support/docs/wireless-mobility/wireless-lan-wlan/212892-802-11ac-wireless-throughput-testing-and.html>.

[3] Skupina IEEE 802.11. Ieee 802.11 wireless local area networks. Dostupné z <http://www.ieee802.org/11/>.

[4] Verney Brett a další. Laminated card 802.11n/ht and 802.11ac/vht | mcs, snr and rssi. Dostupné z <https://www.wlanpros.com/resources/laminated-card-802-11nht-802-11acvht-mcs-snr-rssi/>.

[5] Lashkari Arash Habibi a další. Wired equivalent privacy (wep). In *2009 International Conference on Future Computer and Communication*, pages 492–495. IEEE, 2009.

[6] Lashkari Arash Habibi a další. Wired equivalent privacy (wep) versus wi-fi protected access (wpa). In *2009 International Conference on Signal Processing Systems*, pages 445–449. IEEE, 2009.

[7] Gast Matthew. *802.11 wireless networks: the definitive guide*. "O'Reilly Media, Inc.", 2002.

[8] Shivaputrappa Vibhuti. Ieee 802.11 wep (wired equivalent privacy) concepts and vulnerability. *San Jose State University, CA, USA, CS265 Spring*, 2005.

[9] Šustr Matej. Analýza bezpečnosti štandardu ieee 802.11. *Bratislava: FEI STU*, 2005.

[10] Abdel-Karim Al-Tamimi and Jain Raj. *Security in wireless data networks*, pages 3–38. 2011.

[11] Geier Jim. Wpa security enhancements. *Jupitermedia Corp.*, 2003. Dostupné z <http://www.wifiplanet.com/tutorials/article.php/2148721>.

[12] Lehembre. Wi-fi security–wep, wpa and wpa2. *Hackin9 (January 2006)*, 2005.

[13] Endrle Pavel. Zabezpečení standardu 802.11 a jeho možnosti. Master's thesis, Vysoké učení technické v Brně, 2009.

[14] Vojtíšek Jindřich. Analýza šifrovacích algoritmů ve standardu 802.11. Master's thesis, Vysoké učení technické v Brně, 2014.

[15] Wi-Fi Alliance. Wpa3 security considerations overview, 2019.

[16] Hopping Clare. Wi-fi alliance rolls out wpa3 to boost wireless security. *IT Pro*, 2018. Dostupné z <https://search.proquest.com/docview/2059865943?accountid=17115> Poslední aktualizace - 2018-06-28.

[17] Sari Arif a další. Comparative analysis of wireless security protocols: Wep vs wpa. *International Journal of Communications, Network and System Sciences*, pages 483–491, 2015.

[18] Stoll Cliff. *The cuckoo's egg: tracking a spy through the maze of computer espionage*. Simon and Schuster, 2005.

[19] Cheswick Bill. An evening with berferd in which a cracker is lured, endured, and studied. In *Proc. Winter USENIX Conference, San Francisco*, pages 20–24, 1992.

[20] Cohen Fred. Simulating cyber attacks, defences, and consequences. *Computers & Security*, pages 479–518, 1999.

[21] Spitzner Lance. *Honeypots: Tracking Hackers*, volume 1. Addison-Wesley Longman Publishing Co., Inc., 2003.

[22] Deniz Akkaya and Fabien Thalgott. Honeypots in network security, 2010. Dostupné z <http://www.diva-portal.org/smash/get/diva2:327476/fulltext01>.

[23] Chuvakin Anton a další. Chapter 6 - covert logging. In Anton Chuvakin a další, editor, *Logging and Log Management*, pages 103–114. Syngress, Boston, 2013. Dostupné z <http://www.sciencedirect.com/science/article/pii/B9781597496353000063>.

[24] Anjali Sardana a další. Wireless honeypot: Framework, architectures and tools. Dostupné z <http://ijns.jalaxy.com.tw/contents/ijns-v15-n5/ijns-2013-v15-n5-p373-383.pdf>.

[25] d'Otreppe de Bouvette Thomas a další. Aircrack-ng. Dostupné z <https://www.aircrack-ng.org/documentation.html>.

[26] Měrka Tomáš. Monitorování hrozeb wi-fi sítí za pomocí honeypot. 2012.

[27] Tews Erik a další. Breaking 104 bit wep in less than 60 seconds. In *International Workshop on Information Security Applications*. Springer, 2007.

[28] Vanhoef Mathy a další. Key reinstallation attacks: Forcing nonce reuse in wpa2. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1313–1328. ACM, 2017.

[29] Canonical Ltd. Ubuntu 18.04 dokumentace. Dostupné z <https://help.ubuntu.com/18.04/ubuntu-help/index.html>.

[30] Aharoni Mati a další. Kali. Dostupné z <https://www.kali.org/docs/>.

[31] Sioannis Koniaris a další. Honey drive 3. Dostupné z <https://bruteforce.gr/honeydrive/>.

[32] Paul Blankenbaker a další. Network security tool 30. Dostupné z <https://www.networksecuritytoolkit.org/nst/index.html>.

[33] Mark Cumming a další. Security tool distribution. Dostupné z <https://s-t-d.org/tools.html>.

[34] Simon Quigley a další. Dokumentace lubuntu. Dostupné z <https://docs.lubuntu.net/>.

[35] MikroTik. Dokumentace roueteros. Dostupné z <https://wiki.mikrotik.com/wiki/Manual:System/Packages>.

[36] Schmutzler Adrian a další. Dokumentace openwrt. Dostupné z <https://openwrt.org/cs/docs/start>.

[37] Conner Jimmy a další. Cacti - the complete rrdtool-based graphing solution. Dostupné z <https://www.cacti.net/>.

[38] Kristóf Marussy a další. Firejail suid. Dostupné z <https://firejail.wordpress.com/>.

[39] Provos Niels. Dokumentace honeyd. Dostupné z <http://www.honeyd.org/general.php>.

[40] NETSEC. Specter ids. Dostupné z <http://gabiam.com/software/laura_chapelle/Software/specter/>.

[41] Gerald Combs a další. Wireshark. Dostupné z <https://www.wireshark.org/#learnWS>.

[42] VMware. Dokumentace vmware. Dostupné z <https://www.vmware.com/solutions/virtualization.htm>.

[43] Oracle. Virtualbox. Dostupné z <https://www.virtualbox.org/wiki/Documentation>.

Seznam symbolů, veličin a zkratek

WiFi	Wireless Fidelity
MAC	Media Access Control
ISO/OSI	International Standards Organization/Open Systems Interconnection
LAN	Local Area Network
WLAN	Wireless LAN
DQPSK	Differential Quadrature Phase-Shift Keying
IEEE	Institute of Electrical and Electronics Engineers
MAN	Metropolitan Area Network
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
SU-SISO	Single User - Single Input Singe Output
SU-SIMO	Single User - Single Input Multiple Output
SU-MISO	Single User - Multiple Input Singe Output
SU-MIMO	Single User - Multiple Input Multiple Output
MU-MIMO	Multiple User - Multiple Input Multiple Output
RSSI	Received Signal Strength Indication
BPSK	Binary-Phase Shift Keying
QPSK	Quadrature Phase Shift Keying
QAM	Quadrature Amplitude Modulation
MCS	Modulating and Coding Scheme
AP	Access Point
WEP	Wired Equivalent Privacy
WPA	Wi-Fi Protected Access
RC4	Rivest Cipher 4
CRC	Cyclic Redundancy Check
IV	Initialization Vector
ICV	Integrity Check Value
TKIP	emporal Key Integrity Protocol
MIC	Message Integrity Code
MAC	Message Authentication Code
WPA2	Wi-Fi Protected Access 2
CCMP	Counter Cipher Mode with Block Chaining Message Authentication Code Protocol
AES	Advanced Encryption Standard
DES	Data Encryption Standard
CBC-MAC	Cipher Block Chaining Message Authentication Code
MPDU	Medium Access Control Protocol Data Unit
Ext IV	Extended Initialization Vector

ID	Identification Data
WPA3	Wi-Fi Protected Access 3
AAD	Additional Authentication Data
MSDU	MAC Service Data Unit
SAE	Simultaneous Authentication of Equals
PAKE	Password Authenticated Key Exchange
WPA2-PSK	WPA2-Pre Shared Key
M2M	Machine to Machine communication
SSH	Secure Shell
OS	Operační Systém
SSID	Service Set Identifier
IDS	Intrusion Detection System
DoS	Denial of Service
ARP	Address Resolution Protocol
IP	Internet Protocol
TCP	Transmission Control Protocol
PCIe	Peripheral Component Interconnect Express
USB	Universal Serial Bus
GPS	Global Positioning System
MITM	Man In The Middle
GPLv3	GNU General Public Licence v3
DNS	Domain Name System
ARP	Address Resolution Protocol
KRACK	Key Reinstallation Attack
CPU	Central Processing Unit
RAM	Random Access Memory
NST	Network Security Toolkit 30
STD	Secority Tool Distribution
ARM	Advanced RISC Machine
PPC	Power PC
GUI	Graphical User Interface
CSV	Comma-separated values
GNU	GNU's Not Unix
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
GRUB	GNU GRand Unified Bootloader
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System

Seznam příloh

A Obsah přiloženého ZIP souboru

V adresáři `Skripty` jsou obsaženy veškeré zdrojové kódy WiFi honeypotu. Ve složce `Sluzba_Analyza` je umístěna struktura a kód služby pro volání skriptu analýzy zachycených dat. Profily přístupového bodu, které se periodicky přepínají pomocí nástroje Cron a skriptů jsou ve složce `Profily_AP`. Na snímku umístěném v dokumentu `Cinnost_HP.pdf` je zjednodušeně znázorněna logická struktura vazeb mezi skripty, jejich spouštění a spolu s případnými soubory a složkami na honeypotu pro ukládání a zpracování dat.

- `/` ..... Adresář ZIP souboru
  - `Skripty` ..... Zdrojové kódy honeypotu
  - `Sluzba_Analyza` ..... Služba pro zpracování analýzy
  - `Profily_AP` ..... Profily jednotlivých režimů AP na honeypotu
  - `Cinnost_HP.pdf` ..... Zjednodušené struktura činnosti HP