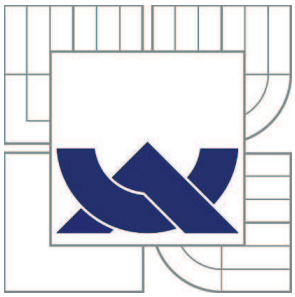
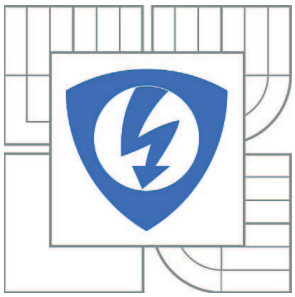


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

ÚTOKY POMOCÍ PROGRAMU CAIN & ABEL

NETWORK ATTACKS BY CAIN & ABEL

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

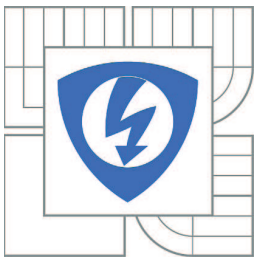
AUTOR PRÁCE — Bc. LUKÁŠ SMÉKAL

AUTHOR

VEDOUCÍ PRÁCE — Ing. JIŘÍ SOBOTKA

SUPERVISOR

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student:	Bc. Lukáš Smékal	***ID:***	78359
Ročník:	2	***Akademický rok:***	2009/2010

NÁZEV TÉMATU:

Útoky pomocí programu Cain & Abel

POKYNY PRO VYPRACOVÁNÍ:

Analyzujte prvky programu a navrhněte jejich využití pro útoky na kryptografické algoritmy a zabezpečení počítačových sítí. Porovnejte jednotlivé metody útoků a proveďte jejich praktické aplikace. Z dosažených výsledků vytvořte návod na použití programu Cain & Abel.
Dále se podrobněji zabývejte modulem RSA SecureID Token Calculator a pokuste se pomocí tohoto modulu prolomit způsob autentizace pomocí RSA tokenů vyučovaný v předmětu Kryptografie v informatice.

DOPORUČENÁ LITERATURA:

[1] SCHNEIER, Bruce. Applied cryptography. 2nd edition. [s.l.] : John Wiley & Sons, 1996. 784 s. ISBN 0-471-11709-9 .
[2] http://www.oxid.it/cain.html

Termín zadání:	29.1.2010	***Termín odevzdání:***	26.5.2010

Vedoucí práce: Ing. Jiří Sobotka

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ANOTACE

Tato diplomová práce se zabývá problematikou zabezpečení lokálních počítačových sítí LAN, kryptografickými algoritmy, jednotlivými metodami útoků na počítačové sítě a praktickou aplikací těchto útoků v lokální síti LAN. Pro aplikaci jednotlivých útoků je využito programu Cain & Abel. Z výsledků těchto útoků je vytvořen podrobný návod na tento program, který obsahuje ukázky využití jednotlivých nástrojů programu, ukázky aplikace jednotlivých útoků, uvážení následků jednotlivých útoků a shrnutí dosažených výsledků při práci s jednotlivými nástroji. Diplomová práce se podrobněji zabývá jedním z nástrojů programu, nazývaným RSA SecureID Token Calculator. Pomocí tohoto nástroje je v diplomové práci nastíněna problematika autentizace pomocí hardwarových tokenů a způsob autentizace pomocí nástroje RSA SecureID Token Calculator bez fyzického vlastnění hardwarového tokenu. Pomocí programu Cain & Abel je v diplomové práci ukázána a vysvětlena nevhodnost ukládání hesel k jednotlivým aplikacím do paměti operačního systému a jsou zde zobrazeny i metody získání těchto hesel z paměti operačního systému. Dále je práce zaměřena na problematiku odchytávání přihlašovacích údajů a různých typů hesel v lokálních počítačových sítích a na aplikaci jednotlivých útoků na kryptografické algoritmy za účelem prolomení těchto přihlašovacích údajů a hesel.

KLÍČOVÁ SLOVA: Cain&Abel, LAN, sniffer, cracker, token, heslo, kryptografie, útok

ABSTRACT

This Master's thesis is dealt in the local area network security, cryptographic algorithms, particular attacks on computer networks a practical application these attacks in local area networks. To application particular attacks is used the Cain & Abel program. The detailed manual for this program is created from the results of these attacks. This manual contains the exhibits of usage particular program tools and the attack application exhibits. This manual considers consequences of particular attacks and summarises achieved results during work with tools too. Master thesis closely deals with one of the program tools called RSA SecureID Token Calculator. Authentication via hardware tokens is contained in this Master thesis. Thesis contains the way of authentication using RSA SecureID Token Calculator without physical owning of the hardware token. Cain & Abel program shows and interprets why cashed passwords in operation system are dangerous and it shows methods how attacker can reveal this passwords from the operation system memory. This Master thesis is focused on sniffing credentials and passwords in local area networks and it is focused on cryptographic algorithms cracking for username and passwords revealing.

KEYWORDS: Cain&Abel, LAN, sniffer, cracker, token, password, cryptography, attack

SMÉKAL, L. *Útoky pomocí programu Cain & Abel.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 80 s. Vedoucí diplomové práce Ing. Jiří Sobotka.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma „Útoky pomocí programu Cain & Abel“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

podpis autora

Poděkování

Na tomto místě bych rád poděkoval mému vedoucímu diplomové práce Ing. Jiřímu Sobotkovi za odbornou a metodickou pomoc při tvorbě této diplomové práce a za ochotu a velmi cenné rady při řešení dané problematiky.

V Brně dne .....

.....
podpis autora

OBSAH

SEZNAM OBRÁZKŮ

Úvod

Tato diplomová práce je zaměřena na problematiku z oblasti bezpečnosti počítačových sítí, kryptografických algoritmů a počítačových útoků na přepínané síti. Obsah dokumentu se postupně zaměřuje na důležitost zabezpečení počítačové sítě, jaká existují rizika, jak asi útočníci postupují a jaké chyby uživatelé dělají, čímž usnadňují útočníkům jejich práci. Dále jsou zde popsány nejznámější šifrovací algoritmy, zabezpečené protokoly a je zde kladen důraz na různé techniky útoku v lokální síti. Hlavním bodem této diplomové práce je seznámení s programem Cain & Abel, analyzování možností všech nástrojů tohoto programu a jejich využití při útocích na lokální počítačové sítě. Ze zjištěných poznatků je zde vytvořen návod pro práci s tímto programem.

Při psaní diplomové práce jsem vycházel z předpokladu znalosti základních funkcí počítačových sítí, znalosti protokolu ARP, znalosti funkce switchů, dobrých znalostí operačního systému Windows nejen z pohledu uživatele ale i administrátora a základní orientace v oboru zabezpečení sítí.

1 Bezpečnost počítačových sítí

Bezpečnost se přesunula do popředí síťového managementu a implementace. Celková bezpečnost sítě spočívá v nalezení rovnováhy mezi dvěma důležitými požadavky. Prvním požadavkem je potřeba otevření sítě k podporování rozvíjejících se podnikatelských příležitostí a druhým je potřeba chránit soukromí, jak osobní tak strategické podnikové informace. Aplikování efektivní bezpečnostní politiky je nejdůležitější krok, který daná organizace musí udělat k ochraně své komunikační sítě. Poskytuje informace o provedených činnostech a prostředcích použitých k zabezpečení sítě organizace. [1]

1.1 Důležitost zabezpečení sítě

Počítačové sítě se rozrostly jak do své velikosti tak do důležitosti za velmi krátkou dobu. Pokud je bezpečnost sítě oslabena, může to mít vážné důsledky, jako je ztráta soukromí, krádež informací, dokonce i právní odpovědnost. Udržení bezpečnosti sítě je stále náročnější, protože druhy potenciálních hrozeb se neustále vyvíjejí a jsou stále sofistikovanější.

E-business, internetové aplikace a komunikace neustále narůstají, proto najít rovnováhu mezi potřebou být izolován a zároveň otevřený je velice obtížné. Navíc nárůst mobilní komerce a bezdrátových sítí vyžaduje, aby se bezpečnostní řešení podařilo integrovat do stávající sítě a bylo transparentnější a flexibilnější.

V průběhu let se nástroje a metody síťových útoků podstatně rozvíjely. Dříve útočník musel mít sofistikované počítačové a programovací znalosti a musel mít výborné znalosti síťových technologií k využití prvotních nástrojů a mohl uskutečnit základní útoky. Doba pokročila, metody a nástroje útočníků se zlepšily a již nevyžadují stejnou úroveň znalostí jako dříve. Tato skutečnost snížila požadavky na znalosti útočníka, aby byl schopen základní útoky provést. Lidé, kteří nebyli schopni se účastnit v počítačové kriminality najednou tuto schopnost mají.

Jak se typy hrozeb, útoků a zneužití vyvíjely, vznikly nové termíny (viz lit. [6]):

White hat – Osoba hledající slabiny v systému nebo síti a potom tyto slabiny oznámí vlastníkovi daného systému nebo sítě. Vlastník poté může nechat tyto slabiny odstranit.

Hacker – Dříve se takto označoval člověk s vynikajícími programovacími schopnostmi, dnes se tak označují také osoby, které se pokoušejí získat neautorizovaný přístup do síťových zdrojů se záludným záměrem.

Black hat – Jiný termín pro osoby, které využívají svých znalostí počítačových systémů k proniknutí do systémů nebo sítí, ve kterých nemají oprávnění využívat síťových zdrojů a služeb. Většinou útočníci pronikají do systémů za účelem osobního nebo finančního zisku.

Cracker – Přesnější termín popisující někoho, kdo se pokouší získat neautorizovaný přístup do síťových zdrojů se záludným záměrem.

Phreaker – Osoba manipulující s telefonní sítí zapříčiňující vykonání funkcí, které nejsou povoleny. Nejčastěji proniknutí do telefonní sítě prostřednictvím telefonního automatu, za účelem vykonání telefonních hovorů zdarma.

Spammer – Osoba rozesílají obrovská kvanta nevyžádaných e-mailových zpráv. Spammer většinou využívá virů, pomocí kterých převezme kontrolu nad domácími počítači a použije je k odeslání určitého počtu zpráv.

Phisher – Osoba používající e-mail nebo jiné prostředky k oklamání důvěřivých uživatelů, aby poskytli své citlivé informace (čísla kreditních karet, hesla, piny,…). Phisher se vydává za důvěryhodnou společnost, která by měla mít legitimní potřebu pro tyto citlivé informace.

Cílem útočníka je kompromitovat síťové zařízení nebo aplikaci běžící uvnitř sítě. Mnoho útočníků používá následující postup k získání informací (viz lit. [1],[6]):

Průzkum

Jedná se o přípravu na útok proti dané síti. V tomto kroku se hacker snaží získat všechny dostupné informace o síťovém prostředí a jeho zabezpečení. Jedná se o informace typu názvy domén, přiřazené bloky veřejných adres, dále které IP adresy provozují služby vhodné pro cílení útoku. Hacker ocení informace o hardware a software, kterým daná síť disponuje, jejich přesné typy a verze. Z těchto informací si hacker vyhodnotí zranitelná místa daného hardware i software. Dále hackera zajímá, jaký je v síti firewal, směrovací protokoly, typ vzdáleného přístupu a jeho kontrola. Získat všechny tyto informace je možné například z webových stránek společnosti, propagačních materiálů, ale většinou nejvíce a nejdůležitější informací hacker získá bez námahy pomocí „sociálního inženýrství“. [1]

Sledování a soupis informací

Na základě již získaných informací si útočník vytvoří jednoduchý náčrt sítě a sestaví si obrázek bezpečnostního profilu společnosti. Další kroky, které útočník podnikne, už mohou být zachyceny v systémových protokolech, proto si útočník musí uvědomit, co si ještě může

dovolit a co ne, aby nebyl odhalen. Na základě svého náčrtu sítě, útočník může rozšířit své informace o síti pomocí monitorování síťového provozu pomocí paketových snifferů, což jsou programy analyzující síťový provoz. Pomocí snifferů může útočník zjistit ještě přesnější informace o síti a jejich službách. Při soupisu prostředí musí útočník z pasivního útoku přejít do aktivního. To znamená, že již musí vytvářet aktivní spojení s konkrétními systémy a již vytvářet přímé požadavky na spojení. [1]

V síti se zjišťují čtyři hlavní kategorie informací (postup se odvíjí od operačního systému):

- Síťové prostředky a sdílené složky
- Uživatelé a skupiny
- Aplikace
- Úvodní zprávy zařízení

Získání přístupu

Útočník zahájí vyhledávání platných uživatelských účtů a sdílených prostředků, které jsou nedostatečně chráněné, pomocí kterých získá přístup do systému. Útočník nakonec musí získat přístup do systému prostřednictvím některé z jeho komponent. Typicky jsou to tyto útoky:

- Útoky na operační systém
- Útoky na aplikace
- Útoky přes nesprávnou konfiguraci
- Útoky pomocí skriptů

V mnoha případech hacker získá přístup opravdu snadno. Například když si zaměstnanci zvolí heslo, které se dá lehce prolomit nebo může být zaměstnanec napálen talentovaným útočníkem a vydá mu nevědomky citlivé informace týkající se přístupu. [1]

Navýšení privilegií

V této chvíli útočník má přístup do systému, ale pomocí obyčejného uživatele, který nemá potřebná oprávnění, pro další práci útočníka. Z tohoto důvodu se útočník snaží svá oprávnění navýšit. Například pomocí těchto metod:

- Spuštění určitého programového kódu
- Prolomení hesla pomocí volně dostupného nástroje
- Zachycení nešifrovaných hesel
- Zjištění vztahů mezi napadeným systémem a ostatními systémy v síti
- Hledání souborů a sdílených složek s chybně nastavenými oprávněními

Stručně řečeno, útočník zkusí, co mu systém dovolí. Pokud žádná metoda nevede k úspěchu nebo se útočník chystá provést útok s odepřením služeb (Denial of Service), pokusí se vyřadit celý systém pomocí speciálního programového kódu pro zneužití (exploit code). Naopak pokud byl útočník úspěšný jeho snaha má jasný cíl, to je získat oprávnění administrátora. Jakmile má toto oprávnění, může si dělat se systémem prakticky, co se mu zlíbí. Shromáždí si dodatečná hesla a tajné informace a začne konat svou „práci“. [1]

Zahlazení stop a instalace zpětných vrátek

Jakmile útočník získá nadvládu nad systémem, musí svou přítomnost zakrýt před administrátorem. Zpětná vrátka a zahlazení stop je jeden z nejzákladnějších úkolů, ovšem také o jeden z nejsložitějších. V systémech Windows se jedná o vymazání a vyčištění protokolu událostí a položky systémového registru. U unixových operačních systémů se jedná o vymazání souboru historie a pomocí nástroje pro „čištění“ protokolu o vymazání položky z adresářů UTMP, WTMP a LastLog. Pokud si útočník chce ponechat možnost pozdějšího návratu, vytvoří si v něm přístupovou cestu tzv. zpětná vrátka (backdoor). Vytvoření těchto zadních vrátek se liší podle operačního systému, ale vesměs vždy se jedná o vytvoření zvláštního účtu, úpravu spouštěcích souborů, zapnutí určitých služeb nebo aplikací pro vzdálené řízení atd. Zpětná vrátka poté poskytují útočníkovi vstup do systému, aniž by byl detekován. [1]

2 Bezpečnostní protokoly a algoritmy

2.1 Symetrické šifrování

Symetrickou šifrou nazýváme takový šifrovací algoritmus, který používá k šifrovaní i dešifrování dat pouze jeden klíč. To znamená, že obě strany účastnící se komunikace sdílejí stejný soukromý klíč (viz Obr. 1). Tím, že klíč musí být k dispozici oběma stranám, vzniká podstatný problém při transferu tohoto klíče od jednoho koncového uživatele k druhému. Naopak mezi výhody symetrického šifrování se řadí nízká výpočetní náročnost, protože algoritmy symetrického šifrování provádí většinou jednodušší matematické operace typu bitový součet, negace, nonekvivalence atd. Díky tomu je symetrické šifrování rychlé a jednoduše hardwarově implementovatelné [4, 10].

Symetrické šifry dělíme do dvou skupin:

- Proudové šifry
- Blokové šifry

Proudové šifry šifrují data postupně bit po bitu. Proudové šifry jsou rychlejší než ty blokové a využívají se v prostředí, kde není přítomný buffer (real-time komunikace). Mezi zástupce proudových šifer patří algoritmy RC4, FISH, Helix atd. Proudové šifry můžeme rozdělit ještě na synchronní a asynchronní (šifry s vlastní synchronizací). Synchronní proudové šifry nepoužívají k šifrování text, ale data se šifrují pomocí klíče a stavu, ve kterém se funkce nachází. Při dešifrování se musí postupně procházet ekvivalentními stavy. Pokud se při přenosu ztratí nebo přibude jediný bit, dojde ke ztrátě synchronizace. Naopak asynchronní proudové šifry využívají k šifrování i dešifrování kromě klíče i „n“ předchozích bitů šifrovaného textu. Díky tomu se zvyšuje bezpečnost. Při ztrátě jednoho bitu se následujících „n“ bitů zprávy dekóduje špatně, ale další data budou už dekódována správně. [3, 4]

Blokové šifry se v praxi využívají častěji než proudové. Šifrují data po pevně daných blocích bitů, většinou se jedná o mocninu dvou (64, 128, 256,…). Bloky bitů jsou zašifrovány a výstupem je část šifrovaného textu. Pro zvýšení bezpečnosti některé algoritmy šifrovaní daného bloku několikrát opakují. Mezi zástupce blokových šifer patří algoritmy DES, AES, IDEA, Blowfish atd. [4, 17]

Obr. 1: Symetrické šifrování

2.2 Asymetrické šifrování

Asymetrické šifrování (viz Obr. 2) na rozdíl od symetrického šifrování používá dva druhy klíčů. Veřejný (public) klíč a soukromý (private) klíč. Pomocí veřejného klíče se data zašifrují a dešifrují se pomocí soukromého klíče. Šifrování probíhá tak, že odesílatel danou zprávu zašifruje veřejným klíčem příjemce a pošle ji příjemci. Zpráva lze dešifrovat pouze soukromým klíčem příjemce. Pokud se zprávy zmocní někdo jiný a nezná soukromý klíč příjemce, nemůže tuto zprávu dešifrovat.

U asymetrického šifrování nastává problém při distribuci veřejného klíče. Odesílatel při šifrování dat používá veřejný klíč příjemce. Než odesílatel tyto data může zašifrovat, musí někde tento klíč získat. Většinou si tento klíč stáhne z internetu nebo si ho musí nechat poslat. Problém nastává právě při přenosu tohoto veřejného klíče, protože je posílán přes ne úplně bezpečné kanály. Útočník, který chce odposlouchávat tuto komunikaci, může podstrčit své klíče, proto se místo veřejných klíčů distribuují certifikáty. [4, 10]

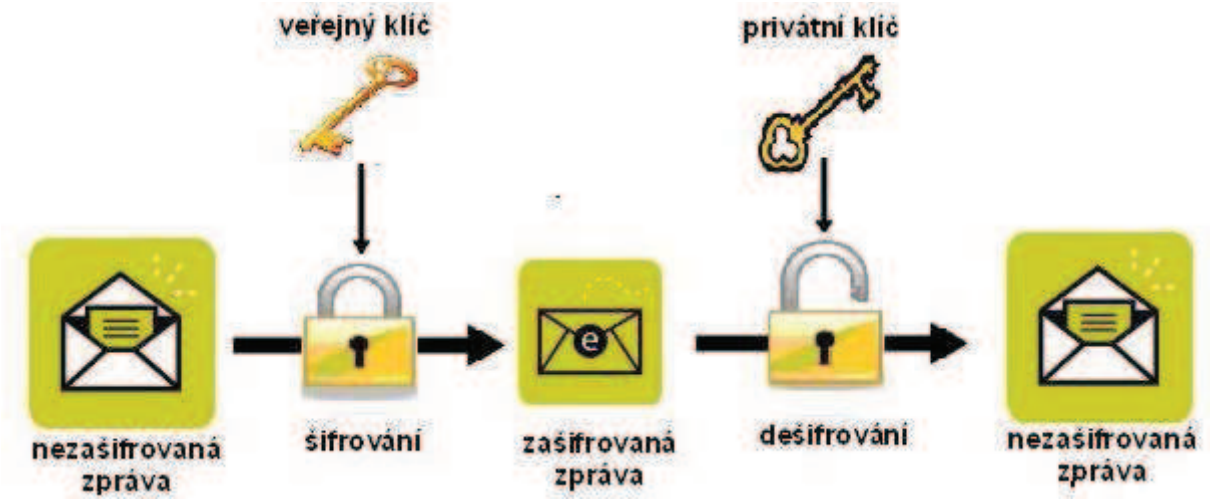
Certifikát je veřejný klíč s informacemi o majiteli veřejného klíče, vydavateli certifikátu, platnosti a tak dále, který je podepsán certifikační autoritou. Certifikační autorita ručí za správnost údajů v certifikátu.

Soukromý a veřejný klíč musí spolu matematicky souviset, ale nesmí být možné z veřejného klíče ten soukromý odvodit. Celková bezpečnost přenosu dat závisí na dobrém šifrovacím algoritmu a klíči. Oba klíče musí mít dostatečnou délku, aby jejich zjištění užitím hrubé síly zabralo roky a více. Šifrovací funkce jsou jednocestné, to znamená, že z dat a klíče získáme šifrovaná data, která nelze stejnou funkcí a klíčem zpětně dešifrovat [4, 17].

Asymetrické šifrování je v současnosti používanější z důvodu nepotřeby předávat sdílený klíč bezpečnou cestou, kterou by nikdo nemohl odposlechnout. Zjištění veřejného klíče útočníkem neznamená velký bezpečnostní problém. Pro komunikující strany je důležité,

aby měly jistotu, že odesílající strana šifruje data klíčem příjemce. Docílit toho není zrovna jednoduché, ale existují způsoby, jak tento problém řešit. Komunikující strany si poté mohou být jisté, že je jejich sdílená data nikdo jiný nezná.

K šifrování dat a k digitálním podpisům se nejčastěji používá algoritmus RSA (iniciály autorů Rivest, Shamir, Adleman) [3, 4].



Obr. 2: Asymetrické šifrování

2.3 Hybridní šifrování

Hybridní šifrovaní se snaží využít výhod a odstranit nedostatky symetrického a asymetrického šifrování. U asymetrického šifrování eliminuje výpočetní náročnost klíče a u symetrického šifrování odstraňuje bezpečnostní riziko přenosu šifrovacího klíče. Výhody hybridního šifrování tedy jsou rychlost a použitelnost.

Princip hybridního šifrování je založen na tom, že odesílatel zvolí nějaký klíč, kterým daná data symetricky zašifruje. Tento klíč se zašifruje veřejným klíčem příjemce a pošle společně s daty příjemci. Příjemce obdrží asymetricky zašifrovaný klíč a symetricky zašifrovaná data. Asymetricky zašifrovaný klíč se dešifruje svým privátním klíčem, který se pak použije k dešifrování textu [10].

Mezi zástupce hybridního šifrování patří šifrovací program Pretty Good Privacy (PGP).

2.4 Technologie AAA

V současných počítačových sítích musíme svá data chránit před zneužitím. Tato ochrana nám přináší mnohá omezení, přitom nezáleží jakou roli v síti zastáváme, protože pro přístup k určitým službám přes počítačovou síť potřebujeme vždy tři věci, které se společně označují zkratkou AAA odvozenou z jejich anglických názvů:

- Autentizace (Authentication)
- Autorizace (Authorization)
- Účtování (Accounting)

2.4.1 Autentizace

Autentizace má za úkol zajistit, aby daným uživatelem sítě a jejích prostředků byl skutečně ten, za koho vydává. Tato funkce znemožní neoprávněným osobám přístup k síti. Autentizace je dosaženo pomocí zadání určité identity (např. uživatelské jméno) a tajemství nebo pověření (např. hesla). Díky autentizaci administrátor sítě jednoduše pozná, kdo se k danému zařízení přihlásil [1, 19].

2.4.2 Autorizace

Po autentizaci následuje autorizace. Autentizovanému uživateli musíme přiřadit určitá oprávnění k provedení požadovaných operací. Autorizaci zajišťují přístupové seznamy nebo zásady. Pomocí autorizace má administrátor kontrolu nad úrovní přístupu, kterou uživatel bude disponovat po úspěšném autentizování. Autorizaci můžeme založit na určitých omezeních, například omezení přihlášení pouze v určitých hodinách, omezení vícenásobného přihlášení jednoho uživatele a podobně [1, 19].

2.4.3 Účtování

Po úspěšném dokončení autentizace a autorizace uživatele nastupují procesy účtování. Pomocí těchto procesů může administrátor shromažďovat informace o uživatelích, kteří jsou přihlášení k daným zařízením a jaké operace tito uživatelé prováděli. Informace touto cestou získané mohou být použity při správě, plánování, účtování a při jiných účelech [1, 19].

2.4.4 Zabezpečovací servery

Po konfiguraci AAA mechanismů může k neoprávněnému přístupu k síti bránit server zabezpečení, pracující nad externím bezpečnostním protokolem. Tyto servery nejčastěji využívají protokoly RADIUS a TACACS. Pokud se pro tyto servery rozhodneme, musíme také spustit mechanismy technologie AAA, protože právě sem se informace z AAA procesů odesílají a zde podléhají dalšímu zpracování [1].

2.5 Šifrovací algoritmy

2.5.1 Data Encryption Standard (DES)

Data Encryption Standard byl vyvinutý v sedmdesátých letech minulého století a řadí se mezi symetrické šifrovací algoritmy. Tato šifra pracuje v blokovém režimu, kdy každý blok má pevnou délku 64 bitů z toho je 8 bitů kontrolních a 56 efektivních. To znamená, že klíč má po dešifrování 56 bitů a může mít podobu jedné z 2^{56} kombinací.

Tento šifrovací algoritmus se již nedoporučuje používat, protože obsahuje určité slabiny a byl již prolomen. DES šifru lze prolomit útokem hrubou silou (Brute-force) za méně než 24 hodin [2, 3].

2.5.2 Triple Data Encryption Standard (3DES)

Algoritmus 3DES je odvozen z původního algoritmu DES, který se aplikuje třikrát po sobě pokaždé s jiným klíčem, tím zvyšuje odolnost šifry proti prolomení. Efektivní délka klíče je zvýšena na 168 bitů. Postup při šifrování dat pomocí algoritmu 3DES probíhá následovně. Původní data se zašifrují pomocí prvního klíče, výsledek tohoto šifrování se zašifruje pomocí druhého klíče a výsledek se opět zašifruje tentokrát pomocí třetího klíče.

Každý blok dat je nutné zašifrovat třikrát po sobě, z toho vyplývá, že algoritmus 3DES je pomalejší oproti svému předchůdci, ale pokud dobře zadáme všechny tři klíče, získáme o několik řádů silnější šifrování než při DES. Používat u všech tří kroků shodný klíč je nevhodné, protože bychom dostali pouze pomalou verzi algoritmu DES bez bezpečnějšího šifrování. Algoritmus 3DES je v současnosti hodně rozšířen a používá se velice často [1, 3].

2.5.3 Advanced Encryption Standard (AES)

Advanced Encryption Standard se řadí mezi symetrické šifrovací algoritmy. Byl vytvořen také jako nástupce algoritmu DES a vychází z Rijndaelova algoritmu. AES pracuje v blokovém režimu. Bloky, které tato metoda šifruje, mají pevnou délku 128 bitů. Pokud jsou šifrovaná data delší, musí se zpracovat po jednotlivých blocích, naopak pokud jsou kratší, musí se do odpovídající délky doplnit (tzv. padding). Délka klíče tohoto algoritmu je 128, 196 nebo 256 bitů, což poskytuje obrovský výběr možných klíčů. Výhoda tohoto algoritmu spočívá také v jeho rychlosti, což je výhodné při šifrování velkého objemu dat. Nevýhodou AES algoritmu je sdílený klíč, který si musí komunikující strany bezpečně předat.

Advanced Encryption Standard je v současnosti hojně využíván v množství komunikačních systémů a protokolů [3].

2.5.4 Message Digest 5 (MD5)

Algoritmus otisku zprávy Message Digest 5 je jedna z metod pro zajištění zabezpečení datové komunikace. MD5 byl vytvořen v devadesátých letech Ronem Rivestem. Algoritmus MD5 pracuje na principu generování ze vstupních dat daný výstup, který má pevnou délku. Tento výstup se nazývá hash (otisk). MD5 je jednosměrný hashovaní algoritmus, který převezme data, která mají libovolnou délku a vytvoří z nich 128 bitový nevratný hash. Jedná se o jednosměrný nevratný hash proto, protože žádnou jeho zpětnou analýzou nelze zpětně odvodit jeho původní obsah. Pokud máme dvě zprávy, které se mění pouze v jednom jediném bitu, hashe budou díky tzv. lavinovému efektu při výpočtu podstatně rozdílné. Je nutné podotknout, že algoritmus MD5 data nijak nešifruje ani nepozměňuje. Pouze vytvoří jakýsi otisk, podle něhož je možné poznat autenticitu (pravost odesílatele) a integritu (data nebyla během přenosu pozměněna) přijatých dat. Neporušenost zprávy se kontroluje tak, že se mezi sebou porovná vypočtený otisk s přijatým. Toto porovnání se označuje jako kontrola hashe. Samotný hash se často označuje jako kontrolní součet. Porovnáním vypočteného a přijatého hashe získáváme určitou ochranu například při stahování software. Získáváme jistotu, že když stahujeme například záplatu pro náš operační systém, nestahujeme zároveň s požadovaným souborem i trojské koně a viry.

Podobně je definován i digitální podpis. Digitální podpisy se používají především v elektronické komerci a jsou důležitou součástí většiny autentizačních schémat [1, 5].

2.5.5 RSA

Algoritmus RSA (zkratka podle inicálů autorů Rivest, Shamir, Adleman) vznikla v sedmdesátých letech a je prvním algoritmem, který je vhodný pro podepisování i pro šifrování dat.

Princip zabezpečení algoritmu RSA je postaven na předpokladu, že rozložit velké číslo na součin prvočísel (tzv. faktorizace) je velice obtížné. Pokud máme velké číslo $z = x \cdot y$, zjistit činitele x a y v rozumném čase je téměř nemožné, naopak násobení dvou velkých čísel je základní úloha.

Algoritmus RSA šifruje data pomocí veřejného klíče. Tento klíč musí mít značnou délku, používají se hlavně délky 1024, 2048 i 4096 bitů. Délka klíče způsobuje zpomalení algoritmu, proto je tento algoritmus vhodný spíše pro přenos klíčů u symetrického šifrování nebo pro šifrování krátkých zpráv [2, 5].

2.5.6 Rivest Cipher 4 (RC4)

Algoritmus RC4 se řadí se mezi nejpoužívanější proudové šifry, které se používají v internetu a v komerčním užití. Tato šifra nemá jednoznačně určenou délku bloku dat. Vyznačuje se svou jednoduchostí a rychlostí. Tato šifra má také své slabé stránky, které argumentují proti jejímu použití v nových systémech. Klíč tohoto algoritmu může mít

maximálně 2048 bitů. Šifra pracuje na principu míchání bytů klíče spojenou s permutací klíče. Algoritmus RC4 se používá například v nedostatečně chráněném kryptosystému jako je Wired Equivalent Privacy (*WEP)* [1].

2.6 Bezpečnostní protokoly

2.6.1 Secure Shell (SSH)

Protokol SSH je zabezpečený protokol v komunikačních sítích využívající TCP/IP model. Slouží pro přihlašování a spouštění příkazů na vzdálených počítačích a síťových zařízeních. Protokol SSH byl navržen jako náhrada za protokol Telnet a podobné nezabezpečené vzdálené shelly (rsh, rlogin) posílající hesla v nezabezpečené formě a tím umožňující odposlech hesla při přenosu sítí nebo internetem. Protokol pracuje na principu klient server. Komunikace mezi SSH klientem a SSH serverem je šifrovaná. Protokol SSH umožňuje bezpečnou komunikaci mezi dvěmi počítači nebo síťovými zařízeními, používanou pro přístup k příkazovému řádku a k přenosu souborů. Zabezpečuje autentizaci komunikujících účastníků, šifrování přenášených dat, integritu dat a kompresi.

Existují dvě verze protokolu SSHv1 a SSHv2. Pokud je to možné, doporučuje se používat protokol SSHv2, protože používá bezpečnější šifrovací algoritmy než SSHv1. SSH podporuje Data Encryption Standard (DES) algoritmus, algoritmus 3DES a uživatelskou autorizaci založenou na vkládání hesel. K implementaci SSH potřebujeme vygenerovat RSA klíče. RSA zahrnuje veřejný klíč držený na veřejném RSA serveru a privátná klíč, držený pouze odesílatelem a příjemcem.Veřejný klíč může být znám každému a je použit pro šifrování zprávy. Zprávy šifrované veřejným klíčem mohou být dešifrovány pouze privátním klíčem. Toto je známo jako asymetrické šifrování [1, 3, 4].

2.6.2 Secure Sockets Layer (SSL)

Protokol SSL řeší bezpečnost protokolů TCP/IP na transportní vrstvě. Úkolem SSL je poskytnout bezpečný komunikační kanál mezi dvěmi zařízeními v síti Internet na úrovni spojení TCP/IP, který umožní bezpečně implementovat všechny běžné nezabezpečené protokoly (ftp, telnet, http, atd.). Tento protokol může pro ustanovení klíče relace použít různé algoritmy s veřejným klíčem. Po ustanovení relace se další komunikace zabezpečuje šifrováním pomocí některého algoritmu s privátním klíčem.

Protokol SSL je snadno rozšiřitelný o nové kryptografické algoritmy a interoperabilní, což znamená, že schopný efektivně spolupracovat a poskytovat si funkce s jinými protokoly. Základní pojem protokolu SSL je relace. Relace představuje určité spojení mezi klientem a serverem na úrovní transportní vrstvy. Při ustavení relace se může provádět autentizace uživatele a v rámci jedné relace může být současně vytvořeno několik zabezpečených spojení. Každá relace má své stavové informace včetně identifikátoru dané relace, informacích o kompresi, o kryptografických algoritmech atd. Je zde uložen i klíč relace. [3, 4]

Klient, který chce použít SSL, nejdříve kontaktuje server a vyjedná s ním parametry relace (identifikace, verze protokolu, komprese, šifrování, atd.). Komunikující strany si vymění certifikáty svých veřejných klíčů podepsané elektronickým podpisem. Klient vygeneruje klíč relace, zašifruje ho veřejným klíčem serveru a pošle ho serveru společně s náhodnou výzvou. Server dešifruje klíč relace pomocí svého soukromého klíče a autentizuje se klientovi vrácením jeho náhodné výzvy zašifrované klíčem relace [1, 4, 18].

Adresy webových stránek zabezpečených pomocí SSL začínají https:// . Zabezpečené webové stránky je možné poznat také tak, že v prohlížeči se ve stavové liště (nově i v řádku adres) zobrazí ikona zámku. HTTPS/SSL dokáže zajistit důvěrnost dat při transferu mezi klientem a serverem. Port protokolu HTTP je standardně 80. Port protokolu je HTTPS je 443.

3 Základní typy útoků

Počítačové sítě, její zdroje a komponenty dokonce i přenášená data se mohou stát cílem útoku hackera. Hacker má na výběr z velkého množství různých typů útoků. Aby byl administrátor sítě schopen ubránit svou síť proti těmto útokům, musí důkladně znát principy a důsledky každého z nich. Útočník se pokusí využít každého slabého místa v síti, aby dosáhl svého cíle. Útočník může poměrně snadno využít slabé autentizace a autorizace, nedostatečně implementovaného zabezpečení, nesprávného přidělování prostředků nebo i špatných pracovních návyků zaměstnanců. Každá z těchto nebo dalších jiných chyb vede k získání neoprávněného přístupu k síťovým prostředkům.

Během let hackeři vymysleli nebo objevili spoustu různých způsobů, jak obejít i velice propracované bezpečností postupy a technologie. Pokud osoba odpovědná za bezpečnost systémů nemá dostatečné znalosti o metodách a nástrojích vedení útoku proti síti, může se stát, že bezpečnostní technologie, zásady a postupy, mohou ztratit svou účinnost. Proto je důležité vědět, jaké metody a nástroje útočníci používají, abychom mohli úspěšně proti nim bojovat [1].

3.1 Falšování IP adres a únosy relace

Při tomto typu útoku útočník, aby se mu podařilo vstoupit do systému, vytváří paket s jinou zdrojovou IP adresou, než kterou má ve skutečnosti přidělenou. Útočník využije důvěry systému, jelikož se vydává za důvěryhodného uživatele. Aby se tento útok mohl podařit, musí útočník získat informace o rozsahu povolených IP adres. Pokud útočníka systém přijme, může útočník následně pokračovat a tento hostitelský systém napadnout nebo ho vyřadit z provozu. Tento typ útoku se obvykle používá pouze jako první krok při rozsáhlých útocích.

Obrana proti tomuto typu útoku spočívá v zavádění virtuálních privátních sítí (VPN), které své IP adresy šifrují. Pakety, které mají pozměněnou zdrojovou nebo cílovou adresu, se hned odstraní. Aby útočník mohl proniknout do systému, musí znát šifrovací klíče VPN [1].

3.2 Denial of Service (DoS)

Útok Denial of Service se může do češtiny přeložit jako útok s odepřením služeb. Dnes se tento typ útoků označuje spíše jako distribuovaný útok s odepřením služeb (Distributed Denial of Service) nebo paketová bouře (Paket Storm) nebo zaplavení sítě (Tribal Flooding). Základní vlastnost tohoto útoku spočívá vždy v přetížení sítě obrovským množstvím požadavků, kdy následně dojde ke zpomalení nebo dokonce zahlcení sítě a síť nezvládne obsloužit ani právoplatný provoz.

Pokud cíl útoku přestává zvládat provoz, dochází k odepření služeb, protože ani právoplatní uživatelé nejsou schopni se k tomuto cíli připojit. Distribuovaný útok s odepřením služeb (DDoS), pro umocnění útoku, zaplavuje cíl útoku generováním falešného provozu z více hostitelských systémů z různých lokalit. Proto prvním krokem při vedení DDoS útoku je napadení určitého počtu cizích počítačů, do kterých útočník nainstaluje programové démony DDoS a vytvoří z těchto počítačů takzvané zombie.

Démon DDoS je počítačový program, který má za úlohu řídit a koordinovat DDoS útok. Jedná se například o programy Tribal Village (TFN), TFN2K a Trinoo.

Takto napadené počítače (zombie) vytváří velkou síť zombií, které potom posílají požadavky na spojení a rapidně zvýší množství otevřených spojení u cílového počítače (oběti). Vedení útoku s odepřením služeb je poměrně snadné, ale díky němu je možné napáchat obrovské škody, hlavně když se útočníkům podaří vyřadit důležitý server nebo se jim podaří odstřihnout celou podnikovou síť od internetu.

DoS útoky se provádí pomocí různých technik vytváření falešného spojení. Platí obecná poučka, že DoS útoky zneužívají slabá místa v architektuře sítě [1].

3.2.1 Útok se záplavou paketů ICMP

Při útoku se záplavou paketů ICMP útočník využije vlastnosti protokolu ICMP, kdy napadané hostitelské systémy neboli zombie začnou odesílat nepřetržitý tok paketů ping danému cíli. Cíl útoku je doslova bombardován požadavky ping, které mohou být vedeny od stovek zombií. Navíc pokud útočník zároveň mění pakety ICMP s požadavkem na opakování (echo) a do požadavku umístí místo IP adresy napadeného hostitele IP adresu cílového systému (zdrojová i cílová IP adresa ukazuje na oběť útoku), oběť začne na požadavky odpovídat (echo reply), jenže odpovědi vedou opět k oběti, provoz oběť začne postupně zavalovat, dokud nezhavaruje. Útočník může zdrojovou adresu změnit na broadcast adresu, takže provoz zahltí i ostatní síťové prostředky v dané lokální síti.

Tento útok využívá faktu, že jeho provoz se tváří jako standardní povolený provoz, proto jej firewally pustí do vnitřní sítě, kde začne páchat škody [1].

3.2.2 Útok se záplavou paketů SYN

Pro dobré pochopení tohoto útoku je nutné znát, jak se vytváří TCP spojení. Vždy když se chce klient připojit k určité službě (např. FTP, HTTP), navázání komunikace se provede pomocí tree-way handshake.

Three-way handshake probíhá následovně (cit. z lit. [1]):

Klient odešle poskytovateli služby paket, v jehož hlavičce TCP je nastaven synchronizační příznak SYN. Služba odpoví paketem s potvrzením synchronizace (SYN-ACK). Nakonec klient odešle paket s navázáním komunikace (SYN-ACK).
Po těchto krocích je komunikace navázána a může dojít k vlastnímu přenosu dat.

Útok se záplavou paketů SYN je založený na tom, že útočník vyšle množství paketů SYN, služba na ně odpoví synchronizačními pakety SYN-ACK, ale klient už na tyto pakety neodpoví, protože jeho IP adresa je falešná. Služba si vyhradí prostředky na tato spojení a pak už nemůže obsloužit právoplatná spojení, čímž dojde k odepření služeb. Útočník musí posílat pakety SYN poskytovateli služby rychleji, než vyprší časový limit pro odpovědi. Tím dojde ke stavu, kdy se služba přetíží odpovídáním na pakety SYN a bude čekat na potvrzení od klientů, kteří je ale již neodešlou.

3.2.3 Obrana proti útokům s odepřením služeb

Proti útokům s odepřením služeb se dá velice špatně bránit. Jedná se o útoky, které jsou z pohledu obrany jedny z nejobtížnějších, protože většina útoků využívá normální provoz, který se v dané síti standardně vyskytuje. Jako nejčastější mechanismus obrany proti DoS útokům je pravidelné sledování rychlosti určitého druhu provozu. Například povolíme pouze určitý počet dotazů na danou službu. Určitou hranici dotazů za jednotku času budeme považovat již za útok a dané dotazy zakážeme. Je nutné pečlivě sledovat jednotlivé typy provozu, protože příliš velké omezení například webové provozu z Internetu do stránek elektronické komerce, by se nemuselo vyplatit. Proto je nutné obranné kroky vždy uvážit [1].

3.3 Odposlech paketů (sniffing)

U tohoto typu útoku útočník používá nástroje, které odposlouchávají síťový provoz. Těmto nástrojům se říká sniffery. Sniffery zachytávají pakety procházející místem jejich připojení. Může se jednat o nástroje softwarové i hardwarové, které dokáží odposlechnout a dekódovat data všech sedmi vrstev modelu OSI. Útočník je pomocí takového snifferu schopen jednoduše odposlechnout uživatelská jména a hesla, pomocí kterých následně mohou vést další útoky. Při tomto útoku se musí útočník obvykle osobně dostat do firemních prostor a připojit se například notebookem do sítě. Díky bezdrátovému připojení k internetu, může útočník odposlouchávat síťovou komunikaci, například v autě před podnikem. Útočník pak jednoduše zachytí uživatelská jména a hesla u služeb, které nepodporují šifrovanou komunikaci (například FTP, Telnet). Za velmi zranitelné se považují protokoly SMTP, IMAP,POP3, protože používají jednoduchou autentizaci pomocí uživatelského jména a hesla. Jelikož si uživatelé často nechtějí pamatovat různá uživatelská jména a hesla pro různé aplikace, používají jedno heslo, což dává útočníkovi přístup i do zabezpečených aplikací a jiných síťových prostředků [1, 15].

3.4 MAN-IN-THE-MIDDLE

Jeden ze základních typů útoku je útok muž uprostřed neboli man in the middle (MITM). Tento útok je také znám jako zrcadlový útok nebo spoofing. Man in the middle útok je založen na přesměrování provozu mezi dvěma komunikujícími stranami tak, aby procházel přes počítač útočníka. Útočník si daný provoz přečte a přepošle právoplatnému

příjemci. Pokud útočník přestane přeposílat daný provoz, data k příjemci nikdy nedorazí. Díky tomuto útoku může útočník izolovat cílové hosty od sítě, právě díky nepřeposílání daného provozu těmto cílovým hostům a tím znemožní cílovým hostům komunikovat s okolím. Správce sítě bude jen velice těžce odhalovat, proč cílový host není schopen komunikace.

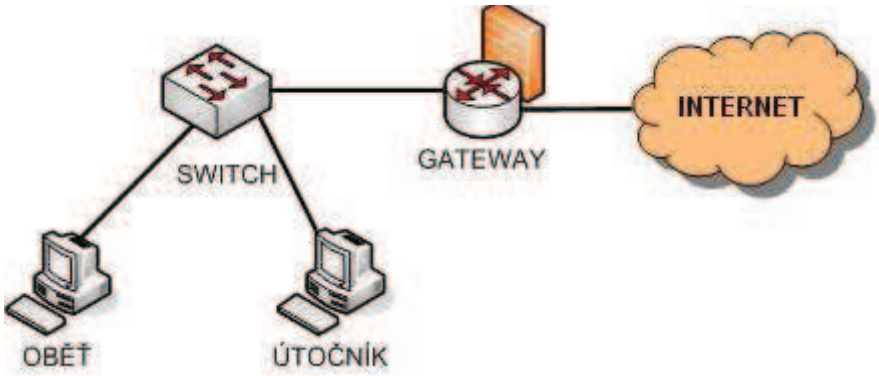
3.4.1 ARP Spoofing

ARP Spoofing (známé také jako ARP Cache Poisoning) je technika útoku využívající slabin protokolu ARP (Address Resolution Protocol), který byl navrhován v době, kdy se o bezpečnost sítě ještě moc nestaralo. Proto protokol ARP nemá žádné ochranné mechanismy.

Protokol ARP slouží pro překlad logických adres (IP adres) na fyzické adresy (MAC adresy). Uživatel ve většině případů zná pouze IP adresu příjemce. Protokol ARP, na základě této IP adresy počítače příjemce, vyhledá příslušnou MAC adresu. Vyhledání se provede pomocí zprávy ARP request. Na tuto zprávu odpoví počítač, který má IP adresu shodnou s IP adresou umístěnou v této žádosti, pomocí zprávy ARP Reply. Na základě této zprávy se odesílatel žádosti dozví fyzickou (MAC) adresu příjemce.

Hlavním cílem tohoto útoku je dostat data z počítače oběti k počítači útočníka, aby si je mohl prohlédnout. Je nutné také podotknout, že při tomto útoku se útočník se svým zařízením musí nacházet v místní síti LAN, která je propojená pomocí switchů. Switche mají v sobě zabudovanou paměť (tzv. ARP Cache), do které si ukládá MAC adresy koncových zařízení, které jsou k němu fyzicky připojeny. Když na některý port switche dorazí rámec obsahující ve svém záhlaví cílovou MAC adresu, switch se podívá do své paměti a na základě záznamů v tabulce rozhodne, na který port rámec přepne.

Pro vysvětlení ARP Spoofing útoku uvažujme následující síť (viz Obr. 3), která obsahuje PC oběti, PC útočníka, switch a bránu (gateway). Každý počítač, který chce vysílat do internetu, musí svůj provoz posílat přes danou bránu. [11, 12, 15]



Obr. 3: Uvažovaná síť při ARP Spoofing útoku

Vlastní útok probíhá následujícím způsobem. Útočník pošle oběti zprávu, která říká, že brána má MAC adresu útočníka a bráně pošleme zprávu, která zase říká, že naše oběť má MAC adresu stejnou jako útočník. Tímto krokem si oběť upraví záznam v tabulce ARP Cache a změní v tomto záznamu MAC adresu brány na MAC adresu útočníka. Brána si změní záznam o MAC adrese oběti a zamění ji s útočníkovou MAC adresou. Tímto způsobem útočník docílí, že komunikace mezi obětí a bránou bude procházet přes útočníka, který si obsah komunikace prohlédne (popřípadě změní) a pošle ji správnému příjemci. Při tomto útoku útočník využije slabiny protokolu ARP, protože si oběť i brána upraví záznamy v ARP Cache a vůbec si nehlídají, zda podali žádost o zjištění MAC adresy toho druhého zařízení.

Podmínka pro tento útok je, aby se záznamy, které jsou požadovány změnit, v tabulce ARP Cache už vyskytovaly. Samozřejmě tohle také není pro útočníka problém. Pokud v tabulce záznam chybí, může například vyslat zprávu ping s falešným odesílatelem. Uživatel nebo brána si IP adresu a MAC adresu z této zprávy uloží do své tabulky. Záznamy v tabulce ARP Cache po určité době smazávají, proto útočník informace o falešných MAC adresách posílá oběti a bráně periodicky po určitém čase.

Obrana proti ARP Spoofing je zavedení statických tabulek. Záznamy se do ARP Cache tabulky zadají staticky. Od této chvíle bude switch zprávy ARP Reply ignorovat a bude se řídit pouze statickými záznamy. Díky tomu není možné tyto záznamy změnit [11, 12].

3.4.2 DHCP spoofing

Tento útok je založen na faktu, že v jedné síti může být přítomno více DHCP (Dynamic Host Configuration Protocol) serverů a tyto servery nepracují příliš rychle. Cílem útoku DHCP spoofing je nastrčit a zprovoznit do sítě nový DHCP server a jakmile uživatelský počítač zažádá o přidělení dynamických síťových parametrů, DHCP server útočníka odpoví na tuto žádost a přidělí mu podstrčené údaje. V těchto údajích, které útočník uživateli podstrčil, může být falešná brána (gateway) nebo DNS server. Důležité pro tento útok je, aby DHCP server útočníka odpověděl na žádost uživatele dříve než právoplatný DHCP server.

Podstrčí-li útočník uživateli pouze falešnou gateway (PC útočníka), komunikace uživatele bude probíhat tak, že odchozí provoz mimo (například podnikovou) síť bude procházet přes počítač útočníka, ten si přečte a následně ji pošle na právoplatnou gateway a dále do internetu. Provoz vracející se z internetu už ale bude posílán z opravdové gateway přímo právoplatnému uživateli. Tedy útočník nebude moci číst příchozí informace z internetu.

Podstrčíme-li uživateli i falešný DNS server, můžeme vytvořit útok, který bude zachytávat oba směry provozu. Této skutečnosti se docílí tím, že na všechny dotazy bude útočník odpovídat svou IP adresou a ze svého počítače vytvoří jakoby proxy server.

U DHCP spoofing útoku musíme rozlišovat situace, které mohou nastat a to je, že se uživatel přihlásí do sítě poprvé nebo již byl uživatel v síti připojen. Podle dané situace se útok musí přizpůsobit. [11, 12, 15]

V prvním případě, kdy se uživatel přihlásí do sítě poprvé, vyšle uživatel do sítě zprávu DHCP Discover. Tato zpráva se vyšle broadcastově a žádá, aby mu odpověděli DHCP servery v síti. Každý DHCP server, ke kterému tato žádost dorazí odpoví zprávou DHCP Offer, kterou nabídne uživateli síťové parametry. Uživatel zažádá o přidělení síťových parametrů zprávou DHCP Request ten DHCP server, od kterého přišla nabídka jako první. DHCP server následně pošle uživateli potvrzení (zpráva DHCP Ack). U toho případu, musí být útočníkův DHCP server nejrychlejší.

V druhém případě, kdy uživatel již byl v síti připojen, je postup jiný. Uživatelský počítač pošle DHCP serveru, od kterého naposled obdržel síťové parametry, zprávu pouze DHCP Request. V této zprávě uživatel žádá o IP adresy, kterou měl přidělenou naposledy. DHCP server uživateli na tu to žádost odpoví potvrzením DHCP Ack nebo může žádost zamítnout a přidělit mu jinou IP adresu. Výměnu těchto zpráv mezi uživatelem a DHCP serverem, útočník nezachytí. Tento problém musí útočník vyřešit tím, že vyčerpá všechny IP adresy, které DHCP server může uživatelům přiřadit. Jakmile DHCP server nemá žádné volné IP adresy pro přiřazení, přestane odpovídat na zprávy DHCP Discover. Následně musí útočník čekat, než uplyne doba propůjčení (lease time) již propůjčených IP adres uživatelů a následně, útočník zabere i tyto IP adresy. Od této doby, když uživatel požádá o svou starou IP adresu, nedostane žádnou odpověď nebo mu přijde odpověď o odmítnutí žádosti. V této situaci uživatel vyšle do sítě zprávu DHCP Discover, jako kdyby v síti ještě nikdy nebyl. V tuto chvíli může IP adresy poskytnout falešný DHCP server a všichni uživatelé potom budou využívat služeb falešného DHCP serveru.

Účinná obrana proti útoku DHCP spoofing je možná pomocí funkce DHCP snooping. Tato služba je založená na důvěryhodných a nedůvěryhodných portech. Pokud se na port switche připojí DHCP server nebo jiný switch podporující tuto službu, přiřadí se danému portu stav důvěryhodný. Všechny ostatní porty se pak označí jako nedůvěryhodné. Pokud uživatel požádá o IP adresu z falešného DHCP serveru, připojeného na nedůvěryhodném portu, switch to rozpozná a tuto zprávu zahodí. Další metoda zabránění útoku je statické definování síťových prostředků na uživatelských stanicích, což není pohodlné řešení, ve středních a velkých firmách prakticky nemožné [11, 12].

3.4.3 DNS spoofing

Útok DNS spoofing spočívá v podvržení IP adresy v paketu vracejícího se jako odpověď na žádost o překlad doménového jména na IP adresu. Při tomto útoku, může být provoz sítě přesměrován i mimo lokální síť, což při útocích na ARP protokol (ARP Spoofing) není možné. Zde bude popsán útok na koncového uživatele uvnitř lokální sítě.

Koncový uživatel používá k překladu doménových jmen DNS Resolver. DNS Resolver se dá popsat jako sada požadavků sloužící k práci s DNS protokolem. Při překladu doménového jména položí DNS Resolver požadavek na DNS server, který má daný uživatel staticky nebo dynamicky nastaven na své počítači. Když uživatel dostane odpověď, uloží si ve své lokální DNS Cache pro případ opětovného použití. Doba, po kterou bude záznam uložen

v DNS Cache, se deklaruje na DNS serveru, který je pro danou doménu autoritativní a je obsažena v odpovědi DNS serveru. Jakmile tato doba uplyne, je záznam z DNS Cache smazán. U protokolu DNS je tomu podobně jako u protokolu DHCP, kdy uživatel použije odpověď serveru, od kterého přijde první. Útočník při DNS spoofing útoku musí odpovědět na dotaz dříve, než přijde odpověď od právoplatného DNS serveru. [11, 12]

Aby útočník mohl zfalšovat DNS odpověď musí znát určité informace. Protokol DNS využívá k přenosu svých zpráv jak protokol TCP, tak i protokol UDP. Primárně se používá protokol UDP, protokol TCP se používá v případě, že se odpověď nevejde do jednoho paketu. Pro útočníka je výhodnější, když je použit protokol UDP, protože mu odpadne práce se zjišťováním sekvenčních čísel a podobně. Další potřebné informace, které musí znát jsou IP adresa DNS serveru, jehož služeb oběť útočníka využívá. Zjistit tuto IP adresu není problém, protože je v lokální síti nastavena na všech počítačích. Dále potřebuje útočník vědět, na kterou doménu se útočník dotazuje, protože může zfalšovat pouze IP adresu pro tuto doménu. Navíc ještě útočník musí znát port, ze kterého se DNS Resolver dotazuje ID (identifikátor) dotazu. ID je číslo složené ze 16-ti bitů a je náhodně generováno. Pomocí ID se páruje dotaz a odpověď, proto se použije v DNS dotazu a musí být uvedené i v DNS odpovědi. Útočník má výhodu, že většina implementací Resolveru nepožaduje všechny tyto informace. Informace, které potřebuje znát vždy je port a ID. [11, 12]

Číslo portu se dá zjistit, pokud nestojí v cestě firewall, například skenováním portů nebo je možnost útočit na více portů v určitém předpokládaném rozsahu. V tomto případě, když se útočí na DNS server, je možné jednoduše zjistit port i aktuální ID dotazu požádáním DNS serveru o překlad adresy, kterou útočník vlastní. Na DNS serveru pustíme sniffer a útočník si data jednoduše odchytí.

Existuje několik metod, jak zajistit, aby odpověď útočníka dorazila k uživateli dříve než z legitimního DNS serveru. Převážně se jedná o DoS útoky snažící se server vyřadit z provozu nebo ho přetížit, aby zpracovával požadavky pomalu [11, 12].

Man in the middle útok na koncového uživatele.

Útočník musí zjistit alespoň port dotazu. Následně spustí DoS útok proti DNS serveru, čímž si zajistí více času pro poslání falešné odpovědi. Nyní útočník začne posílat do nekonečna falešné DNS odpovědi a bude v nich neustále měnit číslo ID. Předpokládá se, že resolver nekontroluje, zda-li je v odpovědi vrácen překlad doménového jména, o který žádal (tuto chybu obsahovali i Windows XP SP1). Pak už jen útočník čeká, kdy uživatel požádá o překlad nějakého doménového jména [12].

Man in the middle útok na koncového uživatele s použitím Additional records.

Efektivnější varianta předchozího útoku, provedená přidáním dalších adres, které chceme zfalšovat. Tyto adresy se přidají do pole Additional records, které slouží k urychlení DNS služby tím, že DNS server pošle klientovi kromě překladu doménového jména, na které se dotazoval, i překlady navíc, které s tímto dotazem souvisí. Díky tomu se ušetří další následující dotazy [12].

Man in the middle útok na koncového uživatele s použitím finty.

Útok využívající určitého triku k zefektivnění útoku, například když síť využívá proxy server a klienti mají proxy server specifikovaný místo IP adresou doménovým jménem. Pak stačí podvrhnout pouze překlad doménového jména a útočník získá veškerou http komunikaci oběti v obou směrech [12].

Man in the middle útok na DNS server s použitím vícenásobných dotazů.

Tento útok se objevil díky chybě v DNS serveru Bind, která měla za následek, že pokud přišlo na jedno doménové jméno naráz více dotazů z více IP adres, DNS server zpracoval každou žádost samostatně a každé žádosti přidělil jiné ID. To znamenalo, že při zaslání tisíce žádostí bude existovat tisíc správných ID. Tím útočník získá už velkou pravděpodobnost, že se trefí. Podle nových informací, byla tato chyba již opravena.

Budeme-li ale uvažovat, že DNS server neporovnává žádost s odpovědí u překládaného doménového jména, lze tento útok provést také pomocí zasílání dotazů náhodných doménových jmen a zfalšovaných odpovědí. Jestliže server některou ze zfalšovaných odpovědí přijme, jeho DNS Cache bude otrávena, ze které se následně rozšíří zfalšované překlady k uživateli.[12]

Využití DNS Spoofingu spolu s jiným útokem.

Útočník odposlouchávající provoz v síti, může odposlechnout ID dotazu a zdrojový port. To znamená, že může přesně falšovat odpovědi. Díky tomu může útočník pomocí DNS spoofingu uživatele cíleně přesměrovávat. Útočník si následně počká na dotaz o překlad například stránek banky, pošle uživateli falešnou odpověď a přesměruje uživatele na nešifrovaný server. Uživatel se přihlásí a útočník zachytí jeho přihlašovací údaje [12].

3.5 ICMP Redirect

Útok ICMP Redirect využívá zpráv protokolu ICMP typu 5. Zprávy ICMP typu 5 jsou zprávy, které se používají k optimalizování směrování (routování) dat v síti. Existuje několik podtypů této zprávy, ale nejvíce se používá podtyp přesměrování pro hostitele. Tato zpráva má za úkol, aby když na gateway přijdou data od uživatele určená dále ven ze sítě a tato gateway zjistí, že bude výhodnější (rychlejší) tato data poslat přes jinou gateway, vyšle o tom uživateli ICMP Redirect zprávu. Data, která už přijal, pošle ještě původní cestou. Uživatel si po obdržení této zprávy pozmění směrovací tabulku a směruje provoz na druhou gateway.

Vlastní útok se tedy provede vysíláním falešných zpráv ICMP Redirect. U ICMP Redirect útoku se vyskytuje ještě problém, protože existují určitá pravidla, která musí tyto ICMP zprávy splňovat. Jedním pravidlem je, že tyto zprávy musí obsahovat 8 bytů dat z paketu, který způsobil vygenerování zprávy ICMP Redirect. Tato podmínka je vyžadována pouze v některých systémech. V systému, ve kterém se tato podmínka nevyžaduje, má útočník podstatně větší šanci útok provést [11, 12, 14].

3.6 MAC Flooding

Před vlastním popisem postupu útoku je nutné vědět, že switch přeposílá datové rámce na základě cílové MAC adresy v hlavičce daného rámce. Aby switch věděl, na který port má datový rámec přepnout, má v sobě zabudovanou paměť, ve které se nachází tabulka se záznamy říkající, na který port přepnout datový rámec, aby dorazil na určitou cílovou MAC adresu. Do této tabulky se vejde omezený počet záznamů (tisíce až statisíce). Velikost tabulky se odvíjí od typu switche. Tabulka je nejdříve prázdná, zaplňuje se až na základě přicházejících rámců na jednotlivé porty switche.

Útok MAC Flooding je založen na zaplnění této tabulky. První metodou je, že útočník začne posílat velké množství rámců, které mají zdrojovou i cílovou MAC adresu náhodně vygenerovanou a tím způsobí, že switch si vždy pro každou MAC adresu, kterou ještě nemá v tabulce uloženou udělá nový záznam a daný rámec pošle broadcastově na všechny porty. Díky tomu se zahltí i ostatní switche v síti. [15]

Druhou metodou je, že útočník nastaví rámcům cílovou MAC adresu na svoji a odchozí MAC adresu generuje náhodně. Jakmile switch přijme takovýto rámec, vytvoří si záznam v tabulce, ale zjistí, že příjemce se nachází na stejném portu odkud rámec přišel, takže rámec už nikam neposílá. Takto útočník přijde o zahlcení ostatních switchů v síti, na druhou stranu jeho útok bude možné odhalit jenom velice těžce.

Jakmile je tabulka se záznamy zcela zaplněna mohou nastat dva případy. Některé switche se po zaplnění tabulky ARP Cache přepnou do stavu „fail open“, to znamená, že se začnou chovat jako hub. Příchozí rámec se pošle na všechny porty kromě portu odkud rámec přišel. Jiné switche po zaplnění tabulky záznamy reagují na příchozí rámec, jehož cílová MAC adresa není obsažena v tabulce, stejně jako v předchozím případě, to je broadcastově odešle rámec na všechny porty, kromě portu odkud rámec přišel. Pokud ale přijde rámec s cílovou MAC adresou, která je obsažena v tabulce, přepne tento rámec pouze na daný port, ke kterému je příjemce rámce připojen.

Záznamy v tabulce ARP Cache jsou po určité době mazány. Každé zařízení může mít dobu, po které se záznam v tabulce smaže, jinou. Administrátor si také může dobu, za jak dlouho se záznamy budou mazat, přenastavit podle svého. Většinou je ale tato doba 300 sekund. Po nějaké době se vymažou záznamy i oprávněných uživatelů a v tuto chvíli musí útočník obsadit místa těchto uživatelů. Když se mu to podaří, veškerá komunikace oprávněných uživatelů bude přicházet i k útočníkovi a ten ji bude moci přijímat a číst. [11, 12]

Pokud se útok provádí stylem, že se nastavuje cílová fyzická adresa na fyzickou adresu útočníka, lze tento útok obtížně vysledovat. Tento útok je možné poznat až nezvykle rychlým blikáním stavové LED diody na switchi nebo u dražších switchů vypsáním tabulky záznamů.

Proti útoku MAC flooding se lze bránit využitím služby DHCP snooping. Tato služba je založená na důvěryhodných a nedůvěryhodných portech. Všechna data, která na switch přijdou z nedůvěryhodného portu, jsou zahazována, pokud není nastavena zdrojová MAC a IP adresa počítače v záznamu DHCP snooping tabulky pro daný port. Při tomto útoku musí útočník měnit MAC adresy, což je odhaleno při porovnávání se záznamy s hodnotami v DHCP snooping tabulce [11, 12].

3.7 Port stealing

Již z názvu tohoto útoku si je možné odvodit, že se bude jednat o krádež portů. Takovéto krádeži může dojít při aktualizaci tabulky v ARP Cache switche při příjmu rámce.

Při útoku se postupuje následovně. Útočník si nejprve zjistí, jakou má oběť fyzickou (MAC) adresu. Jakmile zjistí fyzickou adresu oběti, začne posílat na switch rámce, jejichž cílová fyzická adresa bude rovna fyzické adrese útočníka a zdrojová fyzická adresa bude adresa oběti. Na základě těchto rámců si switch bude myslet, že oběť byla přepojena na port, ze kterého tyto rámce přicházejí, tím pádem si pozmění záznam v tabulce. Protože cílová fyzická adresa se nachází na stejném portu jako fyzická adresa odesílatele, rámce nebudou nikam dále přeposlány. Pokud je oběť připojena na jiný switch, musí útočník cílovou fyzickou adresu nastavit na broadcast. Od této doby, jakmile přijde na switch rámec určený pro oběť, switch přepne tento rámec na port, kde se nachází počítač útočníka. Útočník si příchozí data prohlédne a pokud chce doručit data pravému příjemci, musí opět upravit tabulku se záznamy. Toho útočník dosáhne tím, že přestane vysílat rámce pro ukradení portu a pošle zprávu ARP Request. Oběť odpoví pomocí zprávy ARP Reply. Na základě těchto zpráv si switch danou tabulku opraví do původní podoby před útokem. Potom už ukořistěná data můžeme jednoduše poslat pravému příjemci.

Nevýhoda tohoto útoku spočívá v tom, že jakmile oběť odešle nějaký rámec, tabulka záznamů switche se vždy opraví. Proto útočník musí posílat rámce pro kradení portů rychle za sebou. Občas se tedy může stát, že nějaký rámec útočník nezachytí a dojde správně k oběti.

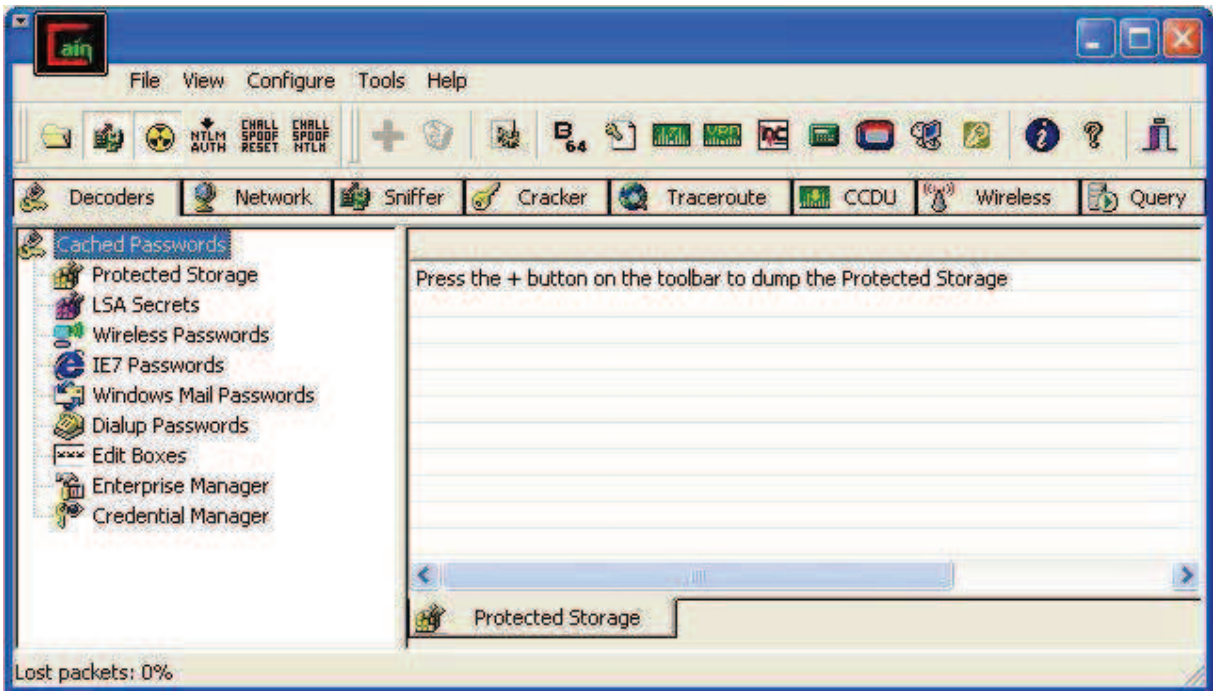
Útok Port stealing lze vysledovat například všimnutím si častých ARP dotazů a nebo podle fyzické adresy příchozích dat. Efektivně lze tomuto útoku předejít, jako při útoku MAC flooding, pomocí služby DHCP snooping [11, 12].

4 CAIN & ABEL

Cain & Abel (viz Obr. 4) je nástroj pracující na operačních systémech Microsoft, který umožňuje jednoduché získání několika druhů hesel pomocí sniffování (sledování síťového provozu) v síti, crackování šifrovaných hesel pomocí slovníkového útoku, útoku hrubou silou a pomocí kryptoanalýzy, dále nahrávání VoIP hovorů, získání klíčů bezdrátových sítí, odhalení hesel skrytých za hvězdičkami a analýzu směrovacích protokolů.

Program Cain & Abel nevyužívá žádných zranitelností software nebo chyb, které by nemohly být s malým úsilím zalátovány. Zahrnuje některé bezpečnostní aspekty/slabiny protokolových standardů, autentizačních metod a ukládacích mechanismů. Jeho hlavním účelem je jednoduché získání hesel a pověřovacích listin od různých zdrojů. Poskytuje také některé nestandardní nástroje pro uživatele operačního systému Microsoft Windows.

Cain & Abel byl vyvinut v naději, že tento nástroj bude užitečný pro síťové administrátory, učitele, bezpečnostní konzultanty, odborníky, prodejce bezpečnostního software, profesionální testery průniku a pro všechny ostatní osoby, kteří neplánují tento nástroj použít z neetických důvodů. Nejedná se tedy o nástroj, který by používaly útočníci pro nabourání se do sítě nebo dokonce vyřazení sítě z provozu, ale spíš o program, který administrátorovi pomůže odhalit slabiny jeho sítě a ukáže mu, jaké citlivé informace se přes jeho síť přenášejí. Dosažené výsledky může následně vyhodnotit a přijmout bezpečnostní opatření. [13].



Obr. 4: Grafické rozhraní programu Cain & Abel

4.1 Instalace

Tento program je poskytován jako freeware, to znamená, že jeho šíření, kopírování, instalování a užívání je zdarma. Program si lze opatřit na internetu na oficiálních stránkách výrobce http://www.oxid.it . Program Cain & Abel se skládá ze dvou částí. Cain je hlavní aplikace s grafickým uživatelským rozhraním a Abel je Windows služba složená dvěmi soubory (Abel.exe a Abel.dll) [13].

Požadavky programu:

10MB volného místa na Hard-Disku
Operační systém - Microsoft Windows 2000/XP/2003/Vista/Windows 7
Winpcap Packet Driver (verze 2.3 nebo vyšší, od verze 4.0 obsahuje i Airpcap)
Airpcap Packet Driver (pro pasivní bezdrátový sniffer/WEP cracker)

Vlastní instalace programu Cain se provede spuštěním samoinstalačního balíku a postupuje se dle instrukcí instalačního průvodce. Balík nakopíruje všechny soubory, které bude program potřebovat do adresáře a podadresářů programu na daném počítači.

Soubory služby Abel (Abel.exe a Abel.dll) se nakopírují do adresáře programu již při instalaci programu Cain, ale služba není automaticky nainstalována do systému. Abel může být instalován lokálně nebo vzdáleně pomocí programu Cain a vyžaduje práva administrátora na daném cílovém stroji.

Abel - Lokální instalace

- Nakopírovat soubory Abel.exe a Abel.dll do adresáře systému (např. C:).
- nainstalovat službu pomocí příkazu „abel“ v příkazové řádce operačního systému.

Abel - Vzdálená instalace

- V programu Cain v záložce Network vybereme cílový vzdálený počítač, kam chceme službu Abel nainstalovat.
- V levém stromu pravým tlačítkem myši klikneme na ikonu počítače a označíme „Connect As“.
- Do následně zobrazené tabulky zadáme uživatelské jméno a heslo administrátora pro daný vzdálený systém.
- Jakmile jsme připojeni, pravým kliknutím myši na ikonu „Services“ zobrazíme nabídku „Install Abel“ a kliknutím zahájíme instalaci.
- Oba soubory Abel.exe a Abel.dll budou automaticky nakopírovány do adresáře administrátora vzdáleného počítače. Služba se automaticky nainstaluje a spustí.

Modifikace Registrů

Program Cain v sobě zahrnuje nástroje pro vytváření změn v registrech systému. Vždy, když provádíme změny v registrech, je vhodné nejdříve dané registry zálohovat. Veškerá nastavení programu Cain jsou umístěny v HKEY_CURRENT_USER registru.

Odinstalování

Program Cain lze odinstalovat přímo pomocí odinstalačního programu nebo pomocí nástroje Přidat/Odebrat Programy v Ovládacích panelech operačního systému Windows. Odinstalační program neodstraní službu Abel.

Služba Abel se musí nejdříve zastavit a potom do příkazové řádky zadat příkaz „Abel -r“. Je možné k tomu použít i Service Managera programu Cain. Jakmile je služba odstraněna, vykonávací soubory mohou být ručně smazány ze systému [13].

4.2 Konfigurace

Před tím než můžeme nástroj Cain & Abel používat, musíme nejdříve nakonfigurovat některé parametry. Všechny parametry mohou být nastaveny z konfiguračního dialogu, do kterého se je možné dostat kliknutím v hlavní liště na „Configure“. Konfigurační dialog se skládá ze šesti záložek.

- Sniffer
- APR (Arp Poison Routing)
- Challenge Spoofing
- Filter and ports
- HTTP Fields
- Traceroute
- Certificate Spoofing

4.2.1 Záložka Sniffer

V této záložce se nastavuje, která síťová karta se použije pro nástroje programu sniffer a APR. Pokud máme více síťových karet (například LAN, Wi-Fi) vybereme jednu z nich. V této záložce je zobrazené číslo verze ovladače Winpcap, podle kterého zjistíme kterými vlastnostmi daný ovladač disponuje. Nutno podotknout, že tento program pracuje pouze s Ethernet síťovými adaptéry. V sekci Option této záložky je možno zaškrtnout tři položky.

- Start Sniffer on startup (zapnutí funkce Sniffer při zapnutí programu)
- Start APR on startup (zapnutí funkce APR při zapnutí programu)
- Don’t use Promiscuous mode (nepoužívat promiskuitní mód síťové karty)

Funkce prvních dvou položek není třeba blíže popisovat. Třetí položka slouží k deaktivování promiskuitního módu. Síťová karta v promiskuitním módu může přijímat i provoz, který není pro ni určený. Pokud se zatrhne volba „Don't use Promiscuous mode", povolí se APR Poisoning na bezdrátové síti, ale v této situaci není možné používat vlastnost MAC spoofing. [13]

4.2.2 Záložka APR

Uvnitř této záložky (viz Obr. 5a) se konfiguruje APR (Arp Poison Routing). Cain používá oddělené procesy, které posílají ARP Poison (tzv. otrávené) pakety napadanému hostovi, defaultně každých 30 sekund. To je nezbytné, protože položky přítomné v ARP cache vzdálených strojů, mohou být odstraněny v případě jejich žádného síťového provozu. Z tohoto dialogu je možné nastavit čas mezi každým vysláním ARP Poison paketů. Nastavením tohoto parametru na několik sekund, způsobí mnoho ARP síťového provozu, zatímco nastavení tohoto parametru na větší rozestup má za následek méně uneseného provozu. V sekci Spoofing Options se definují IP a MAC adresy, které Cain následně vepíše do ARP hlaviček ARP Poison paketů a přesměrovaných paketů. V případě definování falešné IP a MAC adresy ARP Poison útok bude plně anonymní, protože útočníkova reálna IP a MAC adresa nebude nikdy přes síť poslána. Aby se mohlo využít této volby, musí se brát v úvahu.

Falšování ethernetových adres může být použito, pouze když pracovní stanice útočníka je připojena k HUBu nebo switchem přepínané síti, která nepoužívá bezpečnostní vlastnost „Port Security". Port Security povolí připojit ke switchi pouze stanici, která má fyzickou adresu shodnou s adresou uloženou v seznamu. Tím chrání síť před připojením neznámé stanice do sítě. Switch zjistí, že se k jeho portu připojila jiná stanice než je povolená v seznamu, odpojí port a tím útočník ztrácí konektivitu do sítě.[13]

Falešná IP adresa musí být volná adresa v podsíti. ARP protokol neprojde přes routery nebo virtuální LAN sítě (VLAN), takže pokud se nastaví falešná IP, která je mimo danou podsíť, vzdálený host bude odpovídat na defaultní bránu a útočník neuvidí jeho odpovědi. Pokud bude použita falešná IP adresa, která už v podsíti existuje, nastane konflikt IP adres a útok bude jednoduše odhalen. Falešná IP adresa je programem automaticky zkontrolována, jakmile se zmáčkne tlačítko „Apply". Pokud bude IP adresa již použita, upozorní na tento problém varovná zpráva. Falešná MAC adresa nesmí být přítomná v podsíti. Přítomnost dvou identických MAC adres na stejné LAN, může způsobit konvergenční problém. Z tohoto důvodu si nelze jednoduše samostatně nastavit falešnou MAC adresu. Defaultní hodnota je nastavena na 001122334455, která je neočekávatelná v síti a na druhou stranu je lehce identifikovatelná při řešení problémů [13].

Není možné mít ve stejné podsíti dva a více počítače s programem Cain, používající APR falšování MAC adres a stejnou falešnou MAC adresu. Falešnou MAC adresu lze změnit modifikováním hodnoty registru „SpoofMAC" umístěnou v:

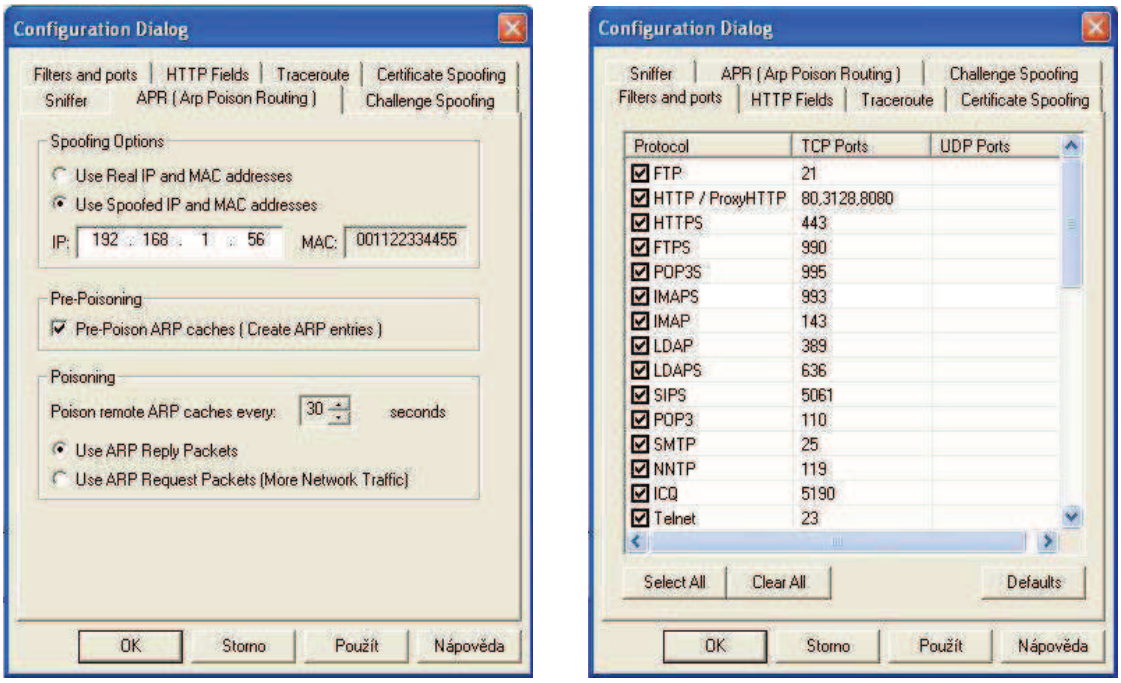
"HKEY_CURRENT_USER\Software\Cain\Settings"

4.2.3 Záložka Filters and Ports

V této záložce (viz Obr. 5b) je možné povolit/zakázat filtry snifferu a TCP/UDP porty aplikačních protokolů. Cain zachytává pouze autentizační informace a ne celý obsah každého paketu, nicméně je možné použít Telnet filtr, který ukládá všechna data přítomná v TCP spojení do souboru modifikováním portu filtru.

Sniffer filtry programu Cain jsou vnitřně navrženy k přežití v nespolehlivém světě jako je síť pod ARP Poison útokem. Cain používá odlišné techniky k extrahování všech informací ze síťových paketů, potřebných k odhalení vysílaných hesel ve formě čistého textu. Některé autentizační protokoly používají mechanismus výzva-odpověď, takže je potřeba posbírat parametry z provozu Client->Server i Server-> Client. Zachycení provozu v obou směrech je možné, když LAN síť je tvořena pouze HUBy nebo pokud je útočník připojen na port switche, kam se provoz zrcadlí, ale na přepínané síti toho může být dosaženo pouze užitím některé techniky únosu provozu jako je ARP Poison Routing (APR). Pokud se provádí sniffing s povoleným APR, sniffer bude extrahovat autentizace výzva-odpověď, pouze když se dosáhne Full-Routing stavu mezi napadenými počítači.

V této záložce je možné také povolit/zakázat analyzování směrovacích protokolů (RIPv1, RIPv2, EIGRP, HSRP,VRRP) a vlastnost ARP-DNS, která se jeví jako DNS Reply Rewriter [13].



a) b)

Obr. 5: Záložka: a) APR, b) Filters and ports

4.2.4 Záložka HTTP Fields

Tato záložka obsahuje seznam polí, do kterých se zapisují uživatelská jména a hesla, která používá HTTP sniffer filtr. Cookies a HTML formáty putující v HTTP paketech jsou snifferem prozkoumány, hledajíce pole s uživatelskými jmény a hesly. Pokud jsou tyto důvěrné informace nalezeny, jsou zachyceny a zobrazeny na obrazovce.

4.2.5 Záložka Traceroute

Zde je možné nastavit získávání uživatelských jmen hostů, objevování síťové masky pomocí protokolu ICMP a povolení/zakázání extrahování WHOIS informací pro každý skok.

4.2.6 Záložka Challenge Spoofing

Zde je možné nastavit hodnoty uživatelských výzev k přepsání NTLM autentizačních paketů. Tato vlastnost může být rychle umožněna z nástrojové lišty a musí se použít společně s APR. Pravidelné výzvy umožní prolomit NTLM hashe zachycené v síti pomocí Rainbow tabulek.

4.2.7 Záložka Certificate Spoofing

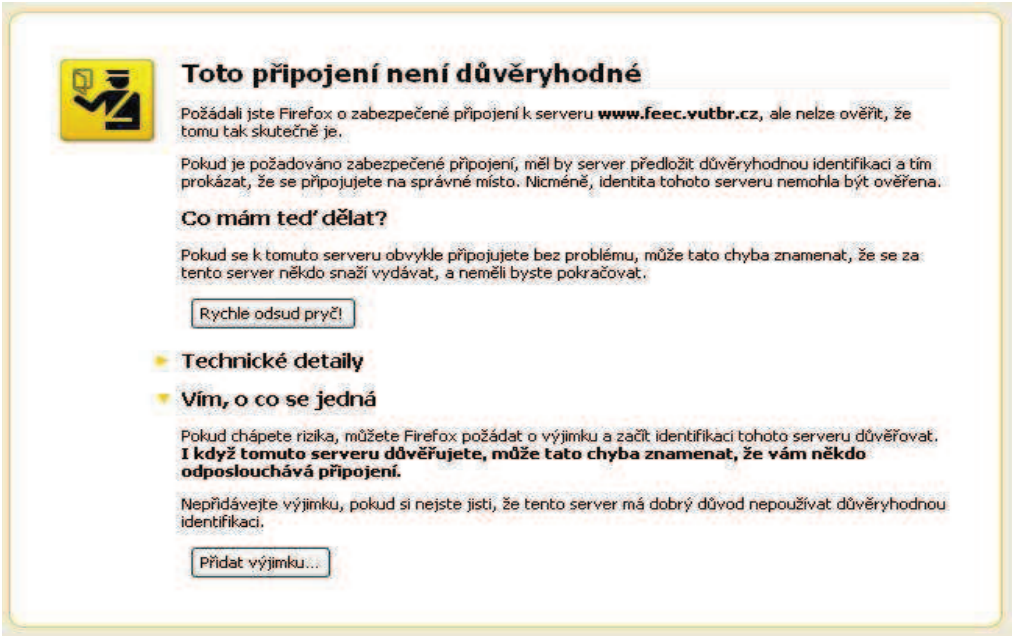
V této záložce se nastavuje nástroj programu Cain, nazývající se Certificates Collector. Jedná se o nástroj umožňující unést nebo vytvořit certifikáty. Certificates Collector uchopí certifikáty serveru z SSL povolených stránek a připraví je pro APR-*S filtr. Tato vlastnost je automaticky použita snifferem, ale je umožněno také manuálně vytvořit seznam předem vypočítaných fiktivních certifikátů. Program Cain v těchto fiktivních certifikátech nahradí asymetricky šifrované klíče novými klíči vygenerovanými programem. Tímto způsobem bude APR-*S filtr schopen šifrovat/dešifrovat SSL provoz. Útočník se strojem, na kterém má nainstalován program Cain, bude prostředník v komunikaci mezi dvěma komunikujícími hosty. Jedná se tedy o útok zvaný „muž uprostřed“ (Man in the middle).

V této záložce si je tedy možné vybrat, jestli se vytvoří a použije samostatně podepsaný falešný certifikát a nebo se vytvoří certifikátový řetězec, který je vytvořen pomocí certifikátů zachycených snifferem a uložených v podadresáři „Certs“ hlavního adresáře programu. Tento certifikát potom bude podsunut napadeným klientům během Man in the middle útoku.

Pokud se použije samostatně podepsaný falešný certifikát, prohlížeč napadeným klientům pravděpodobně zobrazí hlášku, že SSL-serverem povolený certifikát přichází z nedůvěryhodného zdroje (viz Obr. 6). Všechny ostatní parametry uvnitř tohoto certifikátu zůstávají stejné jako u toho skutečného, proto mnoho uživatelů nevěnuje tomuto varování pozornost a takový certifikát přijme. [13]

Falešné certifikáty jsou taktéž uloženy v podadresáři „Certs“ a seznam aktuálně dostupných APR-*S filtrů je udržován v souboru CERT.LST v hlavní adresáři programu. Tento soubor lze manuálně modifikovat k naučení filtru APR-*S, podstrčit vybraný certifikát do spojení napadených počítačů s daným SSL serverem.

Tato vlastnost je použita automaticky APR-*S sniffer filtry. K ručnímu zvolení a připravení falešného certifikátu, je možné použít tlačítko „modrý kříž“ v nástrojové liště. Pomocí syntaxe „hostname:port“ nebo „ip adresa:port“ je možné specifikovat nestandardní porty [13].



Obr. 6: Varování nedůvěryhodný certifikát

4.3 Decoders

Program Cain disponuje dekodéry hesel (Password Decoders), které umožní okamžitě dekódovat šifrovaná hesla z chráněné paměti Windows, LSA secrets, hesla pro připojení k bezdrátové síti WLAN, hesla internetového prohlížeče Internet Explorer, hesla z Windows Mail aplikací, hesla z Dialup aplikací, hesla ukrytá za hvězdičkami, dále hesla z databází a informace pro připojení k serverům a dalším vzdáleným síťovým zdrojům, které je možné si uložit do takzvaného Credential Manageru. Kompletní uživatelská jména, hesla a jiné přihlašovací informace je možné získat z lokálního počítače pouze pokud uživatel používá v programech a aplikacích možnosti zapamatovat přihlašovací údaje. Z toho vyplývá, že dané přihlašovací údaje, již minimálně jednou musely být na daném počítači zadány a musely být uloženy na jeho pevný disk. Jedinou výjimkou je nástroj Edit Boxes, který odkrývá hesla ještě než je uživatel poskytne pro ověření danému programu nebo aplikaci. [13]

Pro zobrazení přihlašovacích údajů z nástrojů v Decoders, je nutné kliknout na požadovaný nástroj a zmáčknout klávesu Insert nebo ikonu modrého kříže v nástrojové liště.

4.3.1 Protected Storage

Protected storage je úložný prostředek, který se převážně používá u operačních systémů MS Windows pro bezpečné uložení privátních klíčů, které byly uživateli vydány. Veškeré informace v tomto úložišti jsou šifrované pomocí klíče, který je odvozen od uživatelského hesla používaného k zalogování. Přístup k informacím je přísně regulovaný a je povolen pouze vlastníkovi. Tohoto úložiště pro citlivé informace využívá mnoho Windows aplikací, mezi které patří například Outlook, Outlook Express, Internet Explorer, MSN Explorer atd. Tyto aplikace zde ukládají uživatelská jména a hesla, například pokud chceme uložit uživatelské jméno a heslo do paměti, abychom ho nemuseli neustále vpisovat. Aplikace pak automaticky vyplní přihlašovací informace z informací uložených v Protected Storage. Citlivé informace tohoto úložiště se fyzicky ukládají do registru:

HKEY_CURRENT_USER\Software\Microsoft\Protected Storage System Provider\

Program Cain&Abel dokáže dekódovat informace uložené v Protected storage a zobrazit je v podobě čistého textu. Pokud uživatel používá Outlook Express pro stahování emailů ze seznamu.cz, tak pomocí tohoto nástroje se v programu zobrazí jako zdroj pop3.seznam.cz, což znamená, že se využívá pro stahování e-mailů protokol pop3. Ve sloupci Username se zobrazí e-mailová adresa uživatele, ve sloupci Password heslo k této adrese v podobě čistého textu, ve sloupci Type poznáme typ použité aplikace v tomto případě Outlook Express POP3 Account a v posledním sloupci Identity zjistíme, jestli se jedná o hlavní identitu. Informace z každé aplikace o jednotlivém účtu se rovná jednomu řádku v programu. Po kliknutí pravým tlačítkem na nějaký řádek, zobrazí se nabídka pro obnovení zachycených informací, odstranění informací v daném řádku, odstranit všechny informace zachycené tímto nástrojem a nebo zachycené informace exportovat do textového souboru. Pro zobrazení těchto informací je nutné zobrazit kartu Decoders→Protected Storage a zmáčknout klávesu Insert nebo ikonu modrý kříž v nástrojové liště. [13]

Tento nástroj programu ukazuje složitost nebo spíše jednoduchost s jakou se útočník může dostat ke zpřístupnění emailových účtů. Pokud se bude jednat o firemní emailové účty, může dojít k úniku velice citlivých firemních informací (například know-how firmy), což může danou firmu stát obrovskou cenu. Tento nástroj při mých pokusech přebíral funkci nástroje Windows Mail Passwords a všechny pokusy zachytit přihlašovací údaje z e-mailových aplikací byly zobrazeny v Protected Storage. Navíc se zde zobrazují i přihlašovací údaje zadaných v internetových prohlížečích Internet Explorer starších jako verze 7.

4.3.2 LSA Secrets

Pod položkou LSA Secrets se ukrývají uložené informace typu, hesla ke služebním účtům používaných pro zahájení služeb pod jiným účtem než se nachází na lokálním systému. To znamená služby jako vzdálená pomoc, vzdálená plocha a přihlašovací informace Dial-up aplikací. V LSA Secrets se pod názvem Default Password nachází i přístupové heslo účtu, na kterém je Cain&Abel nainstalován.

Tato vlastnost programu používá techniku „DLL injection.“ Jedná se o techniku „napíchnutí“ knihovny umožňující spustit podproces ve stejném bezpečnostním kontextu procesu Local Security Authority Subsystem (LSASS). Zdrojový kód vykonávající podproces musí být nejdříve nakopírován do adresového prostoru procesu LSASS. K tomu je zapotřebí účet s SeDebugPrivilege uživatelskými právy. To znamená mít administrátorská práva. Jakmile je podproces jednou vykonán, bude již vždy pracovat se stejnými přístupovými právy jaké má LSASS.

Tato vlastnost programu má za následek, že podproces načte funkci DumpLSA z knihovny Abel.dll, která otevře a položí dotaz na každé heslo používající LsarOpenSecret a LsarQuerySecret API ze souboru lsasrv.dll Tento soubor se nachází ve složce WINDOWS/system32. Podproces následně uloží data získaná pomocí této funkce do dočasného textového souboru lsa.txt nacházejícího se v hlavním adresáři programu Cain. Po zobrazení obsahu souboru na obrazovce se soubor smaže. Program nám dává na výběr zda chceme zobrazit LSA Secrets lokálního systému a nebo z externího registru. [13]

Pro získání LSA Secrets z podprocesu na vzdáleném PC je potřeba mít na daném stroji nainstalován program Abel.

4.3.3 Wireless Passwords

Wireless Zero Configuration Password Dumper je nástroj programu, který umožňuje odkrýt klíč k bezdrátovým sítím. Tento nástroj ukazuje nedokonalost zabezpečení bezdrátové sítě pomocí WEP využívající šifrovací algoritmus RC4. Šifrovací klíče zabezpečovací metody WEP jsou zobrazeny již v podobě čistého textu. U zabezpečení bezdrátové sítě pomocí WPA-PSK je možné zobrazit heslo pouze v hexadecimálním tvaru, protože používá SHA1 hashovací funkci. Pokud klikneme pravým tlačítkem na řádek s informacemi o dané bezdrátové síti, zobrazí se nabídka, kde můžeme daný hash odeslat do Crackeru programu Cain a pomocí něj aplikovat některý z útoků (Brute-force, slovníkový, pomocí Rainbow tabulek). Pokud správce bezdrátové sítě nastaví například devítimístné heslo nedávající smysl (nezjistitelné slovníkovým útokem), obsahující jen malou abecedu a číslice, útok hrubou silou může programu Cain trvat až 41000 let. [13]

Pomocí tohoto nástroje můžeme získat následující informace. Globální univerzální identifikátor (GUID) bezdrátové síťové karty, typ bezdrátové síťové karty, typ zabezpečovací techniky, identifikátor bezdrátové sítě SSID, přístupové heslo k síti a přístupové heslo v hexadecimálním tvaru.

Na základě tohoto nástroje může útočník získat neoprávněný přístup do bezdrátové sítě, kde může provádět následný sniffing síťového provozu a zachytávat hesla od různých síťových zdrojů, programů a aplikací. Správce bezdrátové sítě používající metodu WPA2-PSK, který má síť zabezpečenou heslem s osmi a více znaky, obsahující velkou, malou abecedu, číslice i speciální znak, by se neměl obávat prolomení hesla. Všeobecně se doporučuje v bezdrátových sítí, ve kterých se přenášejí i velmi citlivé informace, měnit přístupové heslo každé 4 měsíce. Překročení této doby by ale nemělo jinak výrazně ohrozit bezpečnost sítě.

4.3.4 IE7 Passwords

Program Cain&Abel umožňuje získat z paměti uložená hesla, která uživatelé používají k přihlášení na různé webové stránky. Tento nástroj odhalí pouze hesla, která uživatel uloží do paměti, to znamená, že kladně odpoví na otázku prohlížeče: „Chcete, aby si aplikace Internet Explorer pamatovala toto heslo?“ V této položce se zobrazují hesla Internet Exploreru verze 7 a novejší. Hesla u verzí starších se zobrazí v položce Protected Storage.

Získané údaje z tohoto nástroje jsou ve tvaru: URL webové stránky, uživatelské jméno a heslo.

Nevýhodou tohoto nástroje je neschopnost zjistit hesla z jiných internetových prohlížečů než je právě Internet Explorer. Tento nástroj s prohlížeči Mozilla Firefox, Opera ani Google Chrome nefunguje.

4.3.5 Windows Mail Passwords

Tento nástroj programu má zobrazovat přihlašovací informace k Microsoft Windows e-mailovým aplikacím. Problém ale je, že přihlašovací údaje od e-mailových aplikací Microsoft Outlook a Outlook Express se zobrazují v Protected Storage a e-mailový klient Hotmail zase v Credential Manageru, i když by je měl zobrazovat právě tento nástroj programu.

Domnívám se, že se jedná o drobnou chybu programu, kdy nástroj tyto informace správně odhalí, ale zobrazí je ve špatné položce karty Decoders.

4.3.6 Dialup Passwords

Pod touto položkou programu je možné zjistit přihlašovací informace o takzvaných Dialup službách. To znamená o službách, které před přenosem dat nejdříve vytvářejí spojení. Vyskytují se zde uživatelská jména a hesla použitá při komunikaci pomocí virtuálních privátních sítí (VPN), modemu a podobně. Program tyto informace hledá v registrech L$_RasDefaultCredentials a RasDialParams!<UserSID>. Informace, které je možné v této položce nalézt jsou: [13]

název firemní sítě nebo serveru, ke kterému bylo připojení uskutečněno, IP adresa serveru nebo jeho doménový název, uživatelské jméno a heslo, název domény, druh zařízení a druh služby (VPN, PPPoE, Modem).

Všechny tyto informace je možné exportovat do textového souboru. Tyto informace je možné získat i ze vzdáleného PC, který má nainstalován na svém disku program Abel. Tyto informace je možné najít v položce LSA Secrets pod zobrazeným obsahem výše jmenovaných registrů.

Díky získaným informacím umístěných v této položce programu, může útočník získat přístup do cizí sítě pomocí VPN, modemu nebo jiných služeb, které před přenos dat nejdříve sestavují spojení. Útočník se pak následně může dostat k dalším síťovým zdrojům a z nich získat další citlivé informace. Útočník se pomocí těchto informací jednoduše dostane přes zabezpečení sítě například na vnějším routeru firemní sítě.

4.3.7 Edit Boxes

Pod touto položkou se nacházejí informace, které zpracoval nástroj Box Revealer. Box Revealer umožňuje odkrýt hesla, která se skrývají v přihlašovacích text-boxech za hvězdičkami. Tento nástroj podporuje většinu standardních text-boxů, do kterých se vkládají hesla. Nástroj umožňuje zjistit heslo ukrývající se za hvězdičkami u aplikací jako je ICQ, Skype, softwarový VoIP telefon X-lite, ale i v Total Commanderu při přihlašování k FTP serveru, VNC vieweru nebo při přihlašování ke vzdálené ploše. Hlavně takto snadno získané heslo z přihlašovacího dialogu ke vzdálené ploše, umožní útočníkovi získat kontrolu nad daným vzdáleným PC a využít ho k dalším útokům. [13]

Tento nástroj je velice nebezpečný na počítačích, který používá více uživatelů pod stejným účtem. Jakmile uživatel zadá u nějaké aplikace uložit heslo, útočník si může, při pozdějším zapnutím počítače, k uživatelskému jménu, které je normálně vidět, zobrazit i heslo a přistupovat k daným aplikacím i z jiných fyzických počítačů, na kterých se například nekontrolují logy. Díky tomu útočník může lépe skrýt svoji identitu a má větší šanci, že nebude odhalen on ani jeho nekalá práce.

4.3.8 Enterprise Manager

Enterprise Manager je aplikace používaná u operačních systémů Windows k řízení a správě MS SQL Serverů verze 7.0 a novějších. Jakmile se nakonfiguruje používání autentizace SQL Serveru, veškeré přihlašovací informace se uloží pomocí Enterprise Manageru do registru.

Přihlašovací informace k SQL Serveru se šifrují nebo dešifrují pomocí funkce XOR. SQL Server pomocí funkce CryptProtectData následně chrání XOR zašifrované informace před jejich zapsáním do registru. Z toho vyplývá, že citlivé přihlašovací údaje mohou být dešifrovány pouze z uživatelského účtu a z fyzického stroje, z kterého byly tyto přihlašovací informace původně vytvořeny. Jestli je zapotřebí dekódovat přihlašovací údaje, je nutné použít CryptUnprotectData API z knihovny crypt32.dll a vykoná se dešifrování pomocí operace XOR. [13]

V Enterprise Manageru v programu Cain z dešifrovaných informací získáme informace o názvu SQL Serveru, skupině ve které se nachází, verzi SQL Serveru, dále přihlašovací uživatelské jméno, heslo a číslo profilu, ze kterého byly přihlašovací údaje vytvořeny.

Získání přihlašovacích údajů k SQL Serveru má pro útočníka využívající program Cain&Abel velkou překážku, kterou je podmínka dešifrovat zašifrované přihlašovací informace z registru ze stejného uživatelského účtu a fyzického stroje.

4.3.9 Credential Manager

Credential Manager je nástroj, který vyvinula společnost Microsoft a nasadila tento nástroj od verzí operačních systémů Windows Server 2003 a Windows XP. Tento nástroj poskytuje bezpečné uložení přihlašovacích informací. Umožňuje uložit uživatelská jména a hesla pro různá síťová zařízení a aplikace do úložiště, z kterého následně při příští návštěvě daného zařízení nebo aplikace tyto informace systém doplní automaticky.

Uložit informace do Credential Manageru lze ve Windows XP pomocí Správce síťových hesel, který lze nalézt pod:

Start → Nastavení → Ovládací Panely → Uživatelské účty → (Název účtu) → Správa síťových hesel.

Pod položkou Credential Manager v programu Cain získáme informace jako jsou název serveru, uživatelské jméno, heslo, typ hesla (např. doménové), alias, komentář, dále jestli se jedná o lokální nebo enterprise heslo a poslední doba použití daných informací. [13]

Útočník, který se dostane k informacím z Credential Manageru, může získat přístup k serverům, uživatelským stanicím nebo aplikacím v dané síti, z kterých může podnikat následné útoky na zbylé síťové zdroje nebo může z těchto zařízení odcizit informace, které jsou pro majitele důležité. V tomto případě se jedná o velké bezpečností riziko, protože v tomto případě se většinou jedná o administrátorská hesla, která útočníkovi poskytnou neomezenou nadvládu nad daným zařízením.

4.4 Network

Pod záložkou Network se nachází nástroj Network Enumerator. Network Enumerator je nástroj, který umožní objevit v lokální síti LAN přítomnost jiných zařízení. Umožňuje rychle a jednoduše identifikovat ostatní počítače v dané LAN, Apple souborové servery, Dial-In servery, doménové řadiče, Novell servery, SQL servery, tiskové servery, terminálové servery a servery pro synchronizaci času (Time servery). Tento nástroj umí zobrazit i verze operačních systémů.

V levé části záložky se nachází strom používaný k prohlížení sítě a pro připojení ke vzdáleným strojům. Jakmile se připojíme k serveru nebo počítači, můžeme zobrazit uživatelská jména, skupiny, služby, a sdílené složky přítomné na daném zařízení. Quick list se používá ke vložení IP adresy hosta, který není při prohlížení sítě vidět.

Při výčtu uživatelů, Cain také zobrazí jejich SID (Security Identifier) a dokáže identifikovat jméno administrátorského účtu, i když byl přejmenován. SID administrátorského účtu končí vždy hodnotou 500. [13]

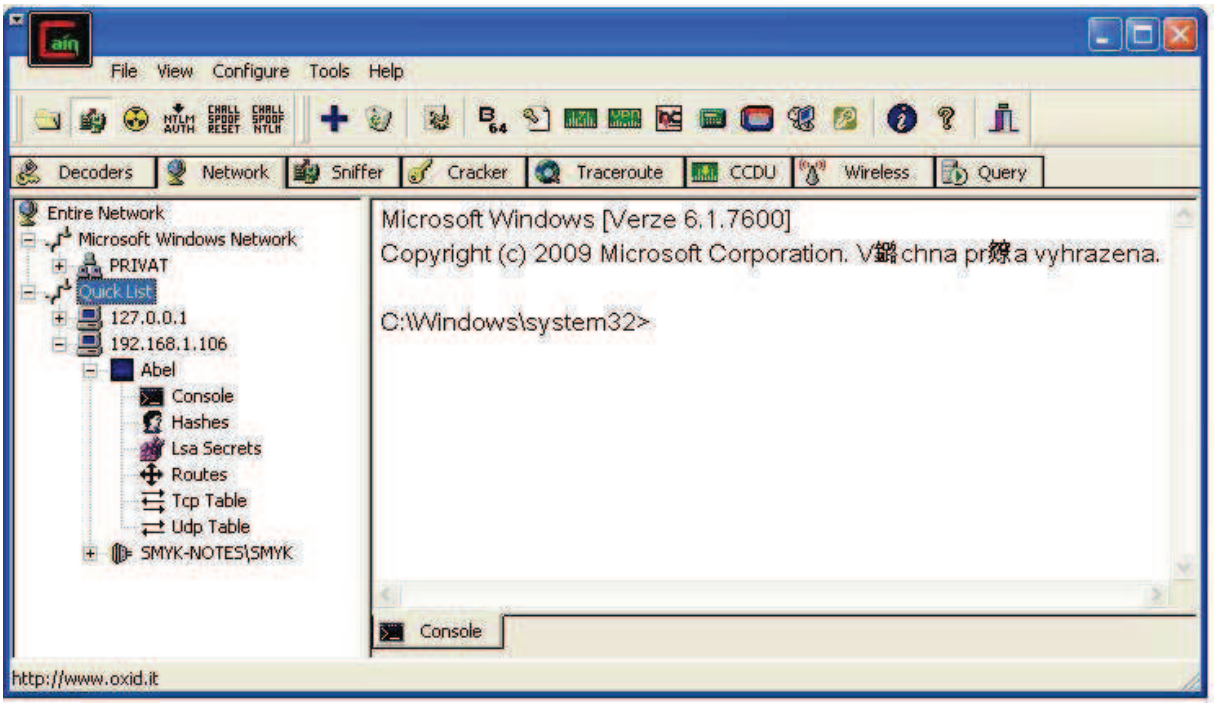
Windows NT a novější verze operačních systémů mají vlastnost, která může omezit schopnost anonymně přihlášeným uživatelům zobrazit uživatelská jména a názvy sdílených složek. Toho je dosaženo nastavením parametru registru

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\LSA\restrictanonymous

na hodnotu 1. Pokud program kvůli tomuto omezení nemůže provést výčet uživatelů, zapne automaticky SID skener a bude dále postupovat podle principu funkce programu „sid2user". Pro vytvoření síťového spojení na cílového hosta, se klikne pravým tlačítkem na některého z výčtu hostů, zvolí se „Connect As" a zadají se přihlašovací údaje. Anonymní spojení na cílového hosta se provádí tak, že v dialogu pro vložení přihlašovacích údajů se nechají daná pole prázdná. Po přihlášení k cílovému hostu se nám v Quick listu zobrazí jeho IP adresa a pod ní se nachází 5 položek. Groups, Registry, Services, Shares a Users. Pod položkou „Groups" se nachází informace o členství hosta v různých skupinách, pod položkou „Registry" se nacházejí registry daného hosta, které je možné následně editovat. Položka „Services" ukrývá informace o službách na cílovém hostu. Po zobrazení dané položky se objeví výčet služeb, jejich názvy, stavy (jestli běží či nikoliv), jestli běží automaticky nebo se spouští uživatelem a umístění spouštěcího souboru služby. Po kliknutím pravým tlačítkem na tuto položku se zobrazí nabídka pro nainstalování programu Abel do cílového hosta. Položka „Shares" zobrazí sdílené složky a „Users" zjistí jaké existují na daném stroji lokální uživatelské účty. [13]

4.4.1 Abel

Pomocí programu Abel můžeme cílového hosta ovládat pomocí vzdálené příkazové řádky (položka Console), zjistit hashe hesel k uživatelským účtům, z kterých je možné v nástroji LM & NTLM Cracker v programu Cain daná hesla odkrýt do podoby čistého textu. Dále je možné pomocí programu Abel nahlédnout do některých registrů a vyčíst z nich některé nešifrované přihlašovací údaje. Abel také umí zobrazit lokální směrovací tabulku daného vzdáleného hosta a zobrazit tabulku otevřených TCP a UDP spojení. Lokální směrovací tabulka (Routes) a tabulka otevřených TCP (TCP Table) a UDP (UDP Table) spojení mi připadají zbytečné, protože vzdálená příkazová řádka všechny tyto funkce v sobě zahrnuje. Lokální směrovací tabulka vzdáleného hosta se ve vzdálené příkazové řádce zobrazí pomocí příkazu „route print" a tabulky TCP a UDP spojení se zobrazí pomocí příkazů „netstat –a". Jediné co trošku stojí za zmínku, je jednoduché přidávání a odebírání směrovacích pravidel. Zde se nabízí možnost pro útočníka vnutit vzdálenému počítači svoji IP adresu jako default gateway, čímž má útočník zajištěno, že veškerý odchozí provoz do internetu půjde přes něj. Toho je ovšem možno dosáhnout i pomocí vzdálené příkazové řádky. Naopak nástroj vzdálená příkazová řádka je pro útočníka vítězství, protože díky ní cílového hosta prakticky ovládá.



Obr. 7: Ovládnutí příkazového řádku vzdáleného PC pomocí programu Abel

4.5 Sniffer

Hlavním nástrojem programu Cain & Abel je nástroj sniffer. Tento nástroj umožňuje zachytávat hesla a autentizační informace, které putují napříč sítí. Sniffer byl vyvinut pro práci na přepínaných sítích pomocí APR (ARP Poison Routing). Sniffer disponuje protokolovými filtry a filtry pro hesla, které mohou být povoleny/zakázány z hlavního konfiguračního dialogu.

Hlavní protokolový filtr je filtr BPF (Berkeley Packet Filter), který vykonává počáteční třídění provozu. Filtr dává pokyny protokolovému ovladači, aby zpracovával pouze ARP a IP provoz. Ostatní protokoly se nebudou zpracovávat. Sniffer se aktivuje a deaktivuje pomocí ikony v nástrojové liště a jeho parametry mohou být konfigurovány v hlavním konfiguračním dialogu (Configure). [13]

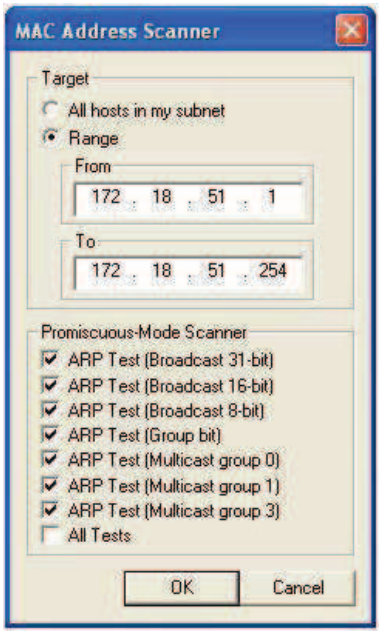
Záložka Sniffer se dělí na pět částí, kdy každá část vykonává odlišnou funkci. Jsou to části Hosts, APR, Routing, Passwords a VoIP.

4.5.1 Hosts

V první části snifferu zvané Hosts se nachází nástroj MAC Scanner (viz Obr. 8). Jedná se o nástroj, který dokáže ve velmi krátké době objevit ostatní síťová zařízení v dané podsíti. Pro objevení ostatních síťových zařízení v podsíti využívá ARP rámců, které pracují na principu žádost/odpověď. Aby bylo možné MAC Scanner používat, musí být sniffer aktivován.

Pro objevení ostatních síťových zařízení v podsíti je třeba do MAC Scanneru vložit rozsah IP adres (podsíť, ve které se nachází počítač s programem Cain) a zvolit některý nebo všechny ARP testy. Nástroj následně prozkoumá danou podsít a zobrazí do přehledné tabulky IP adresu daného zařízení, jeho MAC adresu, výrobce síťové karty (v případě integrované síťové karty výrobce základní desky) a na který test MAC skeneru zařízení odpovědělo. Nástroj umožňuje zjistit i název daného zařízení (Host name). Pro zjištění názvu zařízení je potřeba kliknout pravým tlačítkem na řádek s informacemi o daném zařízení a zvolit „Resolve Host name“. [13]

Díky tomuto nástroji může útočník jednoduše objevit ostatní zařízení v síti, zjistit si, které IP adresy v rozsahu jsou přiděleny a které jsou volné, zjistí si fyzické adresy jednotlivých zařízení, podle výrobce síťové karty nebo základní desky může zjistit, zda se jedná o PC nebo router, switch, firewall a podobně. Jelikož tento nástroj využívá ARP rámců, je možné odhalit síťová zařízení nacházející se pouze v broadcastové doméně.



Obr. 8: Skener MAC adres

4.5.2 APR

Sniffer se snaží využít různých stavů protokolů, aby ze síťových paketů vytáhnul všechny informace potřebné k odkrytí posílaného hesla v podobě čistého textu. Některé autentizační protokoly používají mechanismy výzva-odpověď, z tohoto důvodu sniffer potřebuje získat parametry z obou směrů provozu (Client ⇔ Server). Toho je možné dosáhnout zrcadlením portu nebo když APR dosáhne stavu, kdy je IP provoz mezi dvěma hosty kompletně unesen a APR pracuje duplexně (Full-routing). Když je APR povoleno, sniffer musí zpracovávat pakety, které nejsou normálně viditelné a také je musí přesměrovat

do správných destinací. To může způsobit snížení výkonu na sítích s velkým provozem. Hlavní výhodou APR je, že umožňuje sniffing na přepínaných sítích a také umožňuje analyzovat šifrované protokoly jako jsou HTTPS a SSH-1.

Všechna zachycená hesla a hashe se ukládají do souborů s koncovkou **.LST** v hlavním adresáři programu. Tyto soubory je možné prohlížet nebo importovat do různých textových editorů. U protokolů HTTPS, SSH a Telnet jsou celé relace dešifrovány a zachyceny do textových souborů pomocí této konvence. [13]

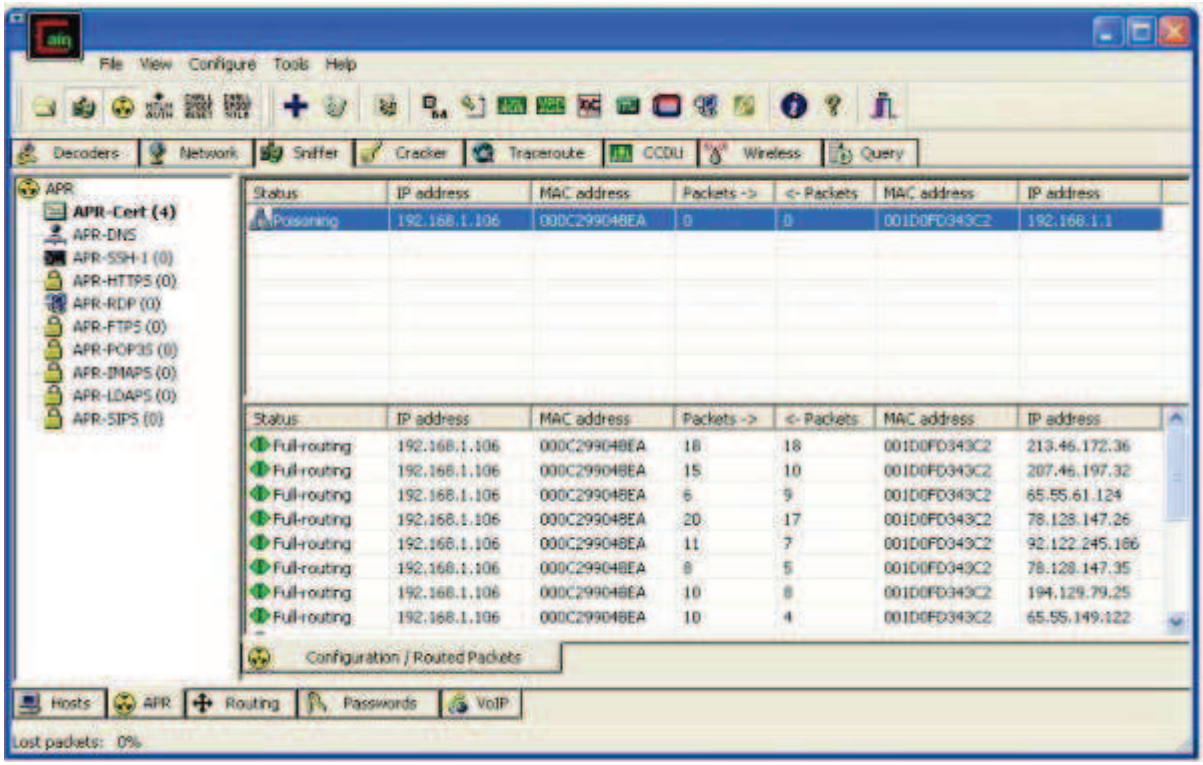
<název protokolu>-<rok> <měsíc> <den> <hodina> <minuta> <sekunda> <milisekunda> -<port>.txt

Příklad: **HTTPS-20091021173870-4202.txt**

Sniffer dokáže také zpracovávat soubory i v off-line módu. Zachycené informace uložené v souborech mohou být importovány do programu pomocí ikony „Open“ v nástrojové liště. [13]

Útok ARP Spoofing

Program Cain & Abel dokáže z útoků popsaných v kapitole 3 pouze ARP Spoofing. Tento útok je podrobně popsán v kapitole 3.4.1. V programu Cain & Abel se tento útok provádí tak, že se nejdříve přepneme do záložky „Sniffer“ a zvolíme kartu APR. Následně musíme vybrat dvě síťová zařízení v síti LAN, mezi kterými budeme komunikaci zachytávat. Tyto zařízení vybereme z nabídky, která se zobrazí po zmáčknutí klávesy „Insert“ nebo ikony v modrého kříže v nástrojové liště. Pro tento útok je potřeba, aby sniffer znal MAC adresy zařízení, mezi kterými bude sledovat komunikaci. Proto je potřeba tyto adresy zjistit pomocí MAC skeneru. Pokud máme zjištěné MAC adresy vybraných zařízení můžeme naši volbu potvrdit. V horní tabulce (viz Obr. 9) se zobrazí parametry sledovaného spojení. Jako poslední je potřeba útok aktivovat pomocí žluté kulaté ikony v nástrojové liště. Po aktivování útoku se v dolní tabulce zobrazí stav sledovaného spojení, jestli sleduje komunikaci v jednom nebo v obou směrech a počty přenesených paketů. V levé části se potom zobrazují zachycené certifikáty, které se nahradí za falešné, aby komunikace mohla probíhat přes náš sniffer. Podobně se zde zobrazují i pakety odpovědí serverů DNS (DNS Reply), e-mailových, FTP, SIP nebo při práci se vzdálenou plochou. Při obdržení falešného certifikátu se napadenému uživateli zobrazí hláška, že certifikát pochází z nedůvěryhodného zdroje. Tuto informaci v drtivé většině běžní uživatelé ignorují, takže útočník může prohlížet i šifrovanou komunikaci.



Obr. 9: Zachytávání komunikace pomocí ARP Spoofing

4.5.3 Routing

Sniffer umí analyzovat i směrovací protokoly. Cain & Abel nepodporuje všechny směrovací protokoly, podporuje pouze HSRP, EIGRP, OSPF, RIPv1, RIPv2 a VRRP. Tato vlastnost umožňuje rychle identifikovat směrování v podsíti. U protokolů EIGRP a RIP lze zachytit aktuální směrovací tabulky, které jsou sdíleny mezi směrovači. Tato vlastnost je umožněna pouze v případě, že tyto protokoly nevyžadují autentizaci. [13]

4.5.4 Passwords

Při sledování komunikace v lokální síti, sniffer hledá v přenášených rámcích hesla a přihlašovací údaje nebo jejich otisky (hashe) z různých protokolů. Pokud sniffer zachytí heslo z protokolu, který nepoužívá šifrování (Telnet, SIP, ICQ, HTTP, POP3, ...), okamžitě zobrazí heslo společně s parametry přenosu (IP adresy, čas zachycení, uživatelské jméno) v kartě Passwords pod příslušným protokolem. Heslo je zobrazeno v podobě čistého textu. Nevýhoda nešifrovaných protokolů je jasná. Můžeme mít sebesilnější heslo, ale pokud komunikaci někdo odchytává, útočník ho zjistí za pár sekund. U protokolů, které jsou šifrované, neposílají hesla v podobě čistého textu, ale posílají hash daného hesla, sniffer zachytí tento hash a zobrazí jej opět v kartě Passwords pod příslušným protokolem. Aby útočník zjistil z daného hashe heslo, musí na daný hash kliknout pravým tlačítkem a zvolit „Send to Cracker“. Hash bude odeslán do Crackeru programu Cain, kde se na něj budou aplikovat útoky (slovníkový, hrubou silou, ...).

Program Cain & Abel umí zachytit hesla z těchto protokolů:

FTP, HTTP / HTTP Proxy, IMAP, POP3, SMTP, LDAP, NNTP, ICQ, VNC, TDS, MySQL, DCE/RPC, SMB, MS Kerberos5, Radius, IKE, SNMP, RIP, HSRP, EIGRP, OSPF, VRRP, SIP, GRE/PPP, PPPoE, Oracle TNS, Telnet, HTTPS, IMAPS, POP3S, LDAPS, FTPS, MGCP/RTP, SSH-1, SIP/RTP [13].

4.5.5 VoIP

Tento nástroj umí zachytávat komunikaci uskutečněnou přes VoIP protokol, nahrát ji a uložit ji na pevný disk. Pokud sniffer objeví hlasová data na síti, zachytí je v obou směrech (volající ⇔ volaný) a uloží je jako mono nebo stereo WAV soubor. VoIP protokoly jako SIP nebo H.323 používají pro přenos multimediální dat přenosový protokol RTP. Sniffer si vytáhne například z protokolu SIP parametry RTP spojení jako RTP porty, IP adresy volajícího a volaného a typy kodeků. Následně zachytává a dekóduje audio data z RTP. [13]

Pokud je sniffer aktivní, zachytává VoIP komunikaci automaticky. Útočník díky této funkci dostává nástroj klasického odposlechu, kdy mu neunikne žádný hovor v lokální síti. Zachycený hovor je zobrazen v řádku tabulky a poskytuje útočníkovi i dodatečné informace jako čas zahájení a konce hovoru a z jakých IP adres a portů byl hovor uskutečněn. Po kliknutí pravým tlačítkem na daný řádek tabulky je možné si okamžitě jednotlivé hovory poslechnout.

Podporované kodeky:

G.711 uLaw, G.771 aLaw, ADPCM, DVI4, LPC, GSM610, Microsoft GSM, L16, G729, Speex, Speex-16Khz, Speex-32Khz, iLBC, G722, G722.1, G723.1, G726-16, G726-24, G726-32, G726-40, LPC-10, SIREN, LRWB-16khz, AMR-NB, AMR-WB [13].

4.6 Cracker

Cain & Abel má v sobě zabudovaný nástroj na „lámaní“ hesel (Password Cracker), který podporuje většinu běžných hashovacích algoritmů a několik na nich založených šifrovacích metod. Tento nástroj se v programu nachází v záložce „Cracker“ na hlavním panelu. Na levé straně se nachází seznam typů hashů a šifrovacích algoritmů, které je možné prolomit.

Podporované hashe:

MD2, MD4, MD5, SHA1, SHA2 (256 bit), SHA2 (384 bit), SHA2 (512 bit), RIPEMD160.

Podporované šifrovací algoritmy:

PWL soubory, Cisco-IOS Type-5 enable passwords, Cisco PIX enable passwords, APOP-MD5, CRAM-MD5, LM, LM + Challenge, NTLM, NTLM + Challenge, NTLM Session Security, NTLMv2, RIPv2-MD5, OSPF-MD5, VRRP-HMAC-96, VNC-3DES, MS-Kerberos5 Pre-Auth, RADIUS Shared Secrets, IKE Pre-Shared Keys, Microsoft SQL Server 2000, Microsoft SQL Server 2005, Oracle, Oracle-TNS-DES, Oracle-TNS-3DES, Oracle-TNS-AES128, Oracle-TNS-AES192, MySQL323, MySQLSHA1, SIP-MD5, WPA-PSK, WPA-PSK-AUTH, CHAP-MD5, MS-CHAPv1, MS-CHAPv2. [13]

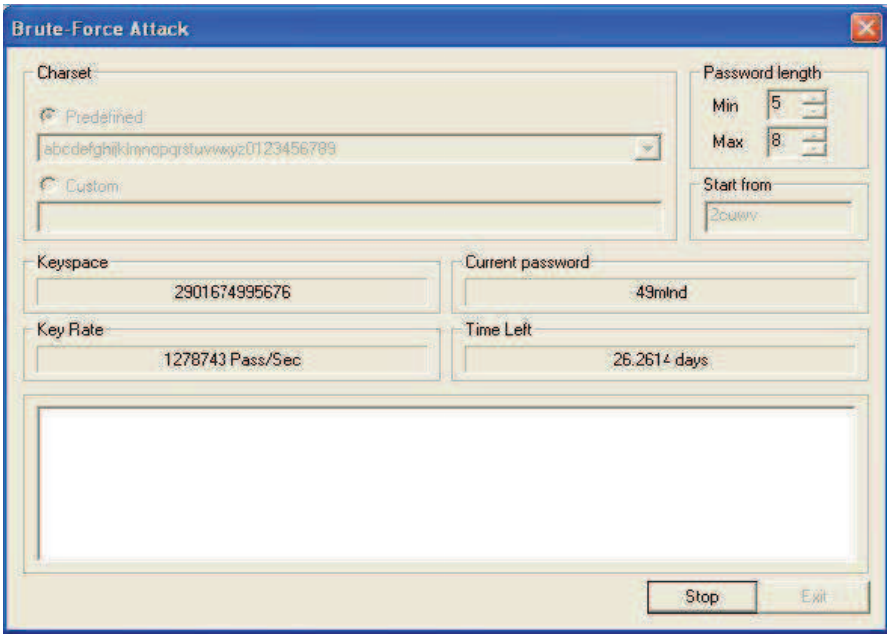
Všechna zachycená hesla se posílají sem do Password Cracker a automaticky se zařadí pod správný typ hashe nebo šifrovacího algoritmu. Kliknutí do seznamu na určitý typ hashe nebo šifrovacího algoritmu zobrazí v hlavním poli všechna zachycená hesla nebo hashe daného typu k prolomení. Zobrazená hesla a hashe je možné označit a následně na ně aplikovat některý z heslových útoků.

- Hrubou silou (Brute-Force Attack)
- Slovníkový (Dictionary Attack)
- Kryptoanalýzu (Cryptanalysis Attack)

4.6.1 Útok hrubou silou

Útok hrubou silou (Brute-Force attack) je metoda „lámání“ šifer, neboli dešifrování specificky šifrovaného textu, zkoušením všech možných klíčů. To znamená, že při tomto útoku, pokud se nedefinují omezení použití pouze určitých znaků, program postupně vyzkouší všechny možné kombinace znaků. Je důležité podotknout, že tento útok vždy danou šifru dešifruje. Vhodnost použití útoku hrubou silou závisí na délce šifry a velikosti výpočetního výkonu útočníkova stroje.

Dialogové okno útoku hrubou silou (viz Obr. 10) umožňuje definovat různé parametry. V sekci „Charset“ lze vybrat znakovou sadu, z jejichž znaků se budou vytvářet kombinace klíčů. Je zde na výběr z předdefinovaných sad znaků, od pouze malých/velkých znaků abecedy, pouze číslic, přes malé znaky abecedy + speciální znaky, až po malou + velkou abecedu + číslice + speciální znaky. Nebo je možné si zde navolit znaky samostatně. Z Obr. 10 je možné vyčíst, že je nastavena znaková sada malá abeceda + číslice.



Obr. 10: Dialogové okno pro útok hrubou silou

V sekci „Password length“ se nastavuje minimální a maximální délka klíče, která se bude testovat pro prolomení klíče. V tomto případě je nastaveno testování kombinací 5 až 8 znaků. V sekci „Keyspace“ je zobrazen počet možných kombinací klíčů. V sekci „Current password“ je zobrazena aktuálně testovaná kombinace, v sekci „Key Rate“ je zobrazen počet otestovaných kombinací klíčů za sekundu a v sekci „Time Left“ je zobrazena doba, která zbývá do otestování všech kombinací klíčů. V tomto případě zbývá něco málo přes 26 dní.

Obranou proti útoku hrubou silou je vytvoření silného hesla, které se bude po určité době měnit. Silné heslo by mělo obsahovat minimálně 8 znaků, malá a velká písmena, číslice a speciální znaky. Silné heslo by se mělo pravidelně měnit zhruba po čtyřech měsících.

Příklad silného hesla: aKw9Ip:0bs

Tento nástroj prolomí jakékoliv heslo, záleží jen, jak dlouho mu to bude trvat. Pokud máme silné heslo o osmi znacích, malá i velká abeceda plus číslice a speciální znak, bude tomuto nástroji, nainstalovaném na běžném počítači, vyzkoušení všech kombinací trvat asi 56 let.

4.6.2 Slovníkový útok

Slovníkový útok pracuje na principu využití základní chyby uživatele - zvolení slabého hesla. Uživatel si často volí heslo, které se mu dobře pamatuje. Proto si volí hesla typu jméno manželky, datum narození, jméno domácího mazlíčka atd. Při tomto útoků se zkouší každé slovo, které je obsaženo ve slovníku, jestli se jedná o dané heslo. Tento útok je účinnější než útok hrubou silou, protože uživatelé si slabá hesla volí velice často.

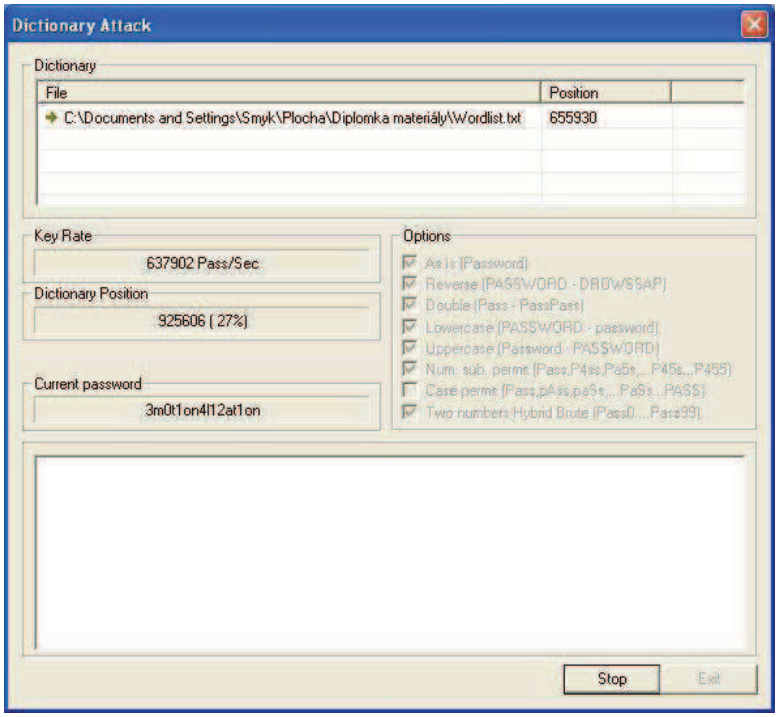
Existují dvě metody pro zvýšení úspěchu slovníkového útoku. První metodou je užití rozsáhlého slovníku nebo užití více slovníků. Zde má útočník výhodu, pokud daného uživatele zná osobně, protože může použít více specifické slovníky. To znamená, že pokud daný uživatel zajímá fotbal, útočník může použít specifický slovník s názvy fotbalových klubů, se jmény hráčů apod. Druhou metodou je manipulace s řetězcem znaků ve slovníku. Každé slovo ve slovníku je možné různě modifikovat, například slovo ve slovníku otestuje pozpátku, převede na velká písmena a podobně.

Dialogové okno slovníkové útoku (viz Obr. 11) umožňuje v sekci „Dictionary“ vložit nebo odstranit dané slovníky, začít hledat ve slovníku od určitého pozice a následně resetovat pozici od které se bude začínat útok provádět. V sekci „Options“ se nastavují operace, které se budou provádět se slovem ve slovníku, při zkoušení najít dané heslo. Je zda na výběr z možností:

- As Is – zkusí se slovo ve tvaru, jak je napsáno ve slovníku
- Reverse – slovo se zkusí pozpátku
- Double – slovo se zdvojí (Heslo -> HesloHeslo)
- Lowercase – slovo se převede na malá písmena
- Uppercase – slovo se převede na velká písmena
- Num. sub. perms – místo každého znaku slova se zkusí číslice (Heslo->H2slo, H2sl8)
- Case Perm - místo každého znaku slova se zkusí velké písmeno (Heslo->HEslo,HesLO)
- Two numbers Hybrid-Brute - na každé slovo se vloží číslice 0 až 99

V sekci „Key Rate“, podobně jako u útoku hrubou silou, ukazuje kolik se při útoku vyzkouší slov ve slovníku za jednu sekundu. „Dictionary Position“ zobrazuje pořadové číslo právě zkoušeného slova ve slovníku a zároveň kolik procent slov ze slovníku již bylo pro prolomení hesla vyzkoušeno. „Current password“ zobrazuje aktuálně testované slovo ve slovníku.

Tento nástroj je velice efektivní při snaze zjistit heslo běžných uživatelů. Pokoušet se prolomit heslo slovníkovým útokem (například administrátorského heslo serveru) nemá moc šancí na úspěch. Tento nástroj rychle objeví hesla, která mají určitý význam nebo něco představují nebo jsou to hesla primitivní, která napadnou každého (12345 a podobně). Při tomto útoků rychlost odhalení hesla záleží jen na velikosti slovníku, který útočník použije. Běžná hesla pomocí tohoto útoku lze odhalit během několika hodin nebo dní.



Obr. 11: Dialogové okno pro slovníkový útok

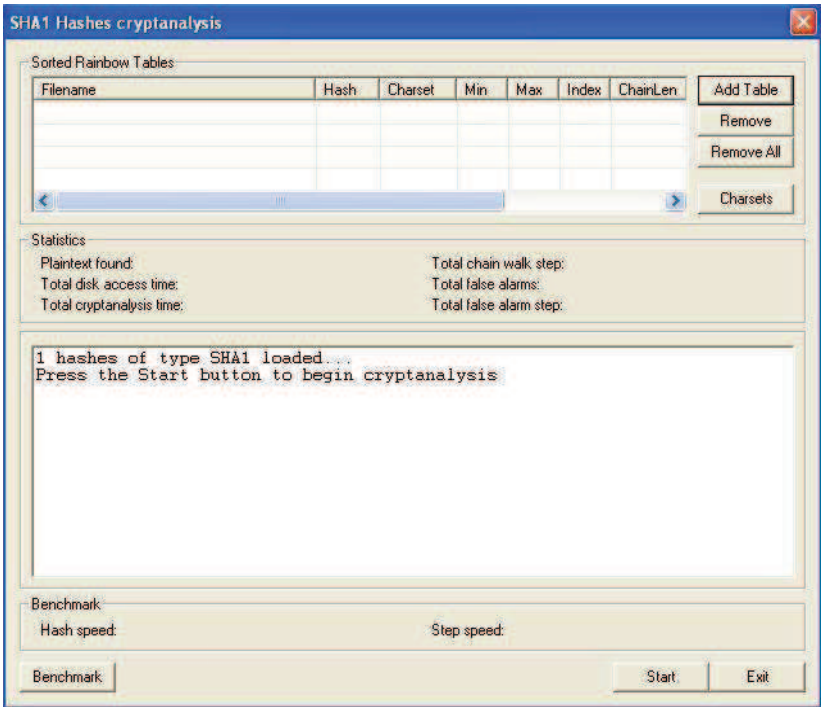
4.6.3 Kryptoanalýza

Tato vlastnost umožňuje zlomit heslo užitím metody Faster Cryptanalyctic time-memory trade off. Tato technika využívá sady velkých tabulek před-vypočítaných šifrovaných hesel, zvané Rainbow tabulky, ke zlepšení dnes známých porovnávacích metod a ke zrychlení odhalení hesla ve formě čistého textu.

Rainbow tabulky mohou být generovány například pomocí programu "RainbowCrack" nebo pomocí Windows utility „winrtgen", která je dostupná na webových stránkách výrobce programu Cain & Abel. [13]

Tato technika je velice rychlá, nicméně je použitelná pouze k lámání hesel určitého typu. Tato technika se nehodí u autentizačních protokolů typu výzva-odpověď a při použití techniky takzvaného solení, kdy se ještě před hashováním k heslu připojí náhodný řetězec dat. U těchto případů by se musela při každé výzvě nebo soli generovat nová Rainbow tabulka, což je velice nepraktické. Tento útok není vhodný na hashe hesla zachycených ze sítě. Naopak tento útok je velice účinný u přímých hashů, které se používají při lokálním ukládáním šifrovaných hesel.

Dialogové okno tohoto útoku (viz Obr. 12) umožňuje v sekci „Sorted Rainbow Tables" přidat/odebrat Rainbow tabulku a přidat sadu znaků, která se použije. Je možné použít předefinované sady ze souboru „charsets.txt" umístěném v hlavním adresáři programu. V sekci „Statistics" se zobrazují statistiky tohoto útoku. Například doba za jakou bylo heslo objeveno a podobně. V hlavní sekci je pak zobrazen podrobný výpis útoku a včetně objeveného hesla. [13]



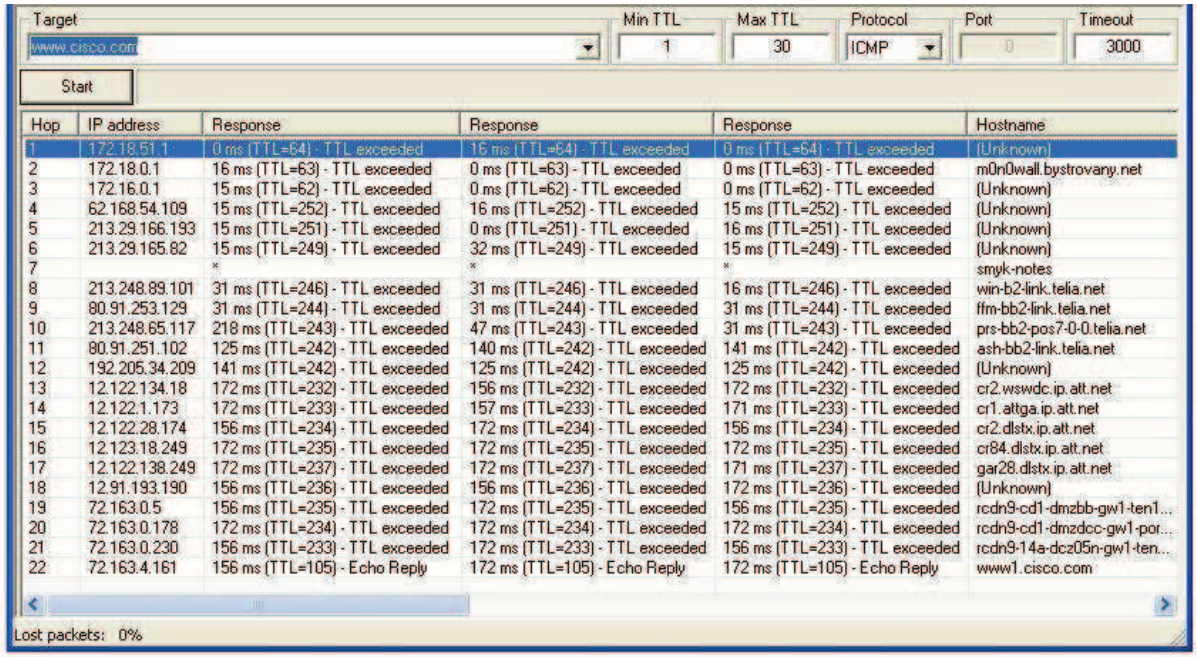
Obr. 12: Dialogové okno lámání SHA1 hash

4.7 Traceroute

Nástroj Traceroute programu Cain&Abel je v podstatě vylepšená verze Windows nástroje tracert. Nástroj tracert se používá pro sledování cesty mezi dvěma síťovými zařízeními, kdy se zobrazuje seznam skoků v síti, které byly napříč cestou úspěšně dosaženy. Tento seznam nám poskytuje důležité informace při ověřování a řešení problémů dostupnosti/nedostupnosti zařízení. Pokud data někde po cestě selžou a nejsou doručeny do cíle, z výpisu skoků zjistíme adresu posledního routeru, do kterého data ještě v pořádku došla. Díky tomuto nástroji zjistíme, kde je v síti problém nebo kde jsou nastavena taková bezpečnostní omezení, že neumožní komunikaci. Po cestě mohou být vyslané ICMP pakety zahozeny, aniž by byla zaslaná zpět informace, proč nebylo možné cíle dosáhnout. To se většinou stává například ve firewallech. K obejití běžných firewallových omezení se používá protokol TCP, proto nástroj Traceroute podporuje i vedle protokolu ICMP i protokoly TCP a UDP. [13]

Uživatelské rozhraní nástroje Traceroute (viz Obr. 13) se skládá z pole Target, kde se vkládá IP adresa nebo doména síťového zařízení, ke kterému chceme zobrazit cestu. Následují pole Min TTL a Max TTL, kde se definuje minimální a maximální počet přeskoků v síti. To znamená přes kolik routerů mohou do cíle pakety minimálně nebo maximálně procházet. Pokud se hodnota pole Max TTL nastaví na hodnotu 30, pokaždé kdy pakety vstoupí do routeru bude se tato hodnota dekrementovat o 1. Jakmile hodnota klesne na 0, paket se zahodí. Dále je pole Protocol, kde se vybere jeden z protokolů ICMP, TCP nebo

UDP. Primárně se pro trasování používá protokol ICMP, ale pro obejití běžných bezpečnostních pravidel na firewallech se může použít i TCP. Následuje pole Port, v kterém se nadefinuje číslo portu (u protokolu TCP nebo UDP), podle kterého firewally budou rozhodovat, jestli dané pakety pustí dále nebo je zahodí. Následuje pole Timeout, kde se definuje maximální doba, za kterou musí router odpovědět zpět, jinak se bude pokládat za nedosažitelný. Nakonec následuje tabulka, kde se vypisují informace, které byly získány z paketů putujících danou cestou. Ve sloupci hop se nachází číslo skoku v síti (pořadí routeru v cestě), IP adresa daného síťového rozhraní, tři sloupce, ve kterých jsou časy odezev paketů od každého routeru. Sloupce jsou tři, protože se odesílají tři pakety za sebou z důvodu, pokud se některý paket zpozdí a nestihne přijít včas zpět, byl by cíl označen za nedostupný, přestože dostupný může být. Zbylé pakety mohou dojít včas a potvrdit dostupnost daného síťového zařízení. Pokud paket úspěšně dorazí do routeru v cestě, pošle se zpět potvrzení ICMP Time Exceeded. Pokud paket dorazí úspěšně do cíle, zašle se potvrzení pomocí zprávy ICMP Echo Reply. Sloupec Hostname udává název daného síťového zařízení. Následují sloupce udávající síťovou masku, rozsah adres, síťové jméno, popisek, zemi výskytu síťového zařízení, název organizace přidělující danou IP adresu, IP adresu podsítě, vlastníka síťového zařízení, následuje číslo autonomního systému a název organizace přidělující číslo autonomního systému.



Hop	IP address	Response	Response	Response	Hostname
1	172.18.51.1	0 ms (TTL=64) - TTL exceeded	16 ms (TTL=64) - TTL exceeded	0 ms (TTL=64) - TTL exceeded	(Unknown)
2	172.18.0.1	16 ms (TTL=63) - TTL exceeded	0 ms (TTL=63) - TTL exceeded	0 ms (TTL=63) - TTL exceeded	m0n0wall.bystrovany.net
3	172.16.0.1	15 ms (TTL=62) - TTL exceeded	0 ms (TTL=62) - TTL exceeded	0 ms (TTL=62) - TTL exceeded	(Unknown)
4	62.168.54.109	15 ms (TTL=252) - TTL exceeded	16 ms (TTL=252) - TTL exceeded	15 ms (TTL=252) - TTL exceeded	(Unknown)
5	213.29.166.193	15 ms (TTL=251) - TTL exceeded	0 ms (TTL=251) - TTL exceeded	16 ms (TTL=251) - TTL exceeded	(Unknown)
6	213.29.165.82	15 ms (TTL=249) - TTL exceeded	32 ms (TTL=249) - TTL exceeded	15 ms (TTL=249) - TTL exceeded	(Unknown)
7		*	*	*	smyk-notes
8	213.248.89.101	31 ms (TTL=246) - TTL exceeded	31 ms (TTL=246) - TTL exceeded	16 ms (TTL=246) - TTL exceeded	win-b2-link.telia.net
9	80.91.253.129	31 ms (TTL=244) - TTL exceeded	31 ms (TTL=244) - TTL exceeded	31 ms (TTL=244) - TTL exceeded	ffm-bb2-link.telia.net
10	213.248.65.117	218 ms (TTL=243) - TTL exceeded	47 ms (TTL=243) - TTL exceeded	31 ms (TTL=243) - TTL exceeded	prs-bb2-pos7-0-0.telia.net
11	80.91.251.102	125 ms (TTL=242) - TTL exceeded	140 ms (TTL=242) - TTL exceeded	141 ms (TTL=242) - TTL exceeded	ash-bb2-link.telia.net
12	192.205.34.209	141 ms (TTL=242) - TTL exceeded	125 ms (TTL=242) - TTL exceeded	125 ms (TTL=242) - TTL exceeded	(Unknown)
13	12.122.134.18	172 ms (TTL=232) - TTL exceeded	156 ms (TTL=232) - TTL exceeded	172 ms (TTL=232) - TTL exceeded	cr2.wswdc.ip.att.net
14	12.122.1.173	172 ms (TTL=233) - TTL exceeded	157 ms (TTL=233) - TTL exceeded	171 ms (TTL=233) - TTL exceeded	cr1.attga.ip.att.net
15	12.122.28.174	156 ms (TTL=234) - TTL exceeded	172 ms (TTL=234) - TTL exceeded	156 ms (TTL=234) - TTL exceeded	cr2.dlstx.ip.att.net
16	12.123.18.249	172 ms (TTL=235) - TTL exceeded	172 ms (TTL=235) - TTL exceeded	172 ms (TTL=235) - TTL exceeded	cr84.dlstx.ip.att.net
17	12.122.138.249	172 ms (TTL=237) - TTL exceeded	172 ms (TTL=237) - TTL exceeded	171 ms (TTL=237) - TTL exceeded	gar28.dlstx.ip.att.net
18	12.91.193.190	156 ms (TTL=236) - TTL exceeded	156 ms (TTL=236) - TTL exceeded	172 ms (TTL=236) - TTL exceeded	(Unknown)
19	72.163.0.5	156 ms (TTL=235) - TTL exceeded	172 ms (TTL=235) - TTL exceeded	156 ms (TTL=235) - TTL exceeded	rcdn9-cd1-dmzbb-gw1-ten1...
20	72.163.0.178	172 ms (TTL=234) - TTL exceeded	172 ms (TTL=234) - TTL exceeded	172 ms (TTL=234) - TTL exceeded	rcdn9-cd1-dmzdcc-gw1-por...
21	72.163.0.230	156 ms (TTL=233) - TTL exceeded	172 ms (TTL=233) - TTL exceeded	156 ms (TTL=233) - TTL exceeded	rcdn9-14a-dcz05n-gw1-ten...
22	72.163.4.161	156 ms (TTL=105) - Echo Reply	172 ms (TTL=105) - Echo Reply	172 ms (TTL=105) - Echo Reply	www1.cisco.com

Obr. 13: Uživatelské rozhraní nástroje Traceroute

Tento nástroj programu Cain&Abel podle mého názoru nemá velkého uplatnění. Zastává funkci nástroje tracert, který je obsažen ve všech operačních systémech Windows a jeho rozšíření funkcí, která přináší Traceroute v tomto programu, nepovažuji za natolik důležité, abych si tento program stahoval jen kvůli tomuto nástroji. Jedná se spíše jen o doplňují informace, které nemají zásadní vliv na řešení problémů s nedostupností daných

síťových zařízení. Spíše se jedná o zajímavosti, jako je název vlastníka daného zařízení, číslo autonomního systému a název organizace přidělující číslo autonomního systému, odkud poznáme, na kterém kontinentu se dané síťové zařízení nachází (RIPE – Evropa, ARIN – Amerika, ...). Asi nejužitečnějším rozšířením oproti nástroje tracert je možnost volby protokolu TCP. Pokud víme, že se po cestě nachází firewall nebo access list na routeru, je možno testovat průchodnost těchto omezovačů spojení na určitých portech. Při práci s tímto nástrojem je vhodné vypnout nebo upravit nastavení firewallu.

4.8 CCDU

Zkratka CCDU znamená Cisco Config Downloader/Uploader a představuje vlastnost programu, která umožňuje stahovat nebo nahrávat konfigurační soubory síťových zařízení od výrobce Cisco. Tento nástroj umožní jednoduše nahrát konfiguraci do daného zařízení, aniž by administrátor musel ručně zadávat všechny příkazy. Konfigurační soubory jsou stahovány nebo nahrávány přes protokol SNMP/TFTP. Nástroj CCDU v programu Cain&Abel je možné používat pouze s routery a switchy, které používají MIB (Management Information Base) Old-Cisco-System-MIB nebo novější Cisco-Config-Copy-MIB. MIB je databáze objektů a veličin, které mohou být na zařízení spravovány. Tyto objekty a veličiny poskytují informace o síťových zařízeních a jeho rozhraních. PIX firewall tyto MIB nepodporují. [13]

Program Cain žádá síťového zařízení Cisco o přenos konfiguračních souborů pomocí protokolu SNMP. Pakety žádostí jsou vytvořené na základě proprietárních objektových identifikátorů, které prodejci poskytují pro tuto funkcionalitu. Pakety žádostí obsahují také další parametry, mezi které patří typ protokolu, IP adresa serveru a název souboru. Tyto informace jsou důležité, aby patřičné zařízení vědělo, kam konfigurační soubory posílat nebo odkud je nahrávat. Při přenosu konfiguračního souboru Cain otevře TFTP spojení (TFTP socket) do naslouchacího módu. Při přenosu konfiguračních souborů není potřeba TFTP server.

Pro přenos konfiguračních souborů Cisco routerů a switchů je potřeba zadat v dialogu Hostname zařízení nebo jeho IP adresu, SNMP Read/Write Community – jedná se o heslo k řízení přístupu SNMP klienta k serveru a Protocol – volba verze SNMP protokolu.

CCDU nástroj nefunguje, pokud jsou nastavena omezení přístupu pomocí přístupových seznamů nebo jsou na firewallu nastavena pravidla nepropouštějící SNMP/TFTP provoz. Tento nástroj má problémy i s dynamickým NAT. [13]

Nástroj CCDU dokáže být pro síťové administrátory velice užitečný, pokud spravuje lokální síť LAN složenou ze switchů a routerů Cisco. Administrátorovi stačí ručně nakonfigurovat dané zařízení pouze jednou, protože si daný konfigurační soubor může pomocí tohoto nástroje zálohovat. Druhé zařízení stejného typu již nemusí kompletně ručně konfigurovat jako v prvním případě, ale stačí pouze pomocí nástroje CCDU, nahrát zálohu konfiguračního souboru do daného zařízení a pak už jen ručně konfigurovat specifické

požadavky na zařízení, které nejsou již obsaženy v zálohovaném souboru. Ze síťového zařízení Cisco se stahuje nebo nahrává soubor „running-config“. Jedná se o soubor, který se po provedení konfiguračních změn, musí zkopírovat do souboru „startup-config“, jinak nebudou změny v konfiguraci uloženy a po restartu zařízení bude načtena konfigurace ze souboru „startup-config“. Výhodou nástroje CCDU je, že administrátoři nemusejí instalovat TFTP server, ale stačí jim pouze program Cain

Na druhou stranu je nástroj CCDU v rukou útočníka velice nebezpečný. Útočník s přístupem do sítě LAN, může jednoduše stáhnout konfigurační soubory síťových zařízení, pozměnit jejich obsahy k obrazu svému, zjistit nastavená hesla nebo hashe hesel, které může prolomit pomocí nástroje Cracker a zase soubory do zařízení nahrát zpět. Pokud si útočník může stáhnout konfigurační soubor, rázem se z něj může stát neomezený vládce nad zařízením.

4.9 Wireless

Pod kartou Wireless se v programu Cain&Abel ukrývá nástroj, který detekuje bezdrátové lokální sítě WLAN. Jedná se o nástroj Wireless Scanner a k ovládání bezdrátové síťové karty používá ovladač Winpcap Packet Driver. Jednotlivé přístupové body (AP) a ad-hoc sítě se vyhledávají každých pět sekund a parametry těchto bezdrátových sítí jsou zobrazeny do seznamu.

Active Scanner (aktivní skener) - aktivní skener pracuje tak, že pomocí ovladače Winpcap otevře adaptér bezdrátové sítě a následně zažádá pomocí funkce „PacketRequest“ o komunikaci s bezdrátovou síťovou kartou.

Passive Scanner (pasivní skener) - pasivní skener vyžaduje AirPcap adaptér od firmy CACE Technologies, který umožní zachytit rámce standardu 802.11 prostřednictvím ovladače AirPcap. Skener rozpozná bezdrátové přístupové body a klienty dekódováním rámců 802.11b/g, které putují vzduchem. Skener každou sekundu mění frekvenci adaptéru a tím může objevit bezdrátové sítě i na ostatních kanálech než který je nastavený.

Program Cain také podporuje techniku automatických ARP žádostí (ARP Requests injection) a umí zachytávat autentizační hashe zabezpečení bezdrátových síti WPA-PSK. Technika ARP Requests Injection umožní urychlit zachytávání unikátních WEP inicializací a je možná pouze se specifickým ovladačem Airpcap TX od firmy CACE Technologies. [13]

Útok na WEP - při zaškrtnutí checkboxu „Capture WEP IVs to dump.ivs file“ Cain začne zachytávat WEP inicializační zprávy do souboru „dump.ivs“. Tento soubor je umístěný v adresáři programu Cain. Tento soubor může být okamžitě otevřen při kliknutí na tlačítko Analyze. WEP IV jsou potřebné pro prolomení WEP šifrovacích klíčů, které jsou použity v chráněné bezdrátové síti. Odborné dokumentace uvádějí, že minimální počet WEP IV inicializačních zpráv potřebných k úspěšnému prolomení 64-bitových WEP klíčů pomocí

Korek útoku (statická crackovací metoda pro odkrytí WEP klíčů) je 250 000 a u 128-bitových klíčů 1 000 000. Při použití novějšího PTW útoků (podle jmen autorů Pyshkin, Tews, Weinmann) lze prolomit 128-bitové klíče již se 70 000 inicializačními zprávami. [13]

Tento nástroj programu Cain je užitečný pro vyhledávání dostupných bezdrátových sítí, umožní kontrolovat obsazení jednotlivých kanálů. O jednotlivých bezdrátových sítích nám tento nástroj poskytne informace jako BSSID (Basic Service Set IDentifier), což je MAC adresa přístupového bodu bezdrátové sítě, dále čas kdy skener danou síť naposledy uviděl, sílu signálu, název sítě (SSID - Service Set IDentifier). SSID musí být v dosahu dané sítě unikátní. Skener dále podává informace, jestli je provoz v dané bezdrátové síti šifrován, kanál a frekvenci na kterém síť pracuje, provozní rychlosti, počet přijatých paketů a unikátních WEP IV zpráv. Aktivním skenováním si útočník na základě zjištěných informací vytvoří obrázek o daných bezdrátových sítích, ale nemůže pomocí něco podniknout útok, ani nemůže sbírat WEP IV unikátní zprávy. Pasivní skenování již umožní provést ARP Requests injection nebo odeslat zachycené WPA-PSK autentizační hashe do Crackeru programu Cain. Pro pasivní skenování je,ale zapotřebí získat ovladač Airpcap TX a USB zařízení pro zachytávání paketů.

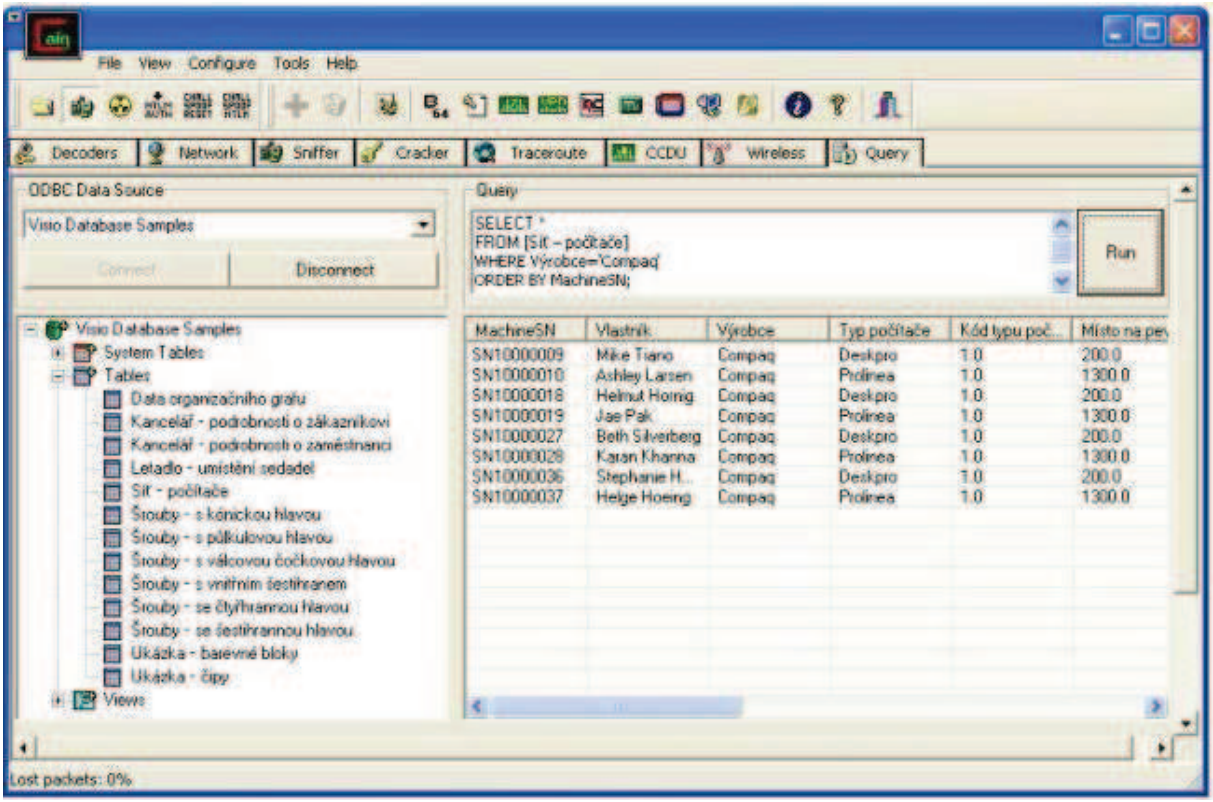
4.10 Query

Pod kartou Query se v programu Cain nachází nástroj, který umí spolupracovat s různými databázemi. Tento nástroj se umí připojit k velkému množství databází, ať už se jedná o profesionální databáze MS SQL, MySQL, MS Access a nebo se jedná o jednoduché databáze vytvořené v excelu. Po připojení k určité databázi se tento nástroj dokáže na danou databázi i dotazovat. Aby se tento nástroj dokázal k dané databázi připojit, musí se provést nastavení datových zdrojů (ODBC) v ovládacích panelech → nástroje pro správu. Zde se specifikuje, kde se daná databáze nachází a kam se tedy Cain má připojit. [13]

Práce s nástrojem je následující. V sekci „ODBC Data Source“ se zvolí zdroj dat a klikne na tlačítko „Connect“. Nástroj načte data ze zdrojového souboru a zobrazí seznam tabulek. Po zvolení některé tabulky se zobrazí její obsah a v sekci „Query“ je možné psát dotazy na databázi.

Na obrázku (viz Obr. 14) je vidět příklad použití nástroje Query, který načetl data z databáze v programu Microsoft Visio. V sekci Query je vidět dotaz na tabulku „Síť - počítače“. Tento dotaz vykonal požadavek, aby databáze vypsala všechny řádky tabulky „Síť - počítače“, ve kterých je výrobce Compaq a seřadila je podle MachineSN.

Práce s tímto nástrojem vyžaduje znalosti příkazů z prostředí databází. V tomto nástroji je možné provádět na databáze pouze dotazy, neumožňuje databáze tvořit ani editovat.



Obr. 14: Nástroj Query pro práci s databázemi

4.11 Tools

4.11.1 Route Table

Pomocí tohoto nástroje je možné zobrazit směrovací tabulku počítače, na kterém je nainstalován program Cain. Umožňuje i přidat nové routy do této tabulky nebo modifikovat či odebrat routy již přítomné v tabulce. Při vytváření nové routy se zadá cílová IP počítače nebo sítě, síťová maska, IP adresa gateway a IP adresa výstupního rozhraní daného počítače. Ještě je možné definovat metriku dané routy.

Tento nástroj poskytuje prakticky stejné informace, jaké poskytuje ve Windows příkazové řádce nástroj route.

4.11.2 TCP/UDP Tables

Pod položkou Route Table se nachází nástroj programu Cain TCP/UDP Table Viewer. Poskytuje informace o parametrech spojení, které počítač navázal. Tabulka v každém řádku tabulky zobrazuje informace o názvu procesu, typ protokolu (TCP nebo UDP), IP adresa počítače, na kterém je Cain nainstalován, dále na kterém portu se spojení navazuje, IP adresa vzdáleného stroje, s kterým je spojení vytvořeno a na jakém portu a u protokolu TCP je i stav spojení.

Tento nástroj není příliš užitečný, protože ve Windows příkazové řádce každého počítače je nástroj „netstat“, který poskytuje stejné informace, dokonce jich poskytuje mnohem více než nástroj v programu Cain.

4.11.3 Base64 Password Decoder

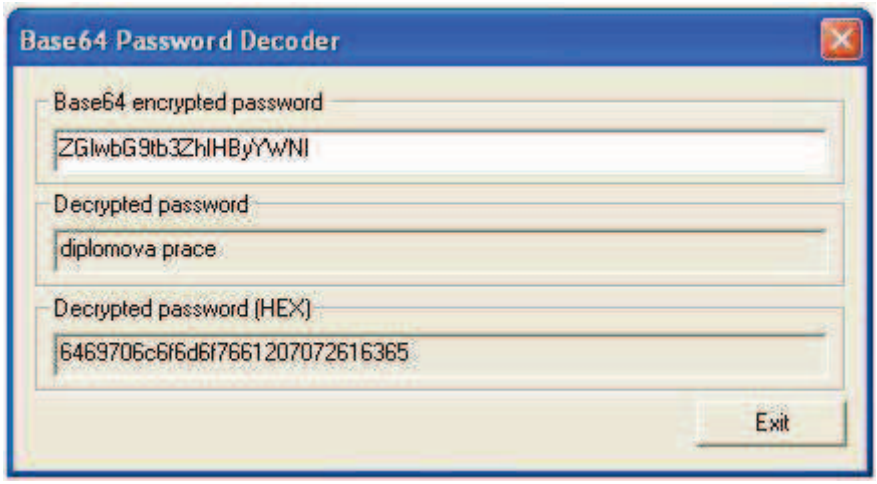
Jedná se o nástroj programu (viz Obr. 15), který umožňuje dekódovat hesla nebo text z base64 kódu zpět do původní podoby. Tento nástroj umí pouze base64 kód dekódovat, kodér v programu Cain&Abel není přítomen.

Kódování base64 je navrženo k tomu, aby reprezentovalo libovolné posloupnosti oktetů ve formě, která vyžadují citlivost na velká a malá písmena, ale nemusí být pro lidi čitelná.

Při procesu kódování do base64 kódu se postupuje tak, že se vstupní bity rozdělí do skupin po 24 bitech, kdy tato skupina je prakticky tvořena 3 skupinami po 8 bitech, kde každý znak v ASCII tabulce je vyjádřen právě 8 bity. Skupina 24 bitů se rozdělí do 4 skupin po 6 bitech, kdy každá skupina těchto bitů (2^6 stavů)bude vyjádřena jako jeden z 64 znaků base64 abecedy. [8]

Base64 se používá v několika internetových protokolech (HTTP, MIME, IMAP) pro zakódování libovolných dat jako prostý text ASCII znaků. Dále se používá například při šifrování binárních dat (multimédia) nebo uložení hashů hesel vypočtených šifrováním v /etc/passwd a nebo spammeři používají base64, aby se vyhnuli základním anti-spamovým nástrojům, které nedekódují tento formát a nemohou tedy zjistit klíčová slova v zakódovaných zprávách. [8, 13]

Tento nástroj je omezen na dekódování hesel a textu dlouhého pouze 76 znaků v base64 kódu. Delší text je nutné rozdělit na menší skupiny, ale musí se dát pozor, aby daná skupina obsahovala počet base64 znaků v násobku 4, jinak poslední znak bude dekódován špatně. Při práci s nástrojem se do pole „Base64 encrypted password“ vloží heslo v base64 kódu a poli „Decrypted password“ se zobrazí hledané dešifrované heslo a v poli „Decrypted password (HEX)“ se zobrazí dešifrované heslo v hexadecimálním tvaru.



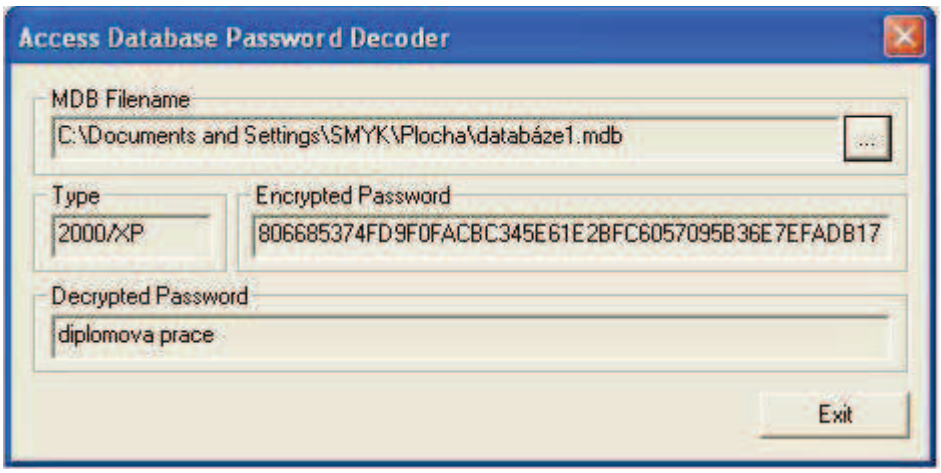
Obr. 15: Dekodér base64 kódu

4.11.4 Access Database Password Decoder

Nástroj Access Database Password Decoder je nástroj programu, který umožní odkrýt heslo chráněného databázového souboru. Jedná se o soubory s příponou „.mdb“ vytvořené programem na tvorbu databází Microsoft Access.

Přístupové heslo k souboru je uloženo přímo v daném databázovém souboru a jako šifrovací techniku používá jednoduché logické funkce XOR. XOR je možné okamžitě otočit a ihned dešifrovat. Při pokusech dešifrovat co nejdelší heslo bylo zjištěno, že maximální délka hesla, které lze dešifrovat, je omezena na 18 znaků. Pokud bude heslo delší jak 18 znaků, dekodér zobrazí správně pouze každý sudý znak hesla. Místo lichých znaků hesla se objeví tlusté svislé čáry. I přes toto omezení považuji nástroj za plně vyhovující, protože naprosté minimum uživatelů si zvolí heslo delší jak 18 znaků. [13]

Práce s nástrojem (viz Obr.16) je plně automatická, pouze v sekci „MDB Filename“ se musí vložit cesta k heslem chráněnému „.mdb“ souboru. V sekci „Encrypted Password“ se zobrazí šifrované heslo v hexadecimálním tvaru a v sekci „Decrypted Password“ se zobrazí hledané heslo k databázi v podobě čistého textu. Tento nástroj je například velice užitečný, když uživatel zapomene heslo k dané databázi. Díky tomuto nástroji si heslo rychle a jednoduše zjistí. Díky tomu odpadají bezpečnostní rizika v podobě lístečků přilepených na monitoru s heslem k dané databázi.



Obr. 16: Nástroj pro dešifrování heslem chráněných Microsoft Access souborů

4.11.5 Cisco Type-7 Password Decoder

Cisco Type-7 Password Decoder je nástroj programu, který dokáže dekódovat přístupová hesla k síťovým zařízením od firmy Cisco.

Cisco používá svůj vlastní šifrovací algoritmus k šifrování hesel, které se používají pro přístup k síťovému zařízení přes konzolový kabel nebo pomocí vzdáleného přístupu přes takzvané vty linky. Cisco používá algoritmus Type-5 a Type-7. Tento nástroj dokáže dekódovat pouze hesla šifrovaná algoritmem Type-7. Šifrování pomocí algoritmu Type-7 se

musí aktivovat v příkazovém řádku Cisco IOS pomocí příkazu „**service password-encryption**“. [7]

Hesla šifrovaná pomocí algoritmu Type-7 je možné nalézt v konfiguračních souborech „startup-config“ nebo „running-config“. Takto zašifrované heslo se zkopíruje do nástroje Cisco Type-7 Password Decoder (viz Obr. 17) a ten automaticky dešifruje dané heslo do podoby čistého textu.

Pomocí programů Packet Tracer a Cain&Abel se podařilo zjistit, že délka hesla nemá vliv na rychlost odkrytí hledaného hesla. Hledané heslo bude mít maximálně 25 znaků, protože delšího heslo nelze dešifrovat a ani v programu Packet Tracer nelze síťovému zařízení delší heslo nastavit. I kdyby šlo na fyzickém zařízení nastavit delší heslo, domnívám se, že téměř žádný administrátor z praktických důvodů tak dlouhá hesla nepoužívá. Proto považuji tuto maximální délku dešifrovatelného hesla za plně dostačující.

Hesla šifrovaná pomocí algoritmu Type-5 nelze tímto nástrojem dešifrovat. Je ale možné zašifrovaná hesla vložit do Cisco IOS-MD5 Hashes Crackeru v programu Cain a aplikovat na něj slovníkový útok nebo útok hrubou silou.

Obr. 17: Cisco Type-7 Password Decoder

4.11.6 Cisco VPN Client Password Decoder

Tento nástroj programu umožňuje dešifrovat hesla VPN klientského softwaru od firmy Cisco, která se ukládají do profilových souborů s příponou „.pcf“. Tyto profilové soubory obsahují všechny potřebné parametry ke vzdálenému připojení přes síť VPN.

Software šifruje hesla pomocí šifrovacích algoritmů SHA1 a 3DES a jsou nezávislé uživatelsky i na zařízení. [13]

Tento nástroj pracuje automaticky a pro dešifrování stačí vložit šifrované heslo do pole „Cisco VPN Profile encrypted password“. V poli decrypted password se zobrazí dešifrované heslo v podobě čistého textu.

Funkci tohoto nástroje nebylo možné ověřit, protože jsem neměl přístup k žádnému Cisco klientskému softwaru ani Cisco síťovému zařízení, ke kterému bych mohl přistupovat přes VPN.

4.11.7 VNC Password Decoder

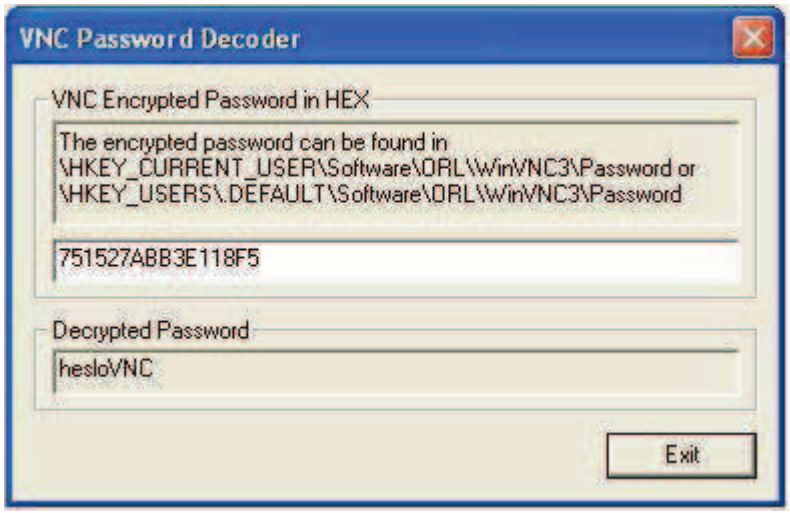
VNC Password Decoder je nástroj programu Cain (viz Obr. 18), který dokáže dekódovat hesla VNC řešení. VNC (Virtual Network Computing) je vzdálené desktopové řešení, které umožní zobrazit plochu a plně ovládat počítač z jiného počítače nebo mobilního zařízení odkudkoliv z Internetu. VNC software umožňuje vzdálenou kontrolu mezi odlišnými typy počítačů i operačních systémů. VNC pracuje na principu klient – server, kdy heslo se nastavuje na serveru a klientovi je umožněno vzdálené řízení počítače, na kterém je nainstalován VNC server, na základě znalosti tohoto hesla. Heslo se ukládá na serveru do registru v hexadecimálním tvaru. [13]

V případě softwaru RealVNC je heslo uloženo v registru:

\\HKEY_CURRENT_USER\\Software\\RealVNC\\WinVNC4\\Password

a ne pod cestou jakou uvádí nástroj VNC Password Decoder.

VNC Password Decoder umí dešifrovat heslo v podobě čistého textu maximálně 8 znaků dlouhé. Myslím si, že heslo kratší než 8 znaků není zrovna bezpečné, z čehož bych usuzoval, že si uživatelé nastaví heslo delší. Jelikož ale uživatel počítače, který bude vzdáleně ovládán, většinou u daného počítače sedí a sleduje, co vzdálený pomocník s jeho počítačem dělá, kdykoliv při nekalé činnosti pomocníka může komunikaci přerušit. Proto se hesla volí spíše krátká, jednoduše srozumitelná a vzdálené ovládání přes VNC se po skončení činnosti nebo pomoci okamžitě vypne.



Obr. 18: VNC Password Decoder

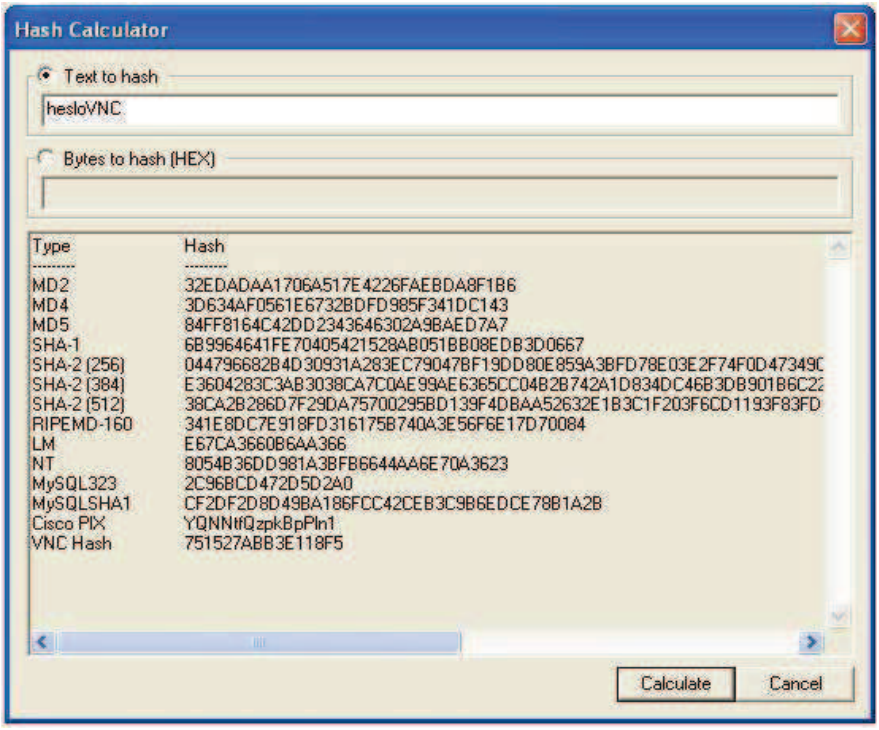
4.11.8 Hash Calculator

Jedná se o nástroj programu Cain, který dokáže z vloženého řetězce znaků vytvořit jejich hashe. Tento nástroj je schopný ze vstupního řetězce vytvořit 11 různých hashů. Hash Calculátor podporuje tyto hashovací algoritmy:

MD2, MD4, MD5, SHA-1,SHA-2 (256), SHA-2 (384), SHA-2 (512), RIPEMD-160, LM, NT, MySQL323, MySQLSHA1, Cisco PIX, VNC Hash. [13]

Nástroj tedy vypočítá například nejpoužívanější algoritmy MD (Message Digest) a SHA (Secure Hash Algorithm), kdy čísla v závorkách znamenají délku hashe v binárním tvaru. RIPEMD-160 (RACE Integrity Primitives Evaluation Message Digest) je algoritmus, který má plnit roli bezpečné náhrady za algoritmy MD. LM a NT jsou hashe, které se používají pro autentizaci v operačních systémech Microsoft, MySQL323 a MySQLSHA1 jsou hashe využívané při autentizaci k databázovému serveru MySQL, Cisco PIX hashe se používají u firewallů od firmy Cisco a VNC Hash je hash používají se při autentizaci k VNC serveru. Všechny hashe jsou v tomto nástroji zobrazeny v hexadecimálním tvaru.

Tímto nástrojem můžeme ověřit správnost dekódovaného hesla z VNC Password Decoderu z kapitoly 4.11.7. Pokud do Hash Calculatoru (viz Obr. 19) vložíme dekódované heslo z VNC Password Decoderu, dostaneme zase původní hash daného hesla.



Obr. 19: Hash Calculator

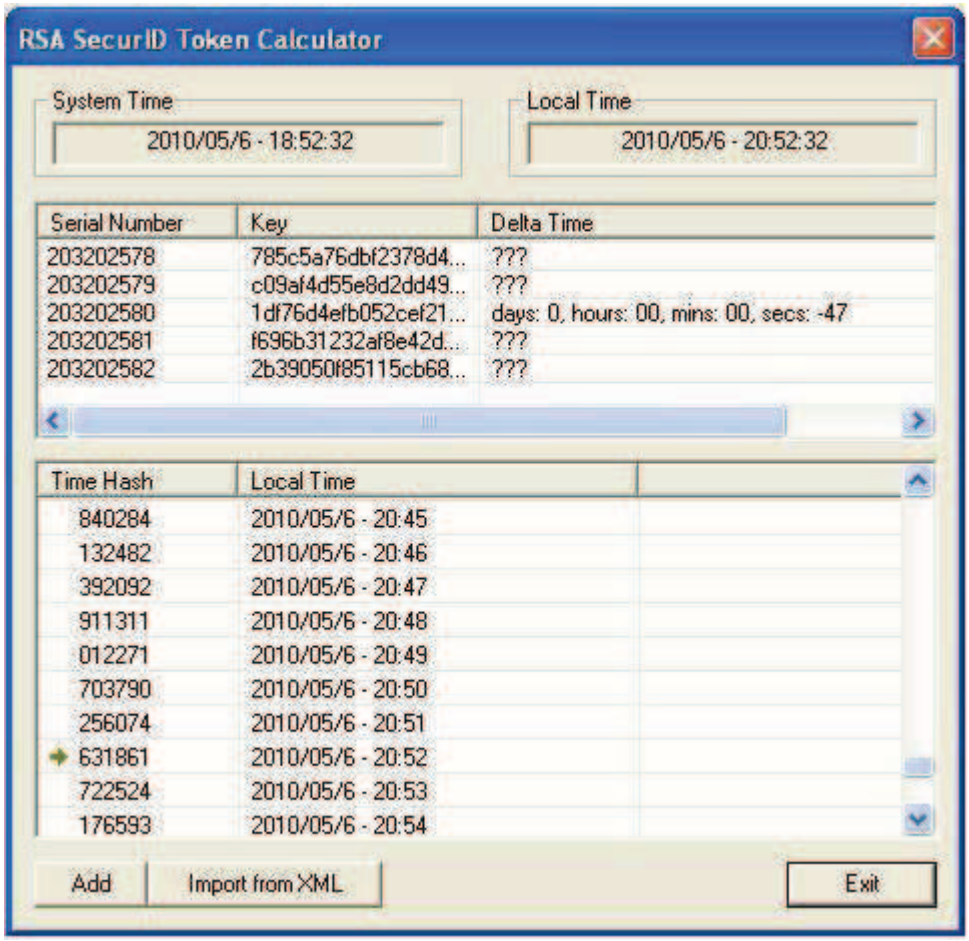
4.11.9 RSA SecurID Token Calculator

RSA SecurID Token se používá všude tam, kde je potřeba bezpečně autentizovat uživatele, především v bankovnictví. Jedná se o hardwarový generátor šestimístných číselných náhodných přístupových kódů, které se pravidelně obměňují každých 60 sekund. Pomocí RSA SecureID Tokenů se dosáhne dvou faktorové autentizace. Například když uživatel chce využívat elektronického bankovnictví, autentizuje se zadáním tajného personálního osobního čísla (PIN) a následně ještě zadáním aktuálně zobrazeného číselného kódu na tokenu. Algoritmus pro generování číselných kódů používá v podstatě dva parametry. Sériové číslo a aktivační klíč tokenu. Oba parametry obvykle prodejce poskytuje v XML souboru. Hodnoty tokenu vypočítány každé dvě minuty a zobrazeny každých 60 sekund. Aby autentizační systém RSA SecureID fungoval správně, je nutné, aby byl systémový čas synchronizován s interním časem tokenu. Pokud by tento čas nebyl synchronizován, zobrazované hodnoty na tokenu by nesouhlasily s hodnotami, které systém akceptuje pro autentizování uživatele. [9, 13, 16]

Nástroj RSA SecureID Token Calculator programu Cain je nástroj, který dokáže generované autentizační přístupové kódy zobrazit ještě před tím, než se objeví na displeji tokenu. Z důvodu správné synchronizace tokenu nástroj musí uvažovat odchylku mezi systémovým časem a interním časem tokenu. Synchronizace času se musí provést pro každý token zvlášť v synchronizačním dialogu, který vyžaduje, aby uživatel vložil právě zobrazený číselný kód na tokenu.

Pro použití tohoto nástroje je nutné mít od prodejce k dispozici XML soubor, který se do tohoto nástroje musí importovat, aby program načetl potřebné informace o tokenu. XML soubor obsahuje informace o tokenu jako je sériové číslo, seed tokenu, datum aktivace tokenu a datum konce platnosti tokenu a fyzickou adresu tokenu. Seed tokenu je aktivační klíč tokenu. Po importování XML souboru se v nástroji programu zobrazí sériové číslo tokenu, seed tokenu (klíč) a odchylka mezi časem systému a interním časem. Následně se provede synchronizace tokenu s nástrojem, což se provede kliknutím pravým tlačítkem na řádek se sériovým číslem tokenu a do následně zobrazeného dialogové okna se zadá právě zobrazený číselný kód na tokenu. Po synchronizaci se zobrazí dva sloupce s hodnotami. V jednom sloupci jsou šestimístné přístupové autentizační kódy a v druhém sloupci je datum a čas, kdy bude daný kód správný pro autentizování. Přístupové kódy s příslušnými časy lze uložit do textového souboru. Při ukládání těchto hodnot do souboru je možné si zvolit dobu, kolik minut, hodin, dní, měsíců nebo roků chceme tyto hodnoty dopředu vygenerovat. [13]

Na Obr. 20 a Obr. 21 je vidět funkce nástroje RSA SecureID Token Calculator a způsob, jakým se útočník může autentizovat na RSA serveru, aniž by měl Token fyzicky k dispozici. Pro útočníka je velmi důležité mít synchronizovaný čas s RSA systémem, protože pokud by měl útočník na svém počítači třeba jen o minutu odlišný čas oproti RSA systému, nemohl by se autentizovat, protože by Cain aktuální přístupový kód ukazoval právě o tuto časovou odchylku s předstihem nebo zpožděně. Na Obr. 20 je vidět posloupnost přístupových kódů měnících se pravidelně po jedné minutě. Na Obr. 21 je vidět přístupový kód na tokenu, který je shodný s přístupovým kódem právě zobrazeným v programu Cain na Obr. 20.



Obr. 20: Použití nástroje RSA SecureID Token Calculator v programu Cain

Obr. 21: RSA SecureID Token

4.11.10 Remote Desktop Password Decoder

V operačních systémech Microsoft je integrován nástroj umožňují se připojit na vzdálený počítač a zobrazit jeho pracovní plochu přesně tak, jak ji vidí lokální uživatel. Tento nástroj se nazývá Remote Desktop (vzdálená plocha). Pokud se administrátor nebo nějaký uživatel často připojuje k více vzdáleným strojům, je možné si parametry spojení uložit do „.RDP souboru“. Do tohoto souboru se uloží IP adresa vzdáleného stroje, uživatelské jméno, heslo a doména. Pokud se v přihlašovacím dialogu zatrhne „Povolit uložení pověření“, přihlašovací údaje se zašifrují pomocí aplikačního programového rozhraní CryptProtectData a uloží se. [13]

Nástroj programu Cain Remote Desktop Password Decoder dokáže přihlašovací údaje z RDP souboru dešifrovat. Nástroj pracuje automaticky, stačí pouze v sekci RDP file vložit cestu k uloženému souboru a nástroj sám zobrazí uživatelské jméno, doménu, zašifrované heslo v hexadecimálním tvaru a dešifrované heslo v podobě čistého textu.

Tento nástroj v rukou útočníka je velice nebezpečný, protože z RDP souborů může útočník získat přihlašovací údaje ke všem strojům, ke kterým byly tyto soubory vytvořeny. Pokud se dostane k RDP souborům s parametry spojení na servery, jedná se o velký bezpečnostní problém, protože si útočník může v síti prakticky dělat co se mu zlíbí.

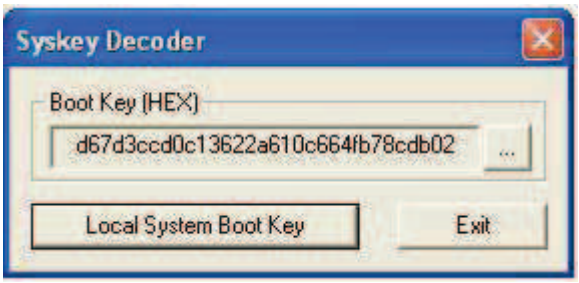
Nevýhodou pro útočníka je, že jelikož bylo využito aplikačního programového rozhraní CryptProtectData, zašifrované heslo je možné dešifrovat z RDP souboru pouze z uživatelského účtu a fyzického stroje, na kterém byly RDP soubory vytvořeny. To znamená, musí mít fyzický přístup k počítači a musí znát přihlašovací údaje k uživatelskému účtu, na kterém byly RDP soubory vytvořeny a uloženy.

4.11.11 Syskey Decoder

Syskey Decoder je nástroj, který extrahuje bootovací klíč, který je generovaný utilitou SYSKEY z lokálního registru nebo souboru SYSTEM. Bootovací klíč je informace, kterou využívá utilita SYSKEY k zašifrování hashů hesel předtím než jsou uloženy do SAM databáze v operačním systému Windows pomocí 128 bitového šifrovacího klíče. Pokud jsou tyto soubory uloženy lokálně, bootovací klíč je ukryt mezi klíče v registru

HKEY_LOCAL_MACHINE \System\CurrentControlSet\Control\Lsa

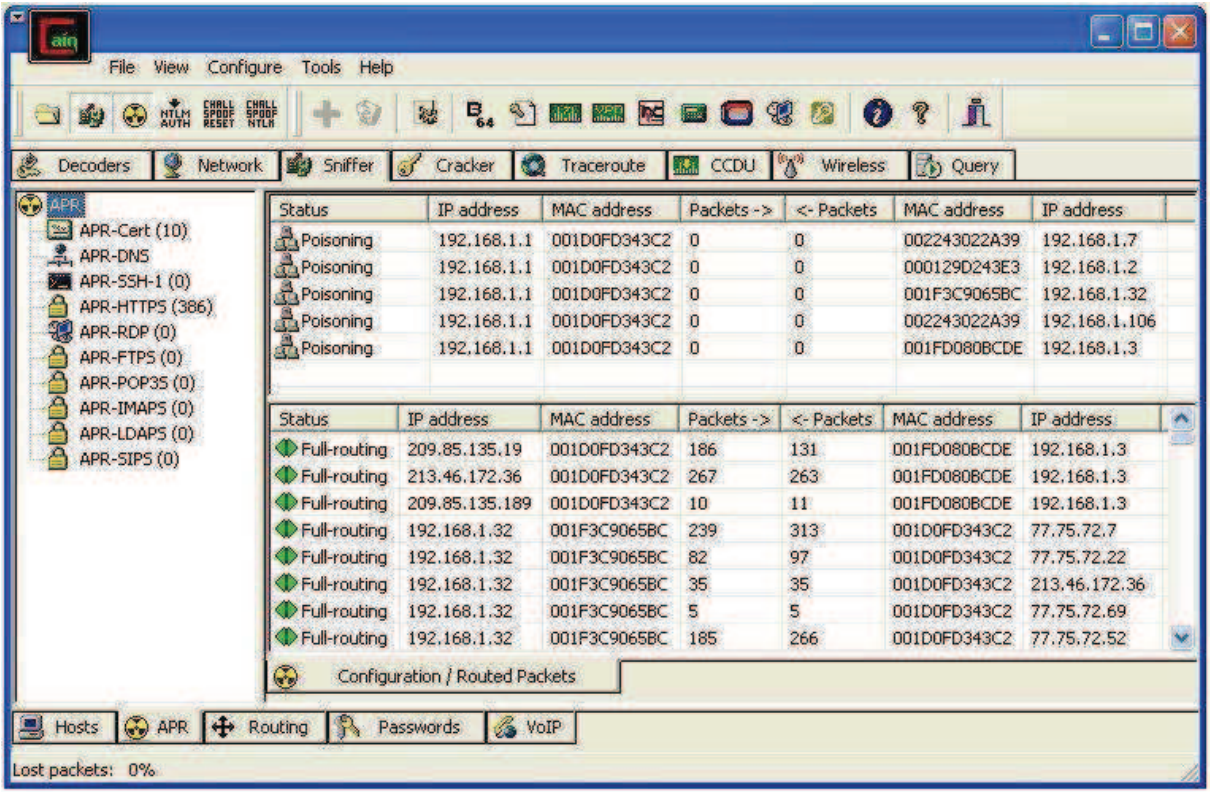
Nástroj Syskey Decoder (viz Obr. 22) může rekonstruovat tyto informace do hexadecimálního tvaru a následně jsou připraveny pro nástroj programu Cain NT Hashes Dumper. Nástroj Syskey pracuje automaticky a po kliknutí na tlačítko Local System Boot Key se zobrazí lokální bootovací klíč v hexadecimálním tvaru. [13]



Obr. 22: Grafické rozhraní nástroje Syskey Decoder

5 Praktická ukázka útoků pomocí programu Cain

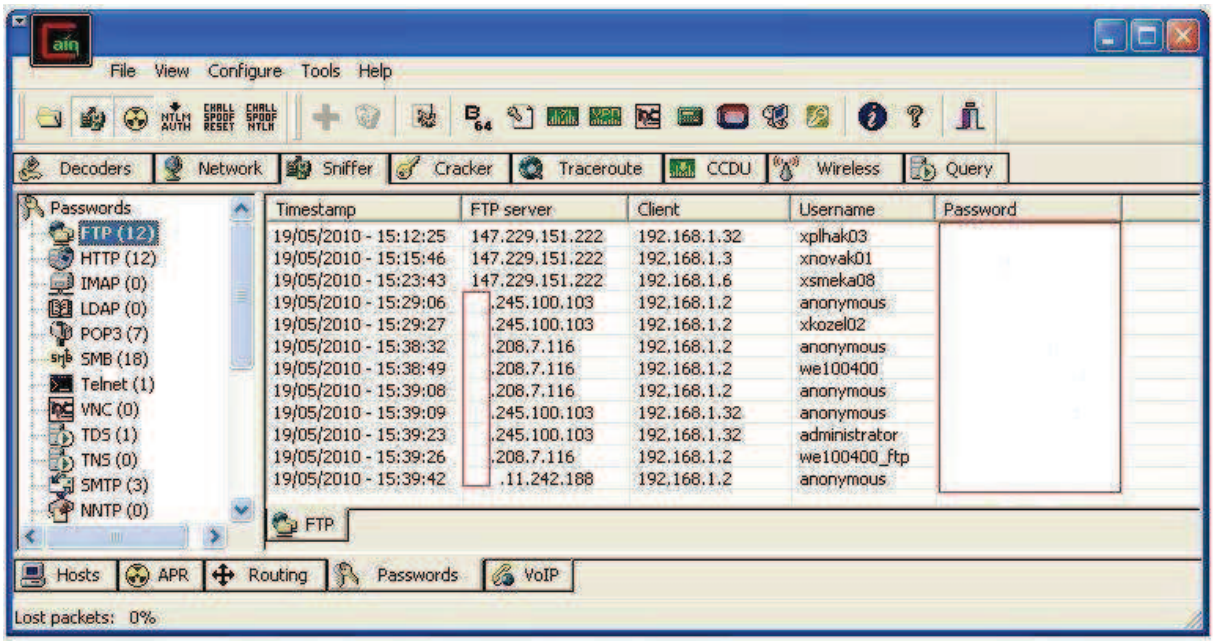
Po prozkoumání všech nástrojů, zjištění a vyzkoušení jejich možností jsem program Cain nainstaloval na počítač, který se nacházel v domácí lokální síti LAN, kde se nacházelo šest počítačů. Čtyři počítače byly připojeny pomocí pevného připojení k portům routeru a dva počítače byly připojeny pomocí bezdrátového připojení. Po instalaci programu Cain jsem nakonfiguroval sniffer, aby zachytával informace z konkrétní síťové karty a přepnul síťovou kartu do promiskuitního módu (viz kapitola 4.2.1). Po konfiguraci snifferu jsem použil nástroj pro skenování MAC adres v síti, abych zjistil MAC adresy hostů v dané síti (viz kapitola 4.5.1). Následně jsem nakonfiguroval nástroj pro ARP Poison Routing (APR), aby zprostředkoval komunikaci mezi výchozí bránou a jednotlivými hosty přes můj (útočník) počítač. Všechny stavy spojení se nacházeli ve stavu „Idle“. Do aktivního stavu jsem tato spojení uvedl aktivováním snifferu a nástroje APR. Jakmile uživatelé jednotlivých počítačů začali komunikovat přes bránu do internetu, jejich komunikace začala přecházet přes můj počítač. Tento stav indikuje status „Full-routing“, což znamená, že komunikace procházela přes můj počítač v obou směrech. Tento stav je patrný z obrázku 23.



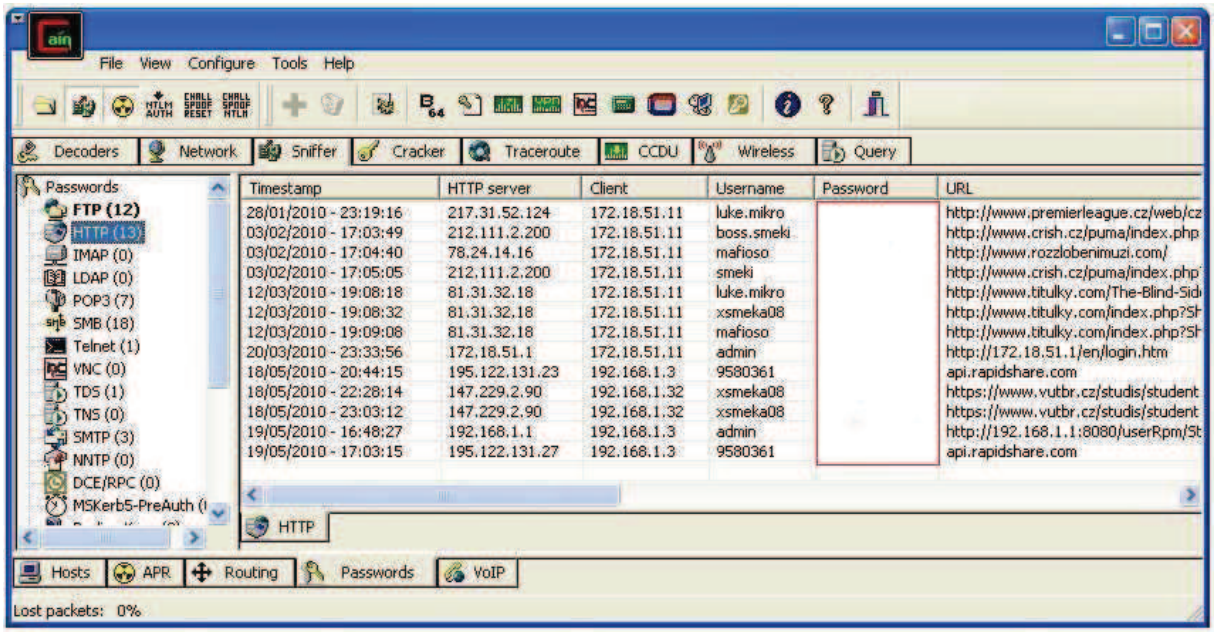
Obr. 23: Full-routing

Jakmile uživatelé jednotlivých počítačů postupně začali používat jednotlivé služby, aplikace a přihlašovat se k účtům na internetových stránkách, sniffer začal analyzovat jednotlivé protokoly a prohledávat je pro získání uživatelských jmen a hesel. Po několika hodinách se mi podařilo odchytit uživatelská jména a hesla z komunikace probíhající přes nešifrované protokoly FTP (viz Obr. 24), HTTP (viz Obr. 25), dále přihlašovací informace k aplikaci ICQ nebo k emailovým účtům, ke kterým se přistupuje například přes Outlook.

Dále se mi podařilo zachytit přihlašovací údaje pro přístup k routeru nebo přihlašovací údaje účtů na ústředně Asterisk a zachytit a nahrát VoIP hovor. Hesla a některé části IP adres jsou na obrázcích záměrně z bezpečnostních důvodů skryty.



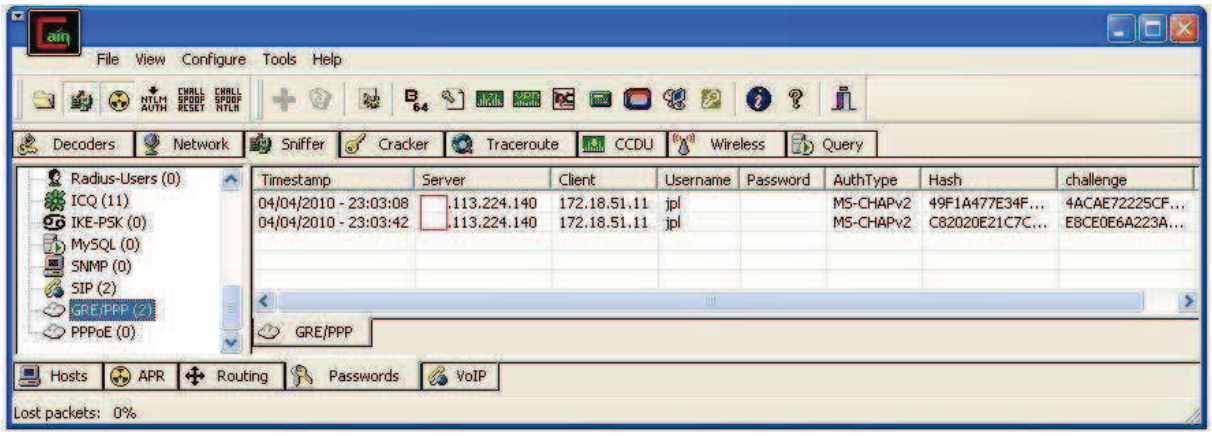
Obr. 24: Zachycené přihlašovací údaje k FTP serveru



Obr. 25: Zachycené přihlašovací údaje k webovým serverům

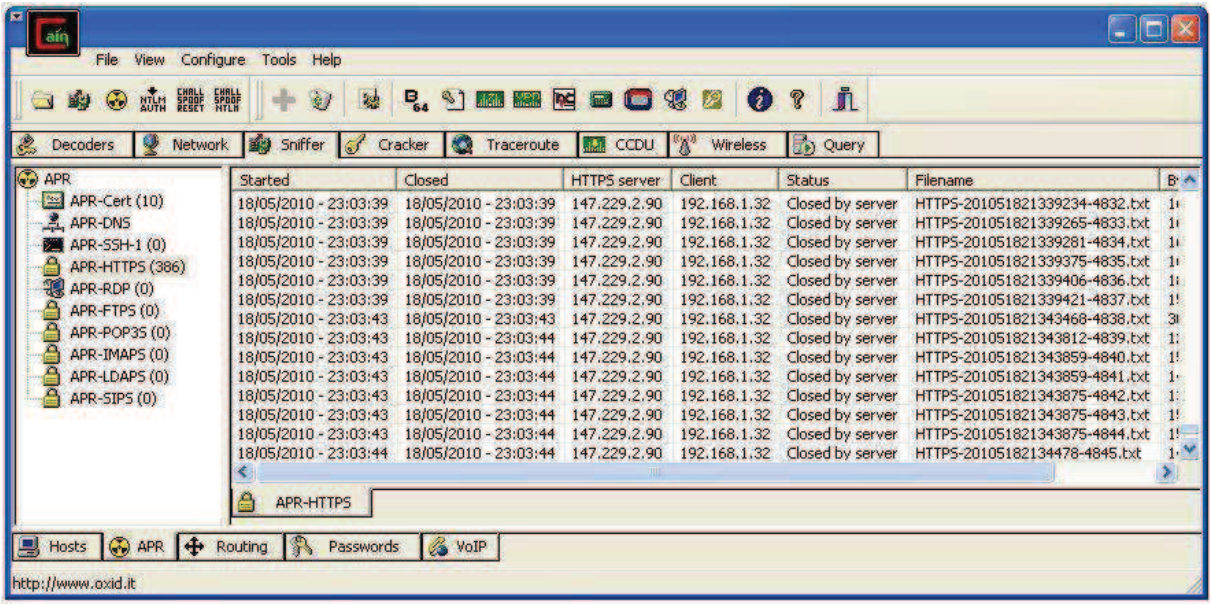
Následně se mi podařilo zachytit i autentizační údaje použité při vytváření VPN spojení do firemní sítě. V tomto případě se mi zobrazilo pouze uživatelské jméno a hash hesla. Proto jsem tento hash poslal do nástroje Cracker a aplikoval jsem na něj slovníkový útok, který nebyl úspěšný. Poté jsem aplikoval útok hrubou silou a díky krátké délce hesla se

mi podařilo dané heslo získat. Jednalo se o ukázkový příklad použití slabého hesla. Kdyby heslo bylo delší jak 8 znaků, obsahovalo malou i velkou abecedu, číslice i speciálník znaky, dané heslo by se mi nepodařilo získat dříve než za několik let.

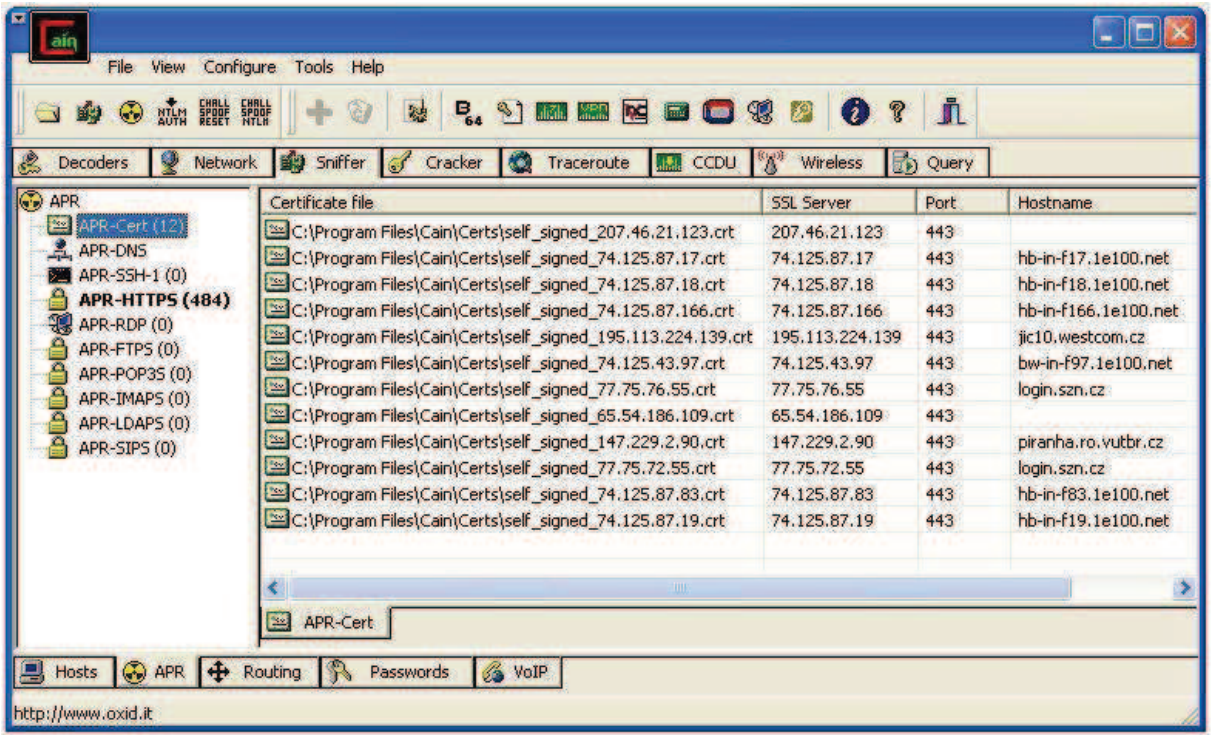


Obr. 26: Zachycené přihlašovací údaje VPN spojení.

Během mého sniffování síťového provozu se někteří uživatelé pokusili komunikovat s webovými servery, které komunikují přes zabezpečený protokol HTTPS. Webový server zaslal uživateli certifikát, ve kterém jsou informace prokazující identitu serveru. Tento certifikát, ale nedošel uživateli, který požádal o komunikaci, ale zachytil ho nástroj APR na mém počítači. Program Cain pozměnil veřejný klíč tohoto certifikátu a poslal danému uživateli. Uživateli se objevila hláška, která ho informovala, že certifikát pochází z nedůvěryhodného zdroje. Tato hláška se objevuje poměrně často a proto uživatel tento certifikát přijal. Díky tomuto kroku je možné, aby šifrovaná komunikace přišla do počítače, kde je program Cain, tam byla dešifrována, uložena a následně zašifrována a poslána uživateli nebo serveru (podle směru komunikace). Zachycená HTTPS komunikace (viz Obr. 27) je ukládána do textových souborů do adresáře „HTTPS“ v hlavním adresáři programu. Zachycené certifikáty (viz Obr. 28) jsou ukládány do adresáře „Certs“.

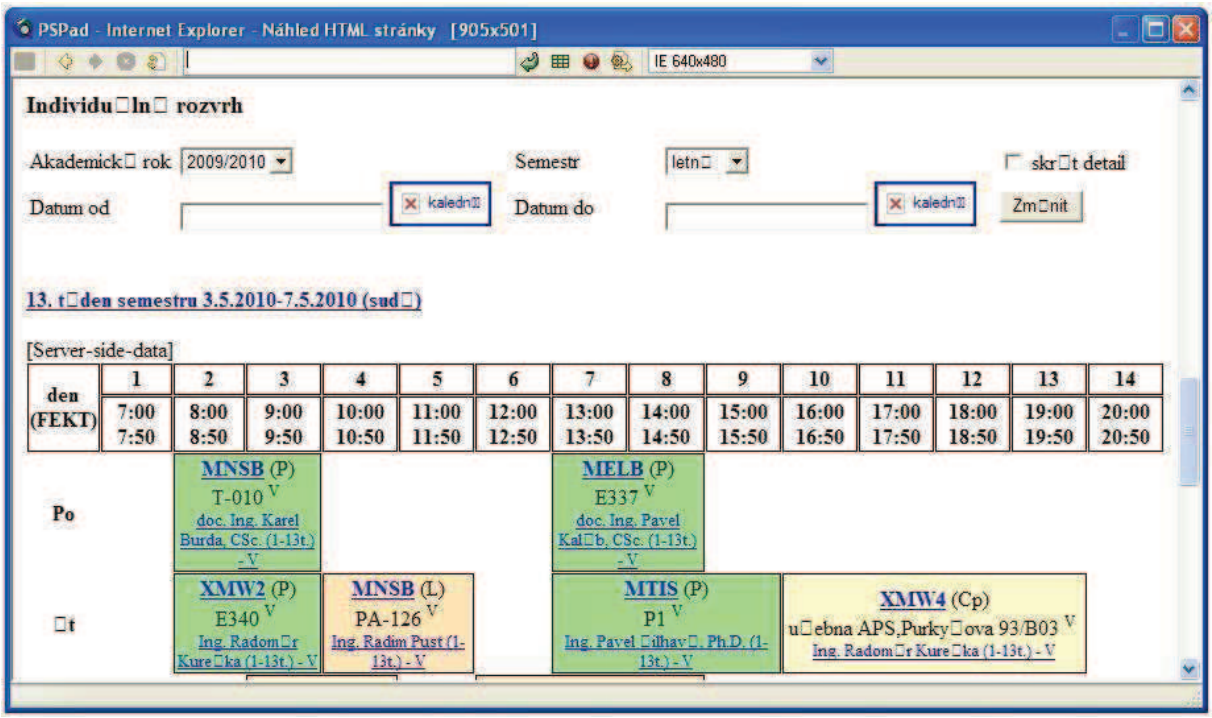


Obr. 27: HTTPS komunikace

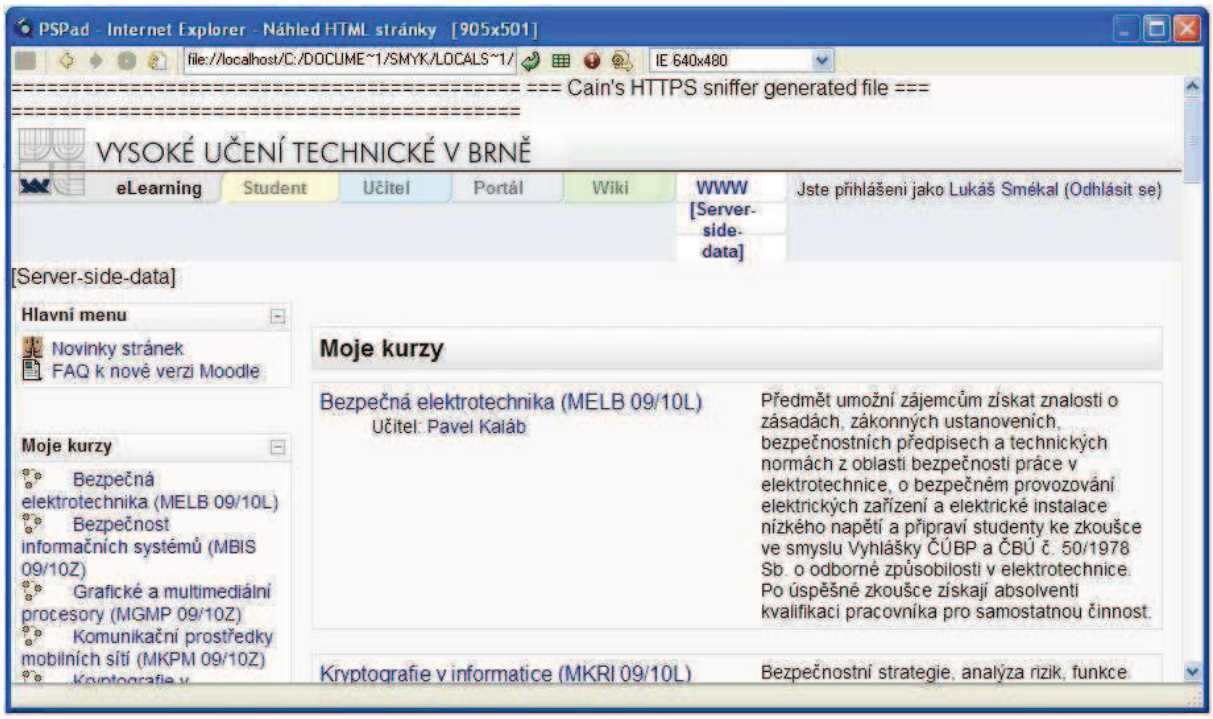


Obr. 28: Zachycené certifikáty

Ze zachycené HTTPS komunikace jsem zjistil, že se jedná o komunikaci s informačním systémem VUT a po vložení dešifrovaných zdrojových kódů do programu PSPad jsem si prohlédl obsah webových stránek (viz Obr. 29 a Obr. 30), které si daný uživatel prohlížel.

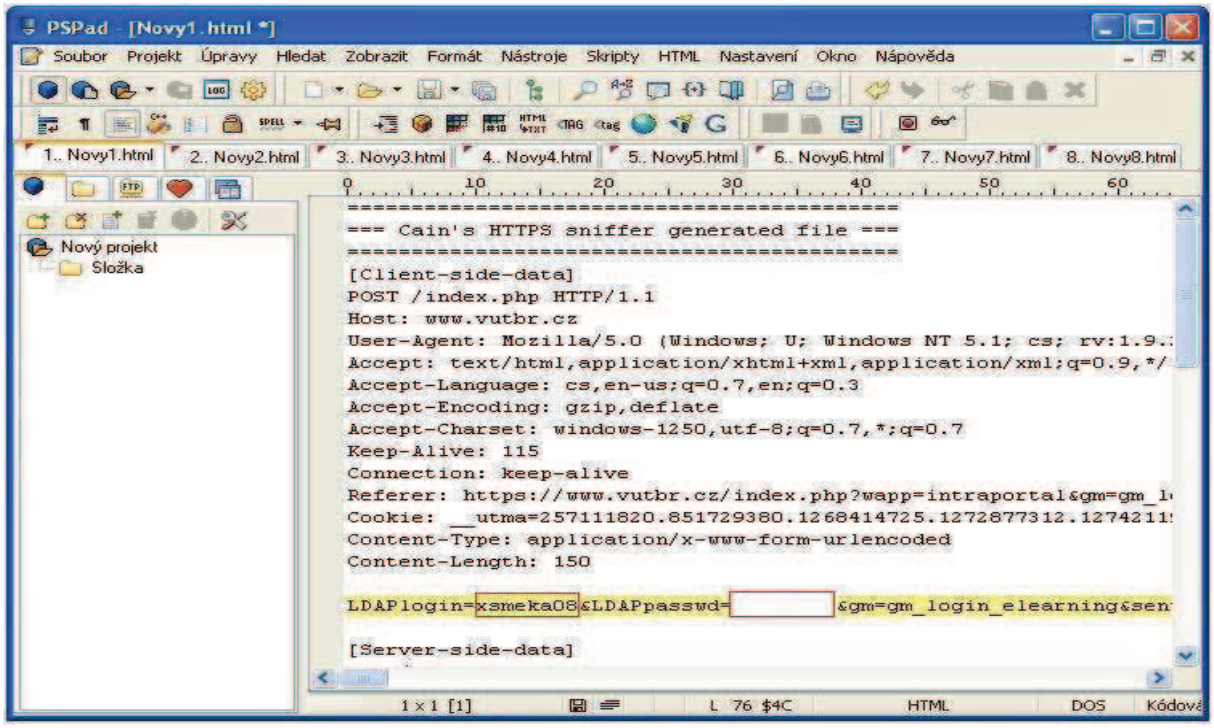


Obr. 29: Zachycený rozvrh hodin zobrazený programem PSPad



Obr. 30: Zachycená stránka z elearningu VUT zobrazená programem PSPad

Protože při vytváření informačního systému je využito kaskádových stylů a java skriptů, není formát zachyceného rozvrhu hodin totožný jako jej měl zobrazen uživatel. Dále jsem analyzoval jednotlivé soubory, abych se pokusil získat přihlašovací údaje k informačnímu systému. Podařilo se mi přihlašovací údaje najít, dokonce ve formě čistého textu (viz Obr. 31).



Obr. 31: Zachycené přihlašovací údaje k informačnímu systému VUT

6 Závěr

Při práci na diplomové práci jsem se postupně seznámil s problematikou zabezpečení počítačových sítí, zabezpečení osobního počítače a bezpečného používání počítače v síti Internet. Dále jsem se seznámil s různými druhy šifrování, s jejich funkcí, využitím a rozdíly. Podobně jsem prostudoval nejznámější šifrovací protokoly a algoritmy. Zaměřil jsem se na jejich výhody a nevýhody, kde se používají a při jakých příležitostech. Následně jsem se věnoval útokům na lokální počítačové sítě, jejich funkci, jejich vzájemné porovnání, jaké informace útočníci při nich mohou získat a jaké důsledky dané útoky mohou způsobit. Protože útoků je obrovské množství zaměřil jsem se pouze na ty nejznámější jako ARP Spoofing, DHCP Spoofing, DNS Spoofing, MAC Flooding, Port Stealing a ICMP Redirect.

Hlavní bodem této diplomové práce je program Cain & Abel, který slouží pro získávání hesel a autentizačních parametrů na základě odposlechu lokálních počítačových sítí. Obsahuje spoustu dalších nástrojů, například nástroje pro získání hesel uložených v paměti operačního systému, nástroje pro dešifrování hashů hesel, různé Windows utility nebo nástroje pro práci s databázemi.

Tato diplomová práce je zamýšlena jako takový návod na program Cain & Abel, který je vytvořen z mých poznatků a zkušeností s tímto programem a z praktický útoků na lokální počítačovou síť. Postupně jsem nastudoval instalaci tohoto programu, seznámil jsem se s grafickým rozhraním programu a konfigurací jeho nástrojů. Postupně jsem vyzkoušel funkci jednotlivých nástrojů a použil tyto nástroje pro útoky na jednotlivé šifrovací algoritmy.

Při práci s programem jsem zjistil i některé nedostatky programu. Při pokusech nainstalovat program Abel na vzdálený počítač, aby posílal informace z tohoto počítače zpět do programu Cain, jsem zjistil, že vzdálená instalace programu Abel, jak ji popisuje výrobce není možná, protože ji znemožní firewall na vzdáleném počítači. Možné je provést pouze instalaci lokální. Naopak je důležité pro bezproblémový příjem informací z programu Abel nebo zachycených přihlašovacích údajů během útoků, vypnout firewall nebo ho vhodně nastavit na počítači, kde je program Cain nainstalován.

Domnívám se, že se v tomto programu nachází mnoho nástrojů, které mi zde připadnou zbytečné, protože jsou obsaženy v příkazové řádce operačního systému. Jedná se například o nástroj Traceroute, tabulka TCP/UDP spojení, směrovací tabulka a podobně. Naopak mi zde chybí nástroj, který by vypočítal hash ze souboru, díky kterému by bylo možné ověřit integritu dat, čímž lze předejít například stažení dat z internetu s viry.

Po osvojení si práce s programem jsem provedl útok ARP Spoofing v domácí lokální síti, který je jediný útok z kapitoly 3, který tento program umí provést. Při tomto útoku jsem zjistil, jak je uživatel zranitelný při útoku „Man in the middle“, i když komunikuje například přes zabezpečený protokol (např. HTTPS).

Při práci s nástrojem programu Cain RSA SecureID Token Calculator jsem dostal k dispozici hardwarový token a synchronizoval jej s tímto nástrojem. Po synchronizaci hardwarového tokenu s nástrojem programu jsem byl schopen zjistit číselné přístupové kódy, aniž bych musel využívat hardwarového tokenu. Pomocí nástroje RSA SecureID Token

Calculator jsem byl schopen vytisknout si číselné přístupové kódy na zvolené časové období dopředu, což umožňuje se autentizovat v libovolný den i čas bez fyzického vlastnictví tokenu. Praktické využití nástroje RSA SecureID Token Calculator jsem měl demonstrovat s pomocí RSA serveru, který měl být k dispozici v předmětu Kryptografie v informatice, ale bohužel z důvodů odpadání výuky a kratšího semestru nebyl RSA server zařazen do výuky a nebyl ani fyzicky k dispozici. Danou problematiku jsem tedy popsal teoreticky a více se zaměřil i na ostatní nástroje programu.

SEZNAM LITERATURY A POUŽITÝCH ZDROJŮ

[1] THOMAS, Thomas M. *Zabezpečení počítačových sítí bez předchozích znalostí*. David Krásenský. Brno : CP Books,a.s., 2005. 338 s. ISBN 80-251-0417-6.

[2] [1] SCHNEIER, Bruce. Applied cryptography. 2nd edition. [s.l.] : John Wiley & Sons, 1996. 784 s. ISBN 0-471-11709-9 .

[3] JEŘÁBEK, J., *Pokročilé komunikační techniky.* Brno: VUT v Brně, 2009.

[4] BURDA, K. *Bezpečnost informačních systémů.* 1. Brno: FEKT VUT Brno, 2005. s. 1-104.

[5] MENEZES, Alfred J. ; VAN OORSCHOT, Paul C.; VANSTONE, Scott A. *Handbook of Applied Cryptography*. [s.l.] : CRC Press, 2008 . 816 s. Dostupné z WWW: <http://www.cacr.math.uwaterloo.ca/hac/>. ISBN 0-8493-8523-7.

[6] *Cisco Networking Academy : CCNA Exploration 4.0 Accessing the WAN* [online]. 2007 [cit. 2009-10-10]. Dostupný z WWW: <http://eviip.netacad.net/virtuoso/servlet/org.cli.delivery.rendering.servlet.CCServlet/LMS_ID=CNAMS,Theme=ccna3theme,Style=ccna3,Language=en,Version=1,RootID=knet-lcms_exploration4_en_40,Engine=static/CHAPID=null/RLOID=null/RIOID=null/index.htm>.

[7] *Cisco Networking Academy : CCNA Exploration 4.0* LAN Switching and Wireless [online]. 2007 [cit. 2009-10-10]. Dostupný z WWW: http://ev-iip.netacad.net/virtuoso/servlet/org.cli.delivery.rendering.servlet.CCServlet/LMS_ID=CNAMS,Theme=ccna3theme,Style=ccna3,Language=en,Version=1,RootID=knet-lcms_exploration3_en_40,Engine=static/CHAPID=null/RLOID=null/RIOID=null/index.html.

[8] RFC 3548. *The Base16, Base32, and Base64 Data Encodings*. [s.l.] : [s.n.], 2003. 13 s. Dostupné z WWW: <http://www.faqs.org/rfcs/rfc3548.html>.

[9] *RSA The Security Division of EMC* [online]. 2007 [cit. 2010-05-13]. RSA SecureID. Dostupné z WWW: <http://www.rsa.com/node.aspx?id=1156>.

[10] Moderní metody šifrování. *PCtuning* [online]. 2005 [cit. 2009-11-02]. Dostupný z WWW:<http://pctuning.tyden.cz/software/ochrana-soukromi/4711-moderni_metody_sifrovani>.

[11] Nejznámější útoky v síti Ethernet. *Connect!* [online]. 2007 [cit. 2009-11-12]. Dostupný z WWW: <http://connect.zive.cz/node/714>.

[12] HALER, Martin. Odposloucháváme data na přepínaném Ethernetu. *LUPA* [online]. 2006 [cit. 2009-10-15]. Dostupný z WWW: <http://www.lupa.cz/serialy/odposlouchavame-data-na-prepinanem-ethernetu/>.

[13] *Oxid.it* [online]. 2001-2009 [cit. 2009-11-20]. Dostupný z WWW: <http://www.oxid.it/cain.html>.

[14] CISCO [online]. 2008 [cit. 2010-02-26]. When Are ICMP Redirects Sent?. Dostupné z WWW: <http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094702.shtml>.

[15] *CISCO* [online]. 2008 [cit. 2010-04-18]. Zabezpečení přepínaných sítí. Dostupné z WWW: <http://www.cisco.cz/index.sub.php?pid=site&typ=sswitch>.

[16] *Svět sítí* [online]. 2000 [cit. 2010-04-17]. SecureID. Dostupné z WWW: <http://www.svetsiti.cz/print.asp?rubrika=Produkty&clanekID=11>.

[17] DRMOLA, Robert. *Základy kryptografie* [online]. 2007 [cit. 2010-05-01]. Bobhy. Dostupné z WWW: <http://bobhy.wz.cz/clanky/kryptografie.html>.

[18] *P. Hanáček. Bezpečnostní funkce v počítačových sítích. Zpravodaj ÚVT MU. ISSN 1212-0901, 1999, roč. X, č. 2, s. 5-9.*

[19] MILFAJT J. Bezpečnostní protokoly v praxi. Brno: Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, 2008. 56s.Vedoucí bakalářské práce Ing. Tomáš Pelka.

SEZNAM ZKRATEK

3DES	Triple **D**ata **E**ncryption **S**tandard
AAA	**A**uthentication, **A**uthorization, **A**coounting
ADPCM	**A**daptive **D**ifferential **P**ulse **C**ode **M**odulation
AES	**A**dvanced **E**ncryption **S**tandard
AP	**A**ccess **P**oint
APR	**A**RP **P**oison **R**outing
ARP	**A**ddress **R**esolution **P**rotocol
CCDU	**C**isco **C**onfig **D**ownloader/**U**ploader
CHAP	**C**hallenge **H**andshake **A**uthentication **P**rotocol
CLI	**C**ommand **L**ine **I**nterface
DCE/RPC	**D**istributed **C**omputing **E**nvironment / **R**emote **P**rocedure **C**alls
DES	**D**ata **E**ncryption **S**tandard
DHCP	**D**ynamic **H**ost **C**onfiguration **P**rotocol
DNS	**D**omain **N**ame **S**ystem
DoS	**D**enial **o**f **S**ervice
EIGRP	**E**nhanced **I**nterior **G**ateway **R**outing **P**rotocol
FTP	**F**ile **T**ransfer **P**rotocol
GRE/PPP	**G**eneric **R**oute **E**ncapsulation/ **P**oint-to-**P**oint **P**rotocol
HSRP	**H**ot **S**tandby **R**outer **P**rotocol
HTTP	**H**yper **T**ext **T**ransfer **P**rotocol
HTTPS	**H**yper **T**ext **T**ransfer **P**rotocol **S**ecure
ICMP	**I**nternet **C**ontrol **M**essage **P**rotocol
iLBC	**i**nternet **L**ow **B**itrate **C**odec
IMAP	**I**nternet **M**essage **A**ccess **P**rotocol
IP	**I**nternet **P**rotocol
LAN	**L**ocal **A**rea **N**etwork
LDAP	**L**ightweight **D**irectory **A**ccess **P**rotocol
MAC	**M**edia **A**ccess **C**ontrol
MD5	**M**essage **D**igest **5**
MIME	**M**ultipurpose **I**nternet **M**ail **E**xtensions
MITM	**M**an **I**n **T**he **M**iddle
NNTP	**N**etwork **N**ews **T**ransfer **P**rotocol
NTLM	**NT L**AN **M**anager
ODBC	**O**pen **D**ata**B**ase **C**onnectivity
PGP	**P**retty **G**ood **P**rivacy
PIX	**P**rivate **I**nternet e**X**change
POP3	**P**ost **O**ffice **P**rotocol **3**
PPPoE	**P**oint-to-**P**oint **P**rotocol **o**ver **E**thernet
RADIUS	**R**emote **A**uthentication **D**ial **I**n **U**ser **S**ervice
RC4	**R**ivest **C**ipher **4**
RDP	**R**emote **D**esktop **P**rotocol
RIP	**R**outing **I**nformation **P**rotocol
RSA	iniciály autorů **R**ivest, **S**hamir, **A**leman
RTP	**R**eal-time **T**ransport **P**rotocol
SHA	**S**ecure **H**ash **A**lgorithm
SIP	**S**ession **I**nitiation **P**rotocol
SMB	**S**erver **M**essage **B**lock
SMTP	**S**imple **M**ail **T**ransfer **P**rotocol

SNMP	**S**imple **N**etwork **M**anagement **P**rotocol
SP	**S**ervice **P**ack
SQL	**S**tructured **Q**uery **L**anguage
SSH	**S**ecure **SH**ell
SSID	**S**ervice **S**et **ID**entifier
SSL	**S**ecure **S**ockets **L**aye**r**
TACACS	**T**erminal **A**ccess **C**ontroller **A**ccess-**C**ontrol **S**ystem
TCP	**T**ransmission **C**ontrol **P**rotocol
TDS	**T**abular **D**ata **S**tream
TNS	**T**ransparent **N**etwork **S**ubstrate
UDP	**U**ser **D**atagram **P**rotocol
VNC	**V**irtual **N**etwork **C**omputing
VPN	**V**irtual **P**rivate **N**etwork
VoIP	**V**oice **o**ver **I**nternet **P**rotocol
VRRP	**V**irtual **R**outer **R**edundancy **P**rotocol
WEP	**W**ired **E**quivalent **P**rivacy
WPA	**W**i-Fi **P**rotected **A**ccess
XML	e**X**tensible **M**arkup **L**anguage