VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

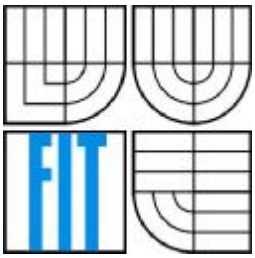
HONEY-POT: SYSTÉM PRO DETEKCI ÚTOKŮ

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE ZBYNĚK MICHLOVSKÝ
AUTHOR

BRNO 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

HONEY-POT: SYSTÉM PRO DETEKCI ÚTOKŮ

HONEY-POT: SYSTEM FOR ATTACK DETECTION

DIPLOMOVÁ PRÁCE
MASTER‘S THESIS

AUTOR PRÁCE — Zbyněk Michlovský
AUTHOR

VEDOUCÍ PRÁCE — Daniel Cvrček
SUPERVISOR

BRNO 2007

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav inteligentních systémů Akademický rok 2006/2007

Zadání diplomové práce

Řešitel: **Michlovský Zbyněk**
Obor: Výpočetní technika a informatika
Téma: **Honey-Pot: Systém pro detekci útoků**
Kategorie: Bezpečnost

Pokyny:

1. Nastudujte systémy pro detekci útoků - honey pot a honey net. Používejte doporučenou literaturu, a hodne samotný Internet.
2. Jednotlivé systémy nainstalujte a vyzkoušejte v praxi.
3. Vytvořte klasifikaci existujících systémů a popište jejich vlastnosti a schopnosti.
4. Vyberte jeden z testovaných systémů a spusťe ho na delší časový interval.
5. Proveďte podrobnou analýzu výsledků.

Literatura:

- L. Spitzner: Honeypots, Tracking Hackers, Addison-Wesley, 2003.
- Honeynet Project: Know Your Enemy: Revealing the Security Tools, Tactics, and Motives of the Blackhat Community, Addison-Wesley, 2001.

Při obhajobě semestrální části diplomového projektu je požadováno:

- Bez požadavků.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Cvrček Daniel, doc. Ing., Ph.D.**, UITS FIT VUT
Datum zadání: 1. listopadu 2006
Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
612 66 Brno, Božetěchova 2

L.S.

doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Zbyněk Michlovský**
Id studenta: 22634
Bytem: Revoluční 691/8, 691 45 Podivín
Narozen: 18. 08. 1982, Valtice
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Honey-Pot: Systém pro detekci útoků
Vedoucí/školitel VŠKP: Cvrček Daniel, doc. Ing., Ph.D.
Ústav: Ústav inteligentních systémů
Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:
tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
   - ☐ ihned po uzavření této smlouvy
   - ☒ 1 rok po uzavření této smlouvy
   - ☐ 3 roky po uzavření této smlouvy
   - ☐ 5 let po uzavření této smlouvy
   - ☐ 10 let po uzavření této smlouvy

   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: .....

.....	.....
Nabyvatel	Autor

Abstrakt

Tento dokument se zabývá problematikou honeypotu a honeynetu. Zaměřuje se na jejich rozdělení a podrobný popis jejich vlastností a schopností. Dále charakterizuje několik volně dostupných řešení. Především představuje honeypot Nepenthes, který byl provozován po dobu jednoho měsíce na nefiltrovaném připojení k internetu a následně přináší analýzu získaných dat.

Klíčová slova

Honeypot, honeynet, útočník, hacker, bezpečnost, Nepenthes, HoneyD, Honeywall, KFsensor, analýza dat.

Abstract

This thesis deals with the area of honeypots and honeynets. It defines their classification and contains detailed descriptions of their properties and features. It further elaborates on several freely available systems. The main focus is given to honeypot Nepethes that was being run for one month on an unfiltered Internet connection. A detailed analysis of the collected data is then given.

Keywords

Honeypot, honeynet, attacker, hacker, security, HoneyD, Honeywall, KFsensor, data analysis.

Citace

Zbyněk Michlovský: Honey-Pot, diplomová práce, Brno, FIT VUT v Brně, rok

Honey-pot: Systém pro detekci útoků

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Daniela Cvrčka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……
Zbyněk Michlovský
17. 5. 2007

Poděkování

Děkuji panu Danielu Cvrčkovi za poskytnuté odborné rady a doporuční. Dále děkuji firmě ICZ a.s. za poskytnuté prostředky k testování honeypotů.

Obsah

1 Úvod

S rozvojem internetu se zvyšuje potřeba stále komplexnější ochrany proti automatickým nástrojům a zejména živým útočníkům, kteří se snaží získat vládu nad našimi systémy. Tato práce se zaměřuje na speciální bezpečnostní prvek sítě – honeypot.

Honeypot je nástroj, jenž přímo nezvyšuje bezpečnost počítačové sítě, ale přináší možnost analýzy způsobu útoků na jednotlivé komponenty interní sítě. Jedná se o jednoúčelový systém, který nepřináší žádnou funkční výhodu, ale pomocí správné konfigurace a umístění v síti působí jako návnada a past na útočníky. Jeho nasazením v ostrém provozu získáme lepší přehled o komunitě útočníku a můžeme zjistit motivaci, proč se snaží o průniky do jednotlivých systémů. V neposlední řadě pomocí analýzy získaných dat snadněji porozumíme metodám útoků a následně s použitím získaných informací dokážeme lépe zabezpečit vlastní provozní systémy.

Cílem této práce je porozumět funkci a způsobu činnosti jednotlivých honeypotů, seznámit se s rozdělením podle jejich vlastností a následně je testovat v ostrém provozu. Zajímavou oblastí jsou bezesporu získaná data z honeypotu, který byl spuštěn na delší časové období.

První kapitola přináší charakterizaci útočníků, jejich důvody k útokům a možnosti průniků do cizích lokálních sítí. V další kapitole představujeme samotné honeypoty a jejich rozdělení podle různých kritérií. Následuje kapitola přibližující problematiku honeynetu. Šestá kapitola přináší přehled testovaných softwarových řešení honeypotů. Předposlední kapitola popisuje způsob nasazení testovaného honeypotu. Závěrečná kapitola se zaměřuje na analýzu získaných dat z testovaného systému.

2 Útoky a útočníci

Hlavním důvodem používání honeypotů je získávání informací o útočnících, způsobu provedení útoku, objevování bezpečnostních slabin produkčních systémů a na základě těchto informací zabezpečení ostrého provozu. Nejprve si musíme položit otázky:

- co útočníci chtějí, proč se snaží napadnout systém,
- jak pronikají do systému,
- jaké nástroje používají k průniku do systému.

V následujících kapitolách si jednotlivé "typy" útočníků představíme a pokusíme se přiblížit důvody jejich počínání.

2.1 Script kiddies

Tito útočníci mají většinou pouze průměrné znalosti o fungování jednotlivých systémů, počítačových sítí a protokolů. K útokům používají poloautomatické nástroje nebo exploity, které byly vytvořeny zkušenějšími hackery (většinou nemají o těchto nástrojích hlubší znalosti). Dívají se na hacking jako na nějaký druh sportu.

Exploit je krátký program, který má za úkol využít známé chyby v programu. Využije například přetečení zásobníku a přepíše návratovou adresu, která bude po našem útoku ukazovat na payload.

Payload je kód, kterému je předáno řízení po zdárné exploataci. Tyto programy plní různé úkoly. Mohou na cílovém stroji otevřít naslouchající spojení na určeném portu, který bude vracet shell, mohou zapříčinit download souboru ze sítě a jeho následné spuštění nebo založení nového uživatelského účtu.[21]

Jsou většinou schopni pronikat do systémů pouze významnými a dlouho známými bezpečnostními děrami. Nepoužívají žádné techniky pro maskování útoků. Jejich hlavní motivací je získat pod svou kontrolu co nejvíc systémů, co nejjednodušším způsobem. Pokud u jednoho neuspějí pokračují jiným => nemají zájem o konkrétní systém.

Script Kiddies nejsou velikou hrozbou, ale jsou nejběžnější skupinou útočníků a svým počtem převyšují ostatní skupiny. Na obranu proti nim většinou stačí mít aktualizovaný operační systém (dále jen OS) a používat vhodný firewall. Pokud se do systému dostanou, využijí ho k útokům na další systémy a většinou se nezajímají o uložená data.

2.2 Bot hackers

Jejich hlavním (a často jediným) cílem je vybudovat si na cizích systémech "armádu" botů. Pro každého bota potřebují samostatný systém. Mají lepší technické a systémové znalosti než script kiddies, ale stále potřebují nástroje (rootkity, exploity) vytvořené jinými lidmi.

Rootkit je program, který se snaží zamaskovat vlastní přítomnost (nebo jiných aplikaci) v sytému. Maskuje své vlastní soubory, logy, služby, ...

Nesnaží se prolomit dobře zabezpečený systém, svoji pozornost zaměřují na nejhůře zabezpečené systémy. Jejich cílem je získat "slávu", která se zvyšuje podle počtu funkčních botů (IRC_BOT) [9]. Snaží se co nejdéle zůstat na cizím systému v utajení. Tito útočníci se stávají hrozbou pouze tehdy, pokud jejich boti zahájí například DOS útok proti konkrétnímu systému.

2.3 Warez hackers

Warez hackers pracují ve skupinách, které se zaměřují na vyhledávání bezpečnostních děr a vyvíjení prostředků k jejich zneužití. Také jsou zodpovědné za "crackování" komerčního software a jeho šíření po internetu. K této činnosti potřebují velkou diskovou kapacitu, kde mohou ukládat výsledky své práce. Často využívají systémy do kterých dříve pronikly, což jim zajišťuje určitou míru utajení jejich skutečné identity a místa pobytu. Členové těchto skupin mají dobré technické znalosti. Jejich cílem je vytvářet nové toolkity, které využívají script kiddies a získat slávu pro sebe nebo pro svou skupinu. Hlavní hrozbu většinou nepředstavuje skupina samotná, ale nástroje, které tato skupina vytvořila.

2.4 Profesional Hackers

Nejmenší a nejnebezpečnější je skupina komunity hackerů, kterou tvoří útočníci s výbornými znalostmi systémů a počítačových sítí. Tito útočníci si na základě nově objevených (případně jimi nalezených) bezpečnostních děr nástroje sami vytvářejí. Cíleně napadají systémy s vysokou hodnotou. Jejich motivací jsou především peníze, ať už se jedná o vykrádání účtů, získávání dat na objednávku nebo prosté vydírání (např. na systému, který mají pod kontrolou, data nesmažou, ale "pouze" zašifrují, následně za dešifrování požadují přiměřené částky).

Své útoky na dané systémy provádí co nejhůře detekovatelné (případně hlavní útok maskují jiným útokem). Pokud proniknou do cizího systému, provádí v něm pouze minimální změny. Ve většině případů využijí napadený systém k dalším útokům.

2.5 Shrnutí

	Script Kiddies	Bot Hackers	Warez Hackers	Prof. Hackers
Technické znalosti	nízké	nízké - střední	střední	vysoké
Vlastní nástroje	-	-	+	+
Detekovatelnost	+	+	+/-	-
Zájem o cizí data	-	-	+	+
Hacking pro slávu	+	+	+	-
Hacking pro peníze	-	-	+/-	+
Vandalismus	+	+	-	-

Tab. 1: *Srovnání útočníků*

V této kapitole jsme rozdělili útočníky podle technických znalostí a především důvodů, proč napadají cizí systémy. *Tabulka 1* poskytuje názorný přehled.

3 Honey-pot

Honey-pot (nebo honeypot) je obecné označení zdroje, jehož smysl spočívá v jeho neautorizovaném využití. Je to úmyslně nastrčená nástraha, která se snaží o aktivní komunikaci s útočníkem. Může se jednat o pracovní stanici, poštovní server, tiskárnu, router, počítačovou sít (honey-net), nebo cokoliv - co dokáže emulovat určité zařízení. Sledování této bezpečnostní nástrahy nám umožní analyzovat bezpečnostní incidenty a na základě získaných informací se přiměřeně bránit.

Honeypot je “pouze“ bezpečnostním nástrojem, který uživateli nepřináší žádnou funkční výhodu, ale z hlediska bezpečnosti a hodnoty získaných informací jsou možnosti jeho využití obrovské.

3.1 Důvody použití Honeypotů

Následující kapitoly popisují hlavní důvody, použití honeypotů.

3.1.1 Low false-positivies

Jedním z hlavních důvodů použití honeypotů je nízké číslo *false-positives* a *false-negatives* záznamů.

Přívlastkem false-positives označujeme záznam v logu, který popisuje stav, kdy bezpečnostní nástroj detekuje bezpečnou aktivitu v systému jako nebezpečnou

Přívlastkem false-negatives označujeme záznam v logu popisující nebezpečnou aktivitu kterou bezpečnostní nástroj nebyl schopen detekovat.

Bezpečnostní logy většiny IDS (Intrusion Detect System) obsahují velké množství false-positives záznamů, proto je velice obtížné v nich najít údaje, které upozorňují na nebezpečné aktivity v sytému.

Honeypoty, slouží v systému pouze jako vábnička, proto veškerou komunikaci (snad kromě systémové a administrátorské), která přichází nebo odchází z honeypotu, můžeme považovat za komunikaci nebezpečnou. To znamená, že v logu, který nám nabídne honeypot, se nachází převážně záznamy o komunikaci s útočníkem – tzn. honeypoty mají nízké false-positives.

3.1.2 Včasná detekce

Nízký výskyt false-positives záznamů vede k rychlému odhalení legitimních hrozeb. Proto někteří administrátoři používají honeypoty – resp. honeytokeny jako mechanismy včasného varování.

Honeytokeny můžou být umístěny na honeypotech nebo na provozních serverech. Honeytoken může být nečinná mystifikace například uživatelský účet nazvaný Administrátor, bez jakýchkoliv práv, kdy administrátorský účet (Windows) je přejmenovaný na účet s neutrálním jménem. Pokud se někdo pokusí přistoupit na účet Administrátor je vygenerována výstraha a odeslána správci. Tyto

mechanismy umožňují správcům systémů v krátkém časovém intervalu reagovat na průniky do provozních sítí.

3.1.3 Zjišťování nových hrozeb

Jak už bylo dříve zmíněno, každý přístup na honeypot (kromě administrátorského) lze považovat za pokus o průnik do systému. Pomocí vhodných sledovacích prostředků a databází známých způsobů průniků (vektorů průniků), je možné detekovat nové metody, jakými se útočníci pokouší nabourat do provozních systémů.

3.1.4 Prevence útoků

Honeypoty obvykle nepoužíváme, kvůli jejich schopnosti bránit útočníkům k přístupu do našich systémů. Především je považujeme za pasivní zapisovací/nahrávací zařízení. Na rozdíl od firewallů nebo IDS honeypot dokáže pouze okrajově předcházet hackingu. Za prevenci útoků můžeme považovat případ, kdy hacker tráví čas útokem na honeypot a tím odvrací pozornost od reálných produkčních cílů – případně tímto útokem dává čas pověřené osobě na zabezpečení děr v provozním sytému, pomocí kterých útočník pronikl do honeypotu.

3.1.5 Honeypot jako soudní nástroj

Záznamy o komunikaci pocházející z honeypotů mohou sloužit jako důkaz v případném soudním sporu s útočníky, kteří se pokoušeli nabourat do našeho systému. Naneštěstí každý hacker může tvrdit, že “nevěděl, co dělá“ a jeho “kompromitovaný“ počítač sloužil právě jako nástroj pro útočníky.

3.2 Nevýhody při použití honeypotů

Hlavním rizikem při použití honeypotu je možnost, že útočník při útoku na honeypot uspěje a podaří se mu ho zmocnit (k tomuto problému se vrátíme v dalších kapitolách). Pokud s tímto případem není počítáno, otvírá se útočníkovi brána do sítě s provozními systémy.

Další nevýhodou je, že honeypot zachytí pouze útok, který směřuje přímo na něj. Pokud se útočník snaží zasáhnout jiné systémy, honeypot nemá šancí útok detekovat.

3.3 Základní součásti honeypotů

Pros správnou funkci honeypotu, je nutné, aby obsahoval několik prvků, pomocí kterých je možné sledovat a obsluhovat jeho činnost.

Síťová zařízení: síťový hardware.

Logovací nástroje: umožňují monitorovat co útočník na honeypotu provádí. Každý honeypot musí mít monitorovací a logovací nástroje, pomocí kterých předává informace logovací stanici.

Logovací stanice: monitorovací a logovací stanice sbírá data z honeypotů v síti. Je nutné ji mít velmi dobře zabezpečenou proti útokům.

Výstražné nástroje: při průniku do honeypotu, umožní poslat varovné hlášení správci systému.

Keystroke logger: je nutný k získání psaných příkazů útočníka, který pronikl do honeypotu.

Analyzátor paketů: je nezbytný k získání informací, které si předává honeypot s okolním světem.

Místo pro zálohy: je vhodné pro zálohování změn, které provedl útočník na honeypotu a k obnovení honeypotu do původního stavu.

3.4 Dělení honeypotů

3.4.1 Dělení podle způsobu emulace

Emulaci je možné provádět v zásadě dvěma způsoby. První z nich vychází ze znalostí chování dané služby z vnějšího pohledu, tj. jak se prezentuje v normálním provozu, jak reaguje na jednotlivé výzvy, nebo jaké má chybové hlášky. Druhý způsob používá skutečný program, který provozujeme pod přísnou kontrolou a v omezeném prostředí:

3.4.1.1 Honeypoty s emulovanými programy

Jsou založeny na znalostech reakcí jednotlivých emulovaných služeb na různé výzvy. Hlavní výhodou je jednoduchost a robustnost - žádná chyba v emulovaných službách nemůže honeypot ohrozit. Díky tomu přináší jejich použití minimální riziko a značný zisk. Nevýhodou je nižší úroveň maskování honeypotu, která je omezena kvalitou emulace služby. Vzhledem k tomu, že značná část emulovaných služeb je vázána na OS, emulují honeypoty služby i s ohledem na OS. Pokročilejší druhy honeypotů emulují též vrstvu IP protokolů, čímž je možné oklamat i sofistikované nástroje typu nmap, které používají odlišnosti implementací TCP/IP k zjištění typu OS.

3.4.1.2 Honeypoty s reálnými programy

Důvodem, proč se nepoužívají pouze honeypoty s emulovanými programy, jsou jejich omezení v zaznamenávaní informací o nových druzích útoků. Honeypoty s reálnými programy je nutné provozovat ve vysoce kontrolovaném prostředí a mít je pod důkladným dohledem, např. pomocí speciálních modulů v jádrech operačních systémů. Útočníkům je umožněno bez omezení posílat data na honeypot, ale odchozí komunikace je omezena, aby nebylo možné zneužít honeypoty k útokům na ostatní systémy.

3.5 Dělení podle způsobu použití

Rozdělení vychází z typů metod ochrany před útoky, ve kterých honeypoty našly uplatnění.

3.5.1 Provozní honeypoty

Jejich hlavním úkolem je snížit rizika útoků na provozní systémy s vysokým ohledem na samotnou bezpečnost honeypotu a okolních ostrých systémů. Správci se snaží tyto honeypotu nakonfigurovat co nejpodobněji reálným systémům včetně služeb a záplat. Od provozních honeypotů se očekává, že zvládnou aktivity popsané v následujících třech odstavcích.

3.5.1.1 Prevence

Spočívá v preventivní ochraně zejména před automatizovanými útoky, které se pokouší na Internetu najít servery se známými chybami a následně se jich zmocnit. Přítomnost honeypotů působí preventivně i proti lidským útočníkům. Usnadňuje získání informací o formě útoku i o samotném útočníkovi.

3.5.1.2 Detekce

Chrání LAN včasným zjištěním pokusu o útok, čímž dává administrátorům čas na přípravu a realizaci opatření proti útoku na provozní systémy.

3.5.1.3 Reakce

Reakce na útok může následovat až po zjištění, kdo a co napáchal. Tyto informace lze jen těžko získat na ostrých systémech, které často není možné jednoduše odstavit a podrobit důkladné analýze. Honeypoty podobná omezení nemají.

3.5.2 Výzkumné honeypoty

Tyto honeypoty slouží k získávání informací o nových metodách, nástrojích útočníků a o samotném chování hackerů v cizích systémech, případně k zjištění bezpečnostních děr v reálných programech. Na základě těchto informací jsou vyvíjeny nové obrané prostředky. Většinou se jedná o honeypoty s vyšší mírou interakce a s nízkou mírou ”zabezpečení” simulovaného systému.

Pokud se jedná o honeypot s reálnými službami je nutné dbát na zabezpečení honeypotu, aby nedošlo k převzetí kontroly honeypotu útočníkem a jeho využití k další nekalé činnosti.

3.6 Dělení podle míry interakce

Dělením podle míry interakce rozumíme rozsah, jakým mohou vzájemně působit operační systém a útočník.

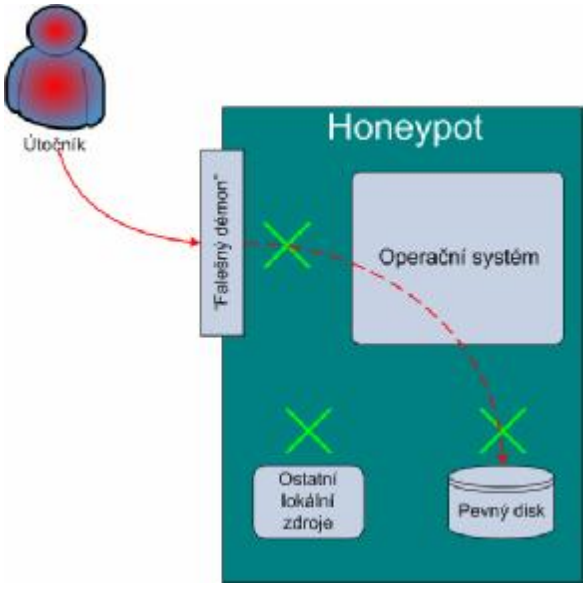
3.6.1 Honeypoty s nízkou úrovní interakce

Honeypoty s nízkou úrovní interakce mají velmi jednoduchou instalaci, konfiguraci a nepotřebují zásadní údržbu. Honeypot se skládá pouze z démonů předstírajících funkci některých běžných služeb (HTTP, FTP, ...). Tito démoni neumožňují útočníkovi operovat s reálným operačním systémem. To minimalizuje bezpečnostní rizika při používání těchto honeypotů. Na druhou stranu není možné zjistit, jakým způsobem by útočník ovlivňoval operační systém.

Honeypoty s nízkou úrovní interakce si můžeme představit jako jednosměrné spojení, tzn.: démoni honeypotu pouze naslouchají, ale sami dotazy neposílají. Jejich role je velmi pasivní a nedokáží objevit neznáme způsoby útoků. Používají se především k získávání těchto informací:

- čas a datum útoku,
- zdrojová ip adresa a port útočníka,
- cílový port a ip adresa útoku.

Vše s minimálním rizikem ovládnutí honeypotu, případně proniknutí do systému.



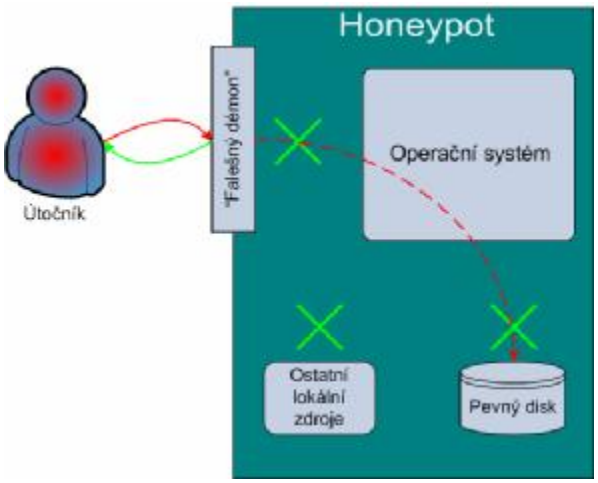
Obr. 1: *Honeypot s nízkou úrovní interakce pouze naslouchá a ukládá získané informace*

3.6.2 Honeypoty se střední úrovní interakce

Tyto honeypoty umožňují útočníkovi vyšší míru interakce. “Falešní“ démoni jednotlivých služeb jsou již více sofistikovanější a poskytují útočníkovi větší míru součinnosti. Tzn. útočník dostává lepší iluzi skutečného operačního systému a má větší možnosti ovlivňovat a zkoumat systém.

S vyšší mírou interakce mohou být logovány a analyzovány komplexnější útoky. Nevýhodou je zvýšení rizika průniku při použití těchto honeypotů. Pravděpodobnost, že útočník najde bezpečnostní díru nebo jiné zranitelné místo, je vyšší stejně jako složitost honeypotu.

Příprava těchto honeypotů je složitá a zdlouhavá. Zvláštní péče musí být věnována bezpečnosti jak z pohledu celého honeypotu, tak z pohledu jednotlivých simulovaných služeb.



Obr. 2: *Honeypot se střední úrovní interakce – démon odpovídá na dotazy*

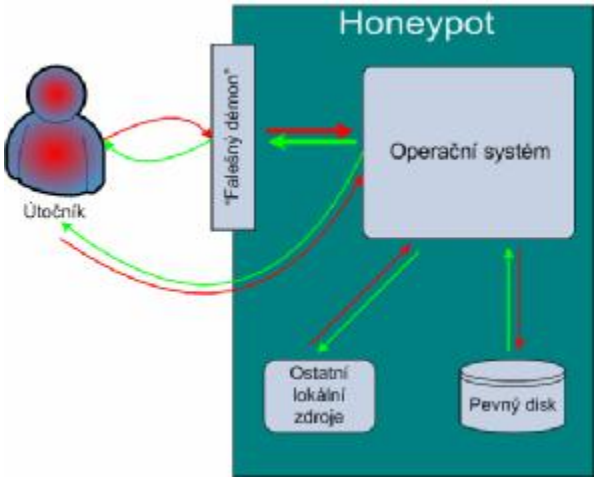
3.6.3 Honeypoty s vysokou úrovní interakce

Jedná se o nejsložitější honeypoty s největší mírou interakce s útočníkem a tím pádem s možností získat nejzajímavější informace. Útočník může s honeypotem nakládat jako s reálným systém – může pracovat s adresářovou strukturou, nahrávat a instalovat nové programy, pracovat s konzolou, ... Všechny tyto činnosti je honeypot schopen zachytit k pozdější analýze.

S mírou interakce roste i míra rizika, tyto honeypoty jsou již postaveny na reálných operačních systémech – to v nejhorším případě umožní útočníkovi využít honeypot k další nekalé činnosti. Honeypot je nutné “oddělit“ od zbytku sítě (ať už umístěním, firewallem nebo jinými omezením).

Tyto honeypoty jsou nejnáročnější k instalaci. Systém musí být pod neustálou kontrolou (například pomocí SMS zpráv). Pokud by nebyl, stal by se příliš velikým bezpečnostním rizikem

v naší síti a jejím okolí. Je velmi důležité omezit honeypotu, přístup k lokálnímu intranetu a především provozním systémům.



Obr. 3: *Honeypot s vysokou úrovní interakce – útočník je schopen pracovat se všemi zdroji honeypotu*

3.6.4 Souhrn

Každá z úrovní interakce honeypotu má své výhody a nevýhody. Následující tabulka je sumarizuje.

	Nízká úroveň	Střední úroveň	Vysoká úroveň
Reálný OS	-	-	+
Riziko	nízké	střední	vysoké
Sbíraná data	konexe	dotazy	vše
Znalosti k vytvoření	nízké	střední	vysoké
Znalosti k provozování	nízké	nízké	střední
Náročnost údržby	nízká	nízká	velmi vysoká

Tab. 2: *Přehled úrovní interakce honeypotu a útočníka*

Pokud použijeme honeypot s nízkou úrovní interakce podstupujeme nejmenší riziko, ale zároveň získáme data s nízkou hodnotou. Naopak pokud použijeme honeypot s vysokou úrovní interakce, riziko je vysoké, ale data, která získáme, nám umožní hlouběji proniknout do myšlení útočníků a studovat způsoby průniků do provozních systémů. Obrázky 1-3 názorně ukazují interakci honeypotu s útočníkem. Červená křivka představuje data/příkazy od útočníka (pod přerušovanou linkou si můžeme představit prozatímní logování komunikace útočníka na honeypotu), zelená čára reprezentuje odpovědi honeypotu a zelený křížek určuje maximální dosah útočníka.

3.7 Umístění honeypotů

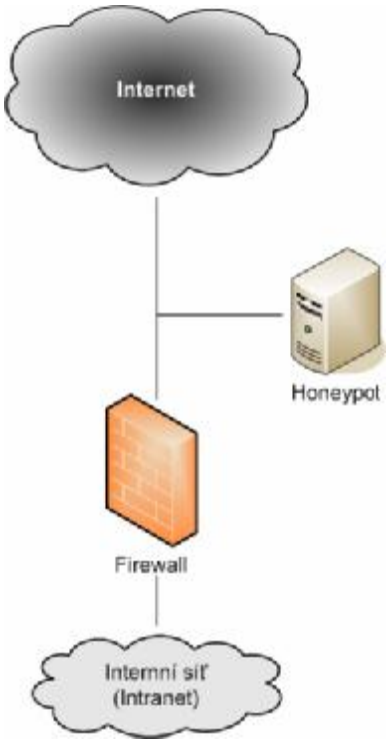
V této kapitole si ukážeme výhody a nevýhody honeypotů v závislosti na jejich umístění v síti. Honeypot jako kterýkoliv jiný server, může být použit kdekoliv v síti, ovšem jeho umístění je významné pro jeho využití.

Honeypot můžeme umístit do těchto tří hlavních lokací:

- před firewallem (obrázek 4),
- za firewallem (v intranetu – obrázek 6),
- v DMZ (obrázek 5).

3.7.1 Honeypot umístěný před firewallem

Honeypot umístěný před firewallem se nejvíce hodí pro výzkumné účely. Je cílem množství útoků a významným způsobem nesnižuje zabezpečení produkčního systému umístěného za firewallem. Je schopen včas detekovat nové hrozby v internetu a umožňuje správcům systémů se na ně připravit. Na druhou stranu, nemá žádnou vazbu na produkční síť a tím pádem není schopen odhalit útočníka, který už pronikl do vnitřní sítě.

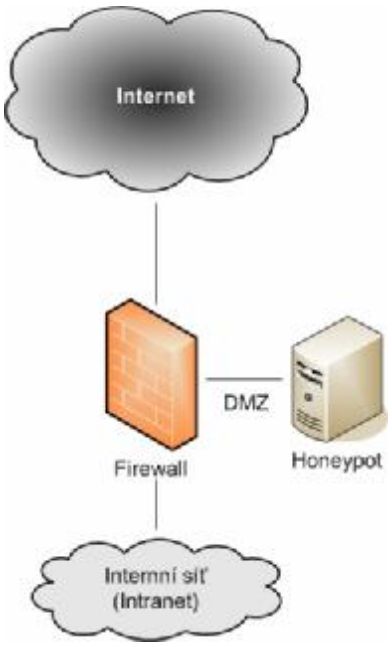


Obr. 4: *Honeypot umístěný před firewallem*

Velkou nevýhodou, v případě špatného nastavení honeypotu, je zachytávání obrovského množství (nechtěného) provozu na síti a tím pádem horší orientace v takovém objemu zachycených dat. Nehledě na to, že pokud útočník získá nad honeypotem kontrolu, stane se nástrojem páchání další nekalé činnosti.

3.7.2 Honeypot umístěný v DMZ

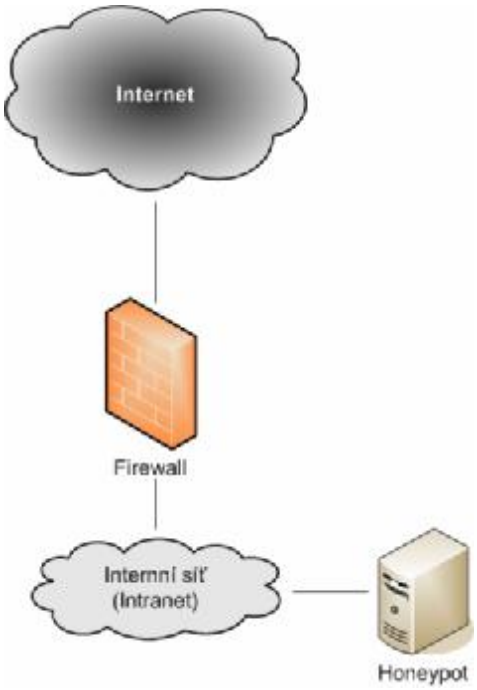
Umístěním honeypotu v DMZ získáme výborný nástroj ke klamání a odrazení útočníků. Donutíme je ztrácet čas a zdroje pronikáním do honeypotu, místo do produkčních systémů. Navíc správně nakonfigurovaný honeypot v DMZ dokáže okamžitě upozornit správce systému na průnik do sítě a tím získat čas na zabezpečení provozních systémů.



Obr. 5: *Honeypot umístěný v DMZ*

3.7.3 Honeypot umístěný za firewallem

Takto umístěný honeypot slouží jako pojistka především proti útočníkům z vnitřní sítě, případně je schopen odhalit chyby v nastavení firewallu. Na druhou stranu představuje bezpečnostní riziko pro vnitřní sít, obzvlášť pokud interní síť není zabezpečena proti útoku z honeypotu dodatečným firewallem.



Obr. 6: *Honeypot umístěný za firewallem*

3.7.4 Shrnutí

Následující tabulky ukazují vhodnost jednotlivých umístění honeypotů v síti a jejich hlavní výhody a nevýhody.

	Vhodnost umístění honeypotu před FW	Vhodnost umístění honeypotu v DMZ	Vhodnost umístění honeypotu za FW
Honeypot určený pro výzkum	vysoká	dobrá	žádná
Honeypot určený pro prevenci	nízká	vysoká	nízká
Honeypot určený pro detekci	nízká	vysoká	vysoká

Tab. 3: *Vhodnost umístění jednotlivých honeypotů*

	Výhody	Nevýhody
Honeypot umístěný před FW	Detekce nových hrozeb Jednoduchost nastavení Malé množství potřebného HW	Zvýšení rizika pro ostré systémy Horší orientace v získaných datech
Honeypot umístěný za FW	Výborný pro napodobení vnitřních systémů Varovný systém průniku do provozní sítě Monitoring interních zaměstnanců	Složitost nastavení Problematická datová kontrola
Honeypot umístěný v DMZ	Výborný pro napodobení vnitřních systémů Systém včasného varování při pokusu o průnik do provozní sítě	Složitost nastavení

Tab. 4: *Výhody a nevýhody umístění honepotů*

4 Honeynet

Honeynetem rozumíme dva nebo více honeypotů s vysokou úrovní interakce, umístěných v jedné síti a navržených k získávání dat o průnicích do provozních systémů. Někdy se také uvádí, že honeynet je typ honeypotu. Obě tvrzení jsou pravdivá.

Můžeme říct, že jde o vysoce kontrolovanou síť (část sítě), kde monitorujeme veškerou aktivitu. Jakoukoliv komunikaci, kromě servisní, považujeme za nežádoucí průnik do provozního systému.

[3] přirovnává honeynet k akváriu – vytvoříme prostředí (stanice, tiskárny, DNS servery, routery, ...) a sledujeme všechno, co se uvnitř stane. Podobně jak na sebe působí rybka a elementy akvária, tak na sebe působí útočník a honeynet.

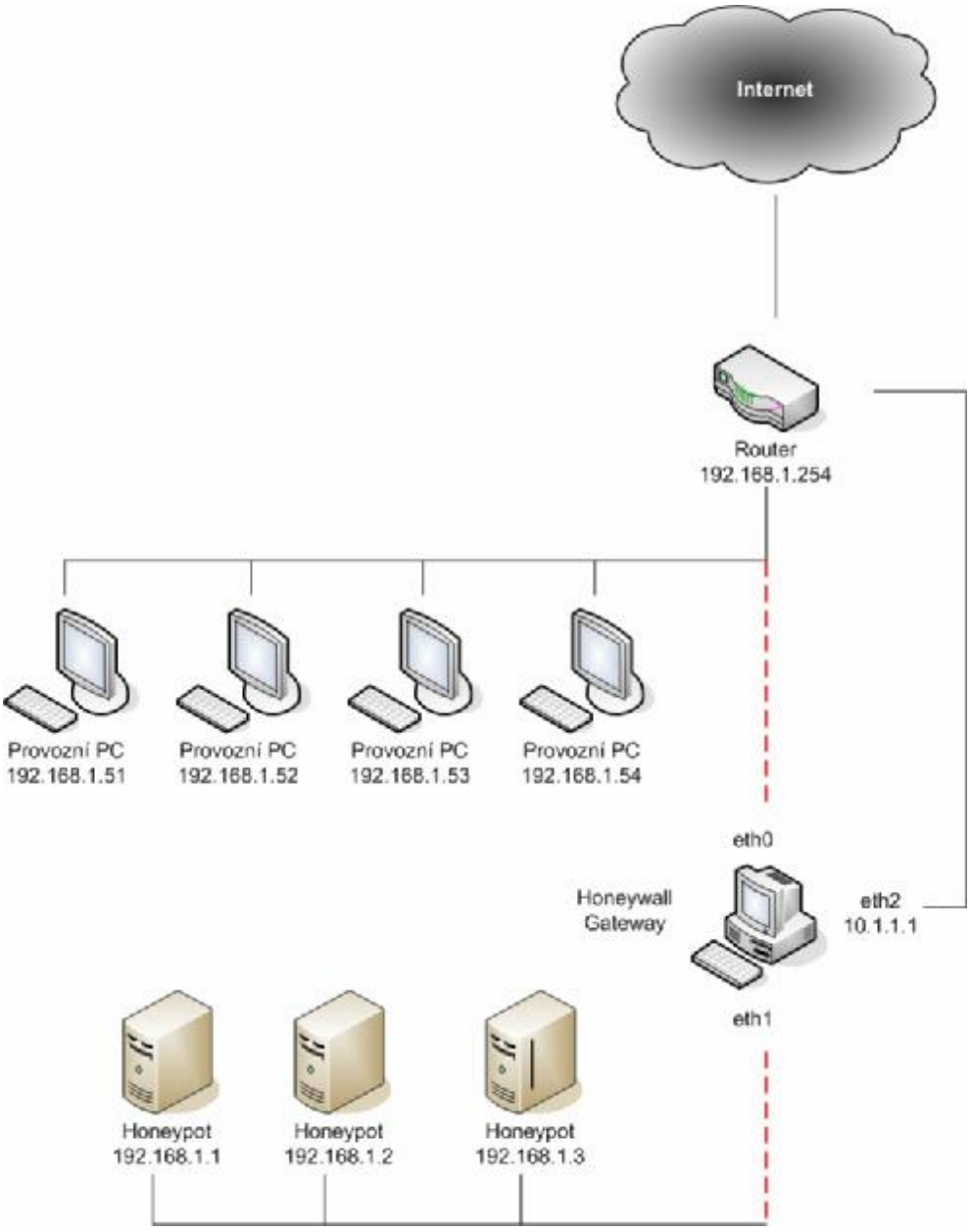
4.1 Architektura honeynetu

K úspěšnému nasazení honeynetu je nutné správně navrhnout architekturu honeypotu. Klíčem této architektury je honeywall. Jedná se o bránu, která odděluje honeypot od zbytku sítě. Jakýkoliv provoz přicházející nebo odcházející z honeypotů musí projít přes honeywall. Toto zařízení pracuje na druhé síťové vrstvě, proto je neviditelné pro každé zařízení komunikující s honeypoty.

Na obrázku č. 7 můžeme vidět ukázku architektury honeynetu. Honeywall má tři síťové karty. První dvě (eth0 a eth1) oddělují honeypot od okolní sítě a třetí síťová karta (eth2) umožňuje vzdálenou správu.

Pokud chceme provozovat honeynet, je zde několik klíčových položek, které v něm musíme implementovat:

- správu dat,
- sběr dat,
- analýza dat,
- shromažďování dat.



Obr. 7: *Ukázka architektury Honeynetu*

Správa dat

Správou dat rozumíme aktivitu, která zmírňuje rizika používaní honeynetu. Rizikem míníme například stav, kdy útočník spustí na honeynetu kód, který mu umožní ohrozit jiné než honeynet systémy, případně zneužije honeynet neočekávaným způsobem. Snažíme se v maximální míře zajistit, aby útočník, který se připojí k honeynetu, nebyl schopen náhodně nebo záměrně ublížit ostatním ne-honeynet systémům. Výzvou je implementovat správu dat tak, aby ji útočník nebyl schopen detekovat.

Útočníkovi musíme dát určitý stupeň volnosti v jednání na honeynetu (s rostoucím stupněm volnosti roste i hodnota informací, které můžeme ze sledování útočníka získat). Na druhou stranu s většími možnostmi útočníka roste riziko "prolomení" našeho honeynetu.

Při implementaci správy dat, není dobré spoléhat pouze na jeden bezpečnostní mechanismus. S výhodou můžeme použít kombinace bezpečnostních mechanismů např. omezení odchozích spojení, omezení množství odchozích dat a použití bran. Bráníme se tak proti jednotlivým selháním, které hrozí převážně u nových způsobů útoků. Je důležité také připravit bezpečnostní pojistku, která při daném selhání (plný disk, zabití procesu, ...) zablokuje veškerý odchozí tok dat.

Sběr dat

Sběr dat znamená monitorování a logování všech aktivit uvnitř honeynetu. Z těchto dat se pak dozvídáme o motivech, taktikách a nástrojích útočníků. Problémem j, zachytit maximální množství dat (i přes šifrované kanály – SSH, SSL, ...) bez toho, aby útočník odhalil naše monitorovací mechanismy. Sběr dat musí být prováděn přes více vrstev a samotná data by se neměla ukládat lokálně, ale na zabezpečený systém. Tím je uchráníme před případným smazáním nebo modifikací útočníkem.

Analýza dat

Hlavním cílem honeynetu je získávání informací. Implementace honeynetu bez schopnosti přeměnit získaná data na informace by byla zbytečná. Proto je nutné mít nástroj, který nám ze získaná data přemění na informace, které právě potřebujeme.

Shromažďování dat

Organizace, které využívají větší množství honeynetů (v distribuovaných prostředích rozmístěných po celém světě), střádají získaná data v centrální databázi. Tímto způsobem je možné data spojovat a kombinovat a zvýšit tím exponenciálně jejich informační hodnotu.

Implementace těchto klíčových položek honeynetu je extrémně obtížná, komplexní a časově náročná. Uvedené nevýhody částečně řeší Honeynet Project [4], kde lze najít množství odladěných nástrojů, které nasazení Honeynetu usnadní.

4.2 Rizika a problémy

Honeynet může být velmi silný nástroj, umožňuje sbírat rozsáhlé množství informací o různorodých hrozbách v internetu. K získání těchto informací, musíme útočníkovi umožnit potencionálně privilegovaný přístup do našeho honeynetu. Cenou za získání dat z honeynetu je riziko, že se útočníkovi podaří honeypot narušit, detekovat, vyřadit nebo ovládnout.

Narušení

Narušením honeynetu míníme stav, kdy je honeynet použit bez našeho "souhlasu" k činnostem, pro které nebyl projektován. Např. útočník se dostane do honeynetu, kde spustí kód, který zahájí útok, proti jiným ne-honeynet systémům.

Právě těmto případům by měla zabránit "správa dat" implementovaná v honeynetu.

Detekce

Rizika vyplývající z odhalení honeynetu jsou vyšší než se zdá. V lepším případě budou útočníci honeynet ignorovat, v horším případě útočník "podstrčí" honeynetu falešné informace, které znemožní správnou analýzu dat.

Vyřazení

Vyřazení honeynetu z činnosti může nastat po útoku na procedury, které zajišťují správa a sběr dat. Pokud se tak stane bez vědomí administrátora, nastává problém, útočník se dostane k honeypotu bez honeynetu. Jedinou obranou je implementovat procedury sběru a správy dat přes více vrstev.

Ovládnutí

Největším rizikem je ovládnutí honeynetu útočníkem. Útočník pak může pomocí našeho honeynetu vyvíjet trestnou činnost aniž by útočil na jiné systémy. Stačí aby využil honeynet, jako datový sklad ilegálních kopií filmů, hudby, SW, ... Pokud je tato činnost útočníka objevena bezpečnostními složkami, bude připsána (alespoň z počátku) nám jako vlastníkovi systému. My pak musíme doložit, že honeynet byl ovládnut a využit bez našeho vědomí. Právní aspekty využívání honeypotů a honeynetu jsou shrnuty v [8].

Zmírnění rizik

Mezi nejúčinnější prostředky pro zmírnění rizik plynoucí z používání honeynetu patří možnost jeho modifikace a monitorování živým člověkem. Pokud honeypot monitoruje člověk, dává nám to schopnost detekovat chyby systému, které by automatický nástroj neodhalil. Případně okamžitě reagovat na nové typy útoků.

Většina honeypotů je vydávána jako OpenSource. To má za následek, že je dostupná i útočníkům. Ti potom mohou, na základě informací ze studia honeypotů/honeynetů, dané systémy detekovat a vyvinout proti nim prostředky "obrany". Velmi účinnou obranou je konfigurace a modifikace zdrojového kódu honeynet-systémů. Čím víc se bude náš honeynet lišit od standartu, tím hůře bude zjistitelný a napadnutelný.

4.3 Shrnutí

Honeynet je forma honeypotu s vysokou mírou interakce. Jeho primární výhodou je schopnost, získat obsáhlé množství informací o chování, znalostech a možnostech útočníků. Architektura honeypotu je někdy přirovnávaná k akváriu – můžeme zde nasadit jakýkoliv systém nebo aplikaci, a pak čekat co útočníci udělají. Klíčové funkce honeynetu jsou: správa, sběr, analýza a shromažďování dat. Naneštěstí honeynet představuje i velké bezpečnostní riziko. Existují mechanismy, které riziko snižují, ale neexistuje způsob jak tato rizika úplně odstranit. Více se o honeynetu můžete dozvědět zde [4].

5 Detekce honeypotů

Existuje mnoho způsobů jak odlišit honeypot od běžného systému v síti. Většina způsobů využívá mezery v možnostech honeypotů – jako neschopnost korektně odpovědět na daný požadavek.

5.1 Detekce založená na ICMP

Jeden z jednodušších způsobů je možné najít na [5]. Postup je následující: předpokládejme, že server odpovídá na obyčejný příkaz Ping. Potom využijeme program Hping [6], jenž jednoduše posílá datapakety serveru, které obsahují shellkód, a porovnává odcházející ICMP pakety s pakety, které vrací server (např. pomocí tcpdump, etheral ...). Jestliže server pošle jako odpověď ping, který neobsahuje shellkód, nebo ho změní, pak se nejedná o provozní server, ale o sledovaný server.

Hping je nástroj, který umožňuje generovat a posílat libovolné ICMP/UDP/TCP pakety a zobrazovat odpovědi protistrany. Může sloužit také k přenosu souborů, testování konfigurace firewallů, skenování portů, testování prostupnosti sítě, detekci vzdáleného OS apod.

5.2 Detekce založená na monitoringu hlášení

Další možnost odhalování honeypot nástrah byla publikována na konferenci Usenix [7]. Pomocí optimalizovaného algoritmu se testované stroje pokoušely spojit s vybranými porty na velkém počtu IP adres. Následně sledovaly veřejně publikovaná hlášení a podle výsledků odhadovaly, kde jsou monitorované servery umístěny. Při praktickém testu odhalily nástrahy Internet Storm Center [13] během jediného týdne. Možné řešení podle výzkumníků spočívá v omezení poskytovaných informací o útocích. Nejúčinnější by bylo zajistit, že hlášení budou dostupná jen pro důvěryhodné osoby, ale tím by podobné služby ztratily smysl. Proto výzkumníci navrhují několik dalších opatření, od publikování pouze "top listů" po omezení dotazů na výsledky. Plné znění v angličtině je možné nalézt zde [7].

6 Jednotlivé honeypot - systémy

Systémy byly nejprve instalovány na OS Arch Linux pod VMware (na Windows XP). Po prověření nastavení a funkčnosti programu byly přeneseny na "ostrý" stroj s nefiltrovanou IP adresou.

Parametry stroje, na kterém byly systémy testovány:

i686 Intel(R) Pentium(R) 4 CPU 2.40GHz,

Mem: 524288 kB,

Intel Corporation 82562EZ 10/100 Ethernet Controller (rev 02),

Linux tmachine 2.6.20-ARCH.

6.1 Nepenthes 0.2

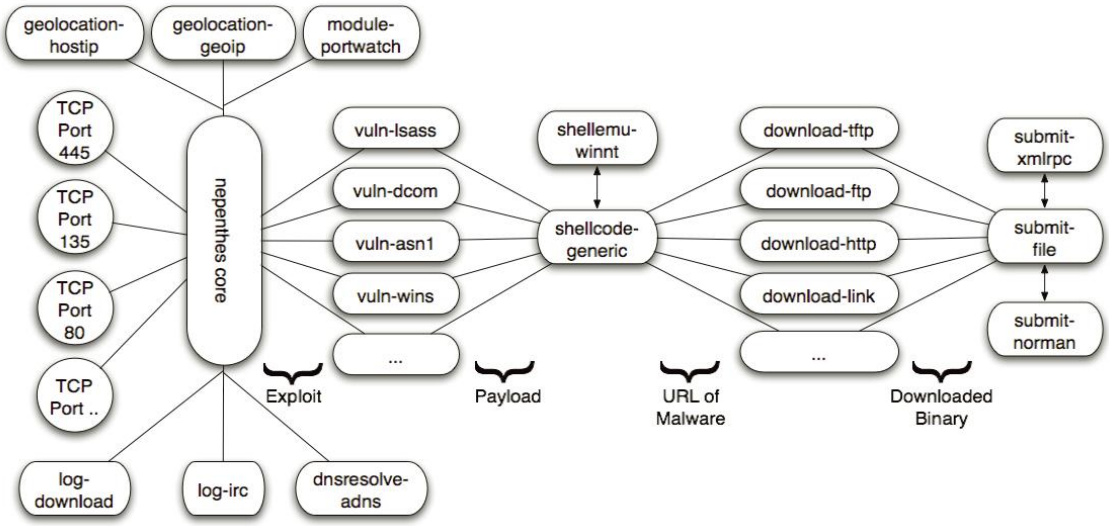
Nepenthes je honeypot s nízkou úrovní interakce určený především pro Unixové platformy (a Cygwin). Je vytvořen především pro emulaci "děravých" služeb a následné získávání informací o útocích na tyto služby – tzn. zjišťovaní způsobu šíření a kolekci červů a mallwaru.

Mallware je počítačový program určený ke vniknutí nebo poškození počítačového systému.

Honeypot Nepenthes se skládá především z těchto hlavních modulů:

- **Moduly zranitelnosti**: provádí emulací zranitelných částí síťových služeb.
- **Moduly analýzy**: analyzují data přijaté moduly emulující zranitelné služby.
- **Moduly přenosu dat**: určené k přenosu informací mezi jednotlivými instancemi Nepenthes.
- **Úložné moduly**: starají se o stáhnutý mallware (uložení na pevném disku, přenos do DB, posílání antivirovým společnostem, ...).
- **Logovací moduly**: zapisují informace o procesu emulace a pomáhají k celkovému přehledu o získaných datech.

Samotný Nepenthes navíc obsahuje několik dalších součástí, které jsou důležité pro funkčnost a efektivitu celého programu: emulace shellu, virtuální souborový systém (pro emulovaný shel), sniffovací moduly, ... Schéma interakce mezi jednotlivými moduly je zobrazeno na *Obr. 8*.



Obr. 8: *Schéma interakce mezi jednotlivými částmi Nepenthes [11]*

Seznam portů na kterých Nepenthes simuluje naslouchání emulovaných služeb. Dostupný pomocí příkazu "lsof –i":

```
nepenthes 5088 root   9u  IPv4  19141      TCP *:smtp (LISTEN)
nepenthes 5088 root  10u  IPv4  19142      TCP *:pop3 (LISTEN)
nepenthes 5088 root  11u  IPv4  19143      TCP *:imap (LISTEN)
nepenthes 5088 root  12u  IPv4  19144      TCP *:imap3 (LISTEN)
nepenthes 5088 root  13u  IPv4  19145      TCP *:465 (LISTEN)
nepenthes 5088 root  14u  IPv4  19146      TCP *:imaps (LISTEN)
nepenthes 5088 root  15u  IPv4  19147      TCP *:pop3s (LISTEN)
nepenthes 5088 root  16u  IPv4  19148      TCP *:2745 (LISTEN)
nepenthes 5088 root  17u  IPv4  19149      TCP *:6129 (LISTEN)
nepenthes 5088 root  18u  IPv4  19150      TCP *:135 (LISTEN)
nepenthes 5088 root  19u  IPv4  19151      TCP *:445 (LISTEN)
nepenthes 5088 root  20u  IPv4  19152      TCP *:1025 (LISTEN)
nepenthes 5088 root  20u  IPv4  19153      TCP *:ftp (LISTEN)
nepenthes 5088 root  21u  IPv4  19154      TCP *:https (LISTEN)
nepenthes 5088 root  22u  IPv4  19155      TCP *:17300 (LISTEN)
nepenthes 5088 root  23u  IPv4  19156      TCP *:2103 (LISTEN)
nepenthes 5088 root  24u  IPv4  19157      TCP *:eklogin (LISTEN)
nepenthes 5088 root  25u  IPv4  19158      TCP *:2107 (LISTEN)
nepenthes 5088 root  26u  IPv4  19159      TCP *:3372 (LISTEN)
nepenthes 5088 root  28u  IPv4  19160      TCP *:3127 (LISTEN)
nepenthes 5088 root  29u  IPv4  19161      TCP *:netbios-ssn (LISTEN)
nepenthes 5088 root  30u  IPv4  19162      TCP *:3140 (LISTEN)
nepenthes 5088 root  31u  IPv4  19163      TCP *:5554 (LISTEN)
nepenthes 5088 root  32u  IPv4  19164      TCP *:1023 (LISTEN)
nepenthes 5088 root  33u  IPv4  19165      TCP *:27347 (LISTEN)
nepenthes 5088 root  34u  IPv4  19166      TCP *:5000 (LISTEN)
nepenthes 5088 root  35u  IPv4  19167      TCP *:10000 (LISTEN)
nepenthes 5088 root  36u  IPv4  19168      TCP *:nameserver (LISTEN)
nepenthes 5088 root  37u  IPv4  19169      TCP *:www (LISTEN)
```

Nepenthes je honeypot speciálně zaměřený na detekci a získávání malware. Je schopen emulovat zranitelné části služeb, které je možné dál konfigurovat a přizpůsobovat. Hlavní výhodou tohoto honeypotu je jeho flexibilita. Umožňuje zachytit i tzv. zero exploits (tzn. exploity, které ještě nebyly zaznamenány a popsány). Není příliš náročný na konfiguraci a následný provoz.

Následující tabulka ukazuje seznam emulovaných služeb - zranitelností, které jsou součástí základního programu. Případné další emulované služby je možné si stáhnout přímo ze stránek projektu nebo je doprogramovat.

Name	Reference
vuln-asn1	ASN .1 Vulnerability Could Allow Code Execution (MS04-007)
vuln-bagle	Emulation of backdoor from Bagle worm
vuln-dcom	Buffer Overrun In RPC Interface (MS03-026)
vuln-iis	IIS SSL Vulnerability (MS04-011 and CAN-2004-0120)
vuln-kuang2	Emulation of backdoor from Kuang2 worm
vuln-lsass	LSASS vulnerability (MS04-011 and CAN-2003-0533)
vuln-msdtc	Vulnerabilities in MSDTC Could Allow Remote Code Execution(MS05-051)
vuln-msmq	Vulnerability in Message Queuing Could Allow Code Execution(MS05-017)
vuln-mssql	Buffer Overruns in SQL Server 2000 Resolution Service (MS02-039)
vuln-mydoom	Emulation of backdoor from myDoom/Novarg worm
vuln-optix	Emulation of backdoor from Optix Pro trojan
vuln-pnp	Vulnerability in Plug and Play Could Allow Remote Code Execution (MS05-039)
vuln-sasserftpd	Sasser Worm FTP Server Buffer Overflow (OSVDB ID: 6197)
vuln-ssh	Logging of SSH password brute-forcing attacks
vuln-sub7	Emulation of backdoor from Sub7 trojan
vuln-wins	Vulnerability in WINS Could Allow Remote Code Execution (MS04-045)

Tab. 5: *Seznam zranitelnostních modulů*

Nepenthes používá jako základní nástroj pro detekci a kolekci mallwaru i český projekt ***honeynet*** (v současné době má na tomto honeypotu postavených 13 senzorů). Projekt je primárně zaměřen na distribuovanou detekci virů/mallware, jeho kolekci, následnou forensní analýzu a tvorbu výstupů formou grafů či textových statistik. Bližší informace k projektu jsou dostupné na `http://honeynet.cz/` .

6.1.1 Souhrnné informace o nepenthes

Cena	zdarma
Systém	Unix
Míra interakce	nízká
Stránky projektu	http://nepenthes.mwcollect.org/

Tab. 6: *Základní Informace o Nepenthes*

Nepenthes je nenativní honeypot, který emuluje "děravé" služby především pro MS OS. Tyto služby dovolí napadat útočníky a následně ukládá získané logy, binární či textové soubory z těchto útoků pro další analýzu.

6.2 KFsensor Profesional

KFSensor je kombinace honeypotu a IDS systému postaveného na platformě Windows. Svou cenou a povahou je určený spíš pro firemní prostředí než pro výzkumné účely. Simulací zranitelných služeb (na nejvyšší vrstvě modelu OSI) působí jako lákadlo na útočníky, proto poskytuje víc informací než běžné IDS. Obsahuje prvky jako vzdálenou správu, kompatibilitu se Snortem a emulaci windowsovských síťových protokolů. Ovládá se pomocí grafického prostředí.

Skládá se ze dvou částí:

- KFSensor Server: naslouchá na TCP a UDP portech serveru a generuje příslušné události. Nemá uživatelský interface a pracuje jako služba/konzolová aplikace systému.
- KFSensor Monitor: obsahuje uživatelské rozhraní a monitoruje události vygenerované KFSensor Serverem.

Výhody a schopnosti KFSensor:

- Identifikace útoku: je schopen identifikovat známé vzory útoků a tím pádem i analyzovat povahu události (v KFSMonitoru se projeví barvou ikony u události). Pravidla mohou být importována z externích zdrojů ve Snort-formátu.
- Detekce útoků na Windows-síťových protokoly: je schopen emulovat NetBios, SMB, CIFS služby, bez rizika přerušení jejich činnosti
- Simulace serverových služeb: je schopen emulovat, FTP, SMB, POP3, Telnet, SMTP a SOCKS.

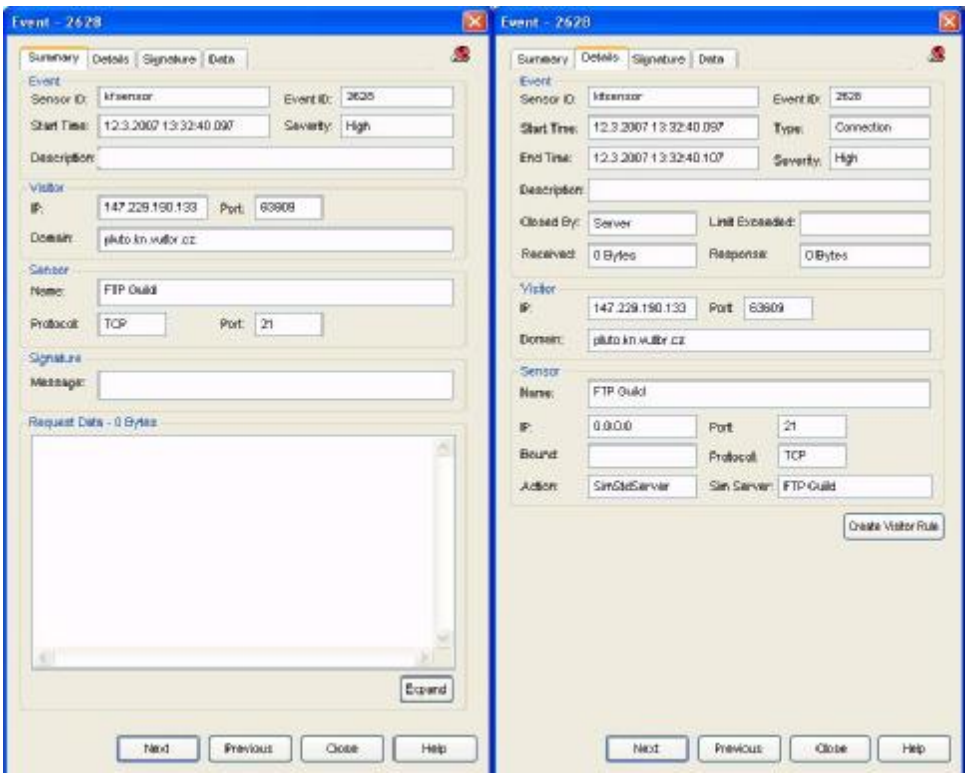
- Rozšiřitelná architektura: události a emulace KFS mohou být dále rozšířeny psaním vlastních skriptů.
- Minimum False Positives: minimum planých poplachů.
- Vzdálená správa.
- Detekce a upozorňování v reálném čase, viz. obrázky 9-10 s ukázkou události.

Nevýhody:

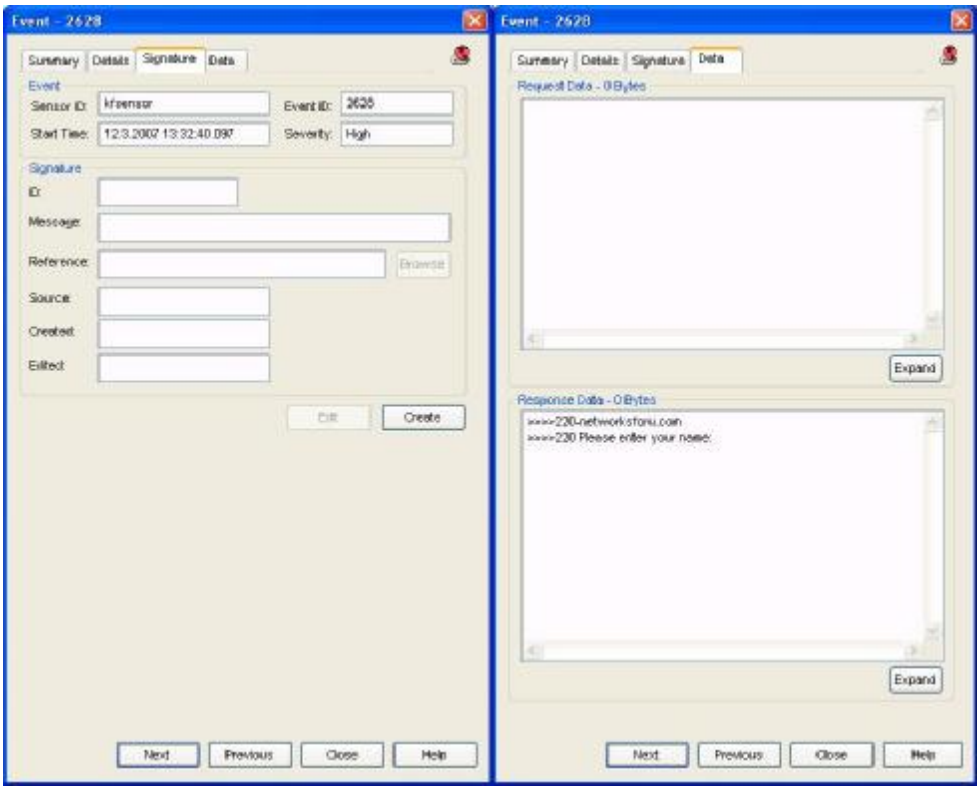
- cena,
- pouze platforma Windows,
- HW nároky,
- problematické logování.

6.2.1 Nasazení

KFSensor byl nasazen a testován na kolejní síti pod OS Windows XP SP2 na HP Pentium 1.7 GHz, 512 MB RAM. Zachycený provoz odpovídal "útokům" na kolejní síti, viz. následující obrázky, kde je zachycen pokus serveru pluto.kn.vutbr.cz o nalogovaní na lokální ftp server. Log tohoto útoku s ukázkou hlavního okna KFSensor se nachází v příloze 2.



Obr. 9: *Ukázka události v KFSensor*



Obr. 10: *Ukázka události v KFSensor*

Cena	14 dnů trial; 199 – 599 USD
Systém	Windows
Míra interakce	nízká - střední
Stránky projektu	http://www.keyfocus.net/kfsensor/

Tab. 7: *Základní Informace o KFSensor*

6.3 HoneyD

Honeyd je malý démon vytvářející v síti virtuální počítače. Každý z nich může předstírat, že je na něm spouštěný určitý operační systém a konkrétní sada služeb. Kromě simulací těchto služeb, mohou rovněž působit jako zprostředkovatelé předávající informace jinému počítači. Díky Honeyd lze pomocí jednoho počítače simulovat existenci celých počítačových sítí. Nejnovější verze přináší především vnitřní webový server, který ukazuje statistiky pohybu dat a možnost upozornění na jednotlivé útoky.

Honeyd může být sloužit k vytvoření upraveného honeynetu nebo být použit jako jeden z hlavních monitorovacích systémů lokální sítě. Podporuje vytváření virtuální síťové topologie (včetně vyhrazených tunelů, routerů a začlenění reálných strojů). Jednotlivé cesty mohou mít vlastnosti opravdových spojení – latenci, ztrátovost paketů a jiné atributy, které zvyšují zdání

reálnosti sítě. Protože honeyd pracuje s potenciálně nebezpečnými útočníky, je vhodné použít nějaký bezpečnostní prvek (např. Systrace), který zabrání útočníkovi zneužít případné bugy v honeyd.

6.3.1 Vlastnosti

Pomocí honeyd je možné vytvořit množství variant démonů jak jednotlivých strojů, tak síťových systémů. Následující seznam ukazuje podporované vlastnosti:

- Simulace tisíců virtuálních strojů v reálném čase.
- Konfigurace libovolných služeb pomocí konfiguračních souborů:
  - komunikace pomocí proxy,
  - pasivní odposlouchávání,
  - náhodné vzorkování loadu.
- Simulace OS na úrovni TCP/IP:
  - schopnost "ošidit" nmap a xprobe,
  - nastavení směrování.
- Simulace libovolných směrovacích topologií:
  - konfigurovatelnost zpoždění a ztrátovosti paketů,
  - asymetrické směrování,
  - integrace reálných strojů do virtuální topologie.
- Virtualizace podsystémů:
  - provozování reálných Unixových aplikací na virtuální Honeyd IP adrese (webové servery, ftp servery, ...),
  - dynamické portování ve virtuálním adresovém prostoru, zahájení spojení na pozadí, ...

6.3.2 Virtualizace subsystému

Honeyd podporuje virtualizaci služeb prováděním unixových aplikací jako podsystémů spuštěných ve virtuálním IP adresovém prostoru nakonfigurovaného honeypotu. To umožňuje jakékoliv síťové aplikaci dynamicky se připojit na port a vytvořit TCP/UDP spojení použitím virtuální IP adresy.

Subsystémy jsou virtualizovány zachycením síťového požadavku a přesměrovány na Honeyd. Každá konfigurace (honeyd) může obsahovat podsystémy, které se spouští jako samostatné procesy, přičemž konfigurace virtuálního stroje je vázaná k virtuální IP adrese. Dodatečná výhoda tohoto řešení je schopnost honeypotů vytvářet sporadický provoz, jako dotazy na www stránky, mailovou komunikaci, ...

6.3.3 Směrování

Honeyd podporuje asymetrické směrování a integraci reálných strojů do virtuální síťové topologie. Z toho důvodu je možné honeyd využít k jednoduché síťové simulaci.

Reálné stroje mohou být vystaveny vysoké latenci nebo ztrátovosti paketů. Viz. následující příklad:

```
route entry 10.0.0.1 network 10.0.0.0/24
route 10.0.0.1 link 10.0.0.0/24
route 10.0.0.1 add net 10.4.0.0/14 tunnel "thishost" "honeyd-b"
route 10.0.0.1 add net 10.1.0.0/16 10.1.0.1 latency 55ms loss 0.1
route 10.0.0.1 add net 10.2.0.0/16 10.2.0.1 latency 20ms loss 0.1
route 10.0.0.1 add net 10.3.0.0/16 10.2.0.1 latency 20ms loss 0.1
route 10.1.0.1 link 10.1.0.0/24
route 10.2.0.1 link 10.2.0.0/24
[...]
route 10.2.0.1 add net 10.3.0.0/16 10.3.0.1 latency 10ms loss 0.1
route 10.3.0.1 link 10.3.0.0/24
route 10.3.0.1 add net 10.3.1.1/24 10.3.1.1 latency 10ms
route 10.3.0.1 add net 10.3.240.0/20 10.3.240.1 latency 5ms
route 10.3.1.1 link 10.3.1.1/24
route 10.3.240.1 link 10.3.240.0/20
route 10.3.240.1 add net 0.0.0.0/0 10.3.0.1 latency 40ms loss 0.5
[...]
bind 10.2.0.243 to fxp0
bind 10.3.1.15 to fxp0
```

6.3.4 Konfigurace

Před samotnou konfigurací virtuálních strojů je nutné jim zajistit dostatečný adresový prostor. To umožní nástroj “arpd“ příkazem:

```
arpd –i eth0 10.0.0.0/24
```

Nyní bude odpovídat host, kde jsme spouštěli arpd z celého rozsahu adres – tedy 10.0.0.x. V praxi to funguje tak, že pokud se někdo bude chtít spojit s například s počítačem s IP adresou 10.0.0.5, vyšle se požadavek a něj odpoví počítač 10.0.0.1 (počítač na kterém jsem spouštěli arpd proti rozhraní eth0), ale s IP adresou 10.0.0.5. Předtím než začnete testovat arpd, ujistěte se, že testovaní provádíte opravdu na své virtuální síti (např. pod VMware). Konfigurace virtuálního stroje

Příklad konfigurace virtuálního stroje:

```
### Linux 2.4.x computer
create linux
set linux personality "Linux 2.4.16 - 2.4.18"
set linux default tcp action reset
set linux default udp action reset
add linux tcp port 110 "sh scripts/unix/general/pop/emulate-pop3.sh"
add linux tcp port 25 "perl scripts/unix/general/smtp.pl"
add linux tcp port 21 "sh scripts/unix/linux/ftp.sh"
set linux uptime 3284460
bind 192.168.20.103 linux
```

Popis jednotlivých řádků konfigurace virtuálního stroje:

`create linux`

Vytvoří systém Linux..

`set linux personality "Linux 2.4.16 - 2.4.18"`

Definuje, že se jedná o linux s jádrem 2.4.16–2.4.18.

`set linux default tcp action reset`

Nastaví odpověď na TCP paket.

`set linux default udp action reset`

Nastaví odpověď na UDP paket.

`add linux tcp port 110 "sh scripts/unix/generál/pop/emulate-pop3.sh"`

Nastaví port 110 jako otevřený a definuje, který skript na portu poslouchá.

`add linux tcp port 25 "perl scripts/unix/generál/smtp.pl"`

`add linux tcp port 21 "sh scripts/unix/linux/ftp.sh"`

`set linux uptime 3284460`

Definuje, dobu od startu systému.

`bind 192.168.20.103 linux`

Definuje IP adresu stroje.

6.3.4.1 Spuštění honeyd

Příklad spuštění honeyd:

```
./honeyd -f honeyd.conf -p nmap.prints -x xprobe2.conf -a nmap.assoc
-0 pf.os -l /var/log/honeyd 192.168.20.101-192.168.20.104
```

`-f <soubor>`

Určuje soubor, ze kterého souboru se bude brát konfigurace.

`-x <soubor>`

Soubor s definicí, jak honeyd reaguje na nástroje využívající ICMP fingerprint, pro určení operačního systému scanovaného zařízení.

`-l <soubor>`

Říká, kam logovat události spojené s honeyd (např. příjem síťového paketu).

`-a <soubor>`

Soubor, který asociuje styl fingerprintů pro xprobe (možnost -x) se stylem fingerprintů, které používá nmap pro určení operačního systému.

`-0 <soubor>`

Soubor s databází fingerprintů. Jména operačních systému specifikována v tomto souboru používá honeyd jako dynamickou šablonu.

`10.0.0.1–10.0.0.255`

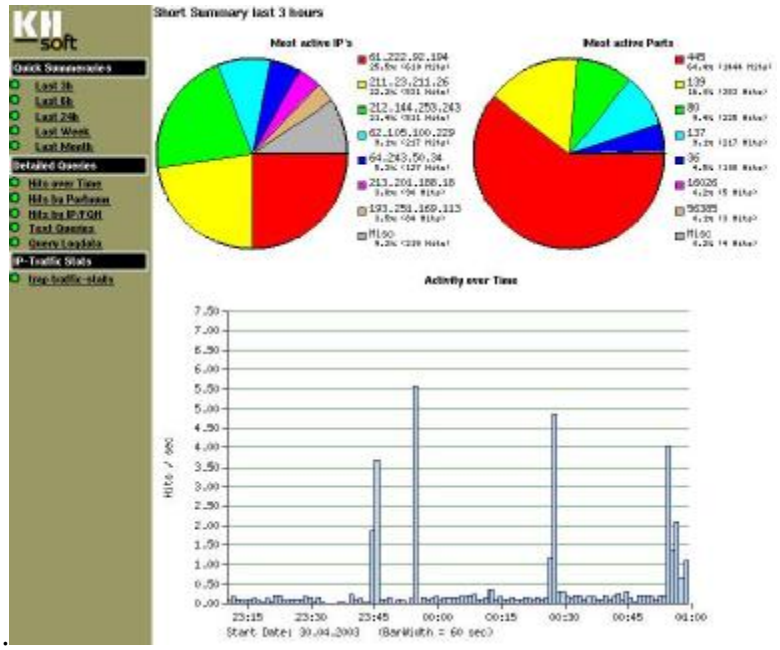
Poslední je rozsah IP adres, na kterém honeyd poslouchá.

6.3.5 Ovládací konzole

Pomocí konzole (honeydctl) lze dynamicky konfigurovat za běhu honeyd.

6.3.6 Shrnutí

Honeyd můžeme považovat za jedno z nejlepších freewarových řešení honeypotů. Mezi jeho výhody patří průhledný zdrojový kód, výborné možnosti parametrizace a výše uvedené schopnosti vytváření a směrování virtuálních strojů a sítí. Honeyd je vynikajícím nástrojem, který navíc lze doplnit některou z řady nadstaveb (seznam lze nalézt na stránce `www.honeyd.org/tools`). Vynikající je například HoneyView, který umožňuje grafické znázornění mimo jiné využití jednotlivých služeb, viz. známý obrázek, který zobrazuje nejvíce aktivní IP adresy.



Obr. 11: *HoneyView – grafické rozšíření honeyd*

Další výhodou je rozsáhlá dokumentace, ať už formě manuálových stránek, nebo příkladů na internetu. Honeyd bych ovšem nedoporučoval jako honeypot určený pro seznámení s těmito systémy. Pro jeho správnou funkci a využití je již nutné plně chápat koncepty honeypotů.

Cena	zdarma
Systém:	Unix
Míra interakce	nízká – střední
Stránky projektu	http://www.honeyd.org/

Tab. 8: *Základní Informace o HoneyD*

6.4 Multipot v0.3

IDefense Multipot je honeypot vytvořený Davidem Zimmeremm [20], určený především k zachytávání exploitů šířících se po síti. Je distribuovaný jako free software s licencí GNU General Public License pod operačními systémy Windows. Hlavním předpokladem toho projektu bylo vytvořit honeypot, nenáročný na údržbu, instalaci a bez rizika průniku útočníka na hostitelský stroj, tzn. je vytvořený k emulaci zranitelných služeb a bezpečnému sbírání záludného kódu.

Zatím se vývojářům nepodařilo výše uvedené požadavky na honeypot dokonale splnit. Multipot potřebuje specificky nakonfigurovaný systém a pro uživatele, který nemá kvalitní znalosti z počítačových sítí, to může být obtížné. Vzhledem k bezpečnosti sami tvůrci přiznávají, že je vhodné spouštět Multipot na vyhrazených strojích. Na druhou stranu veškeré konfigurační možnosti jsou dostupné typickým "jednoduchým" windowsovským klikacím způsobem.

Multipot dokáže reagovat pouze na tyto exploity:

- MyDoom na TCP portu 3127,
- Otpix na TCP portech 2060 a 500,
- Beagle na TCP portech 2745 a 12345,
- Sub7 na TCP portu 27347,
- Kuang na TCP portu 17300,
- Veritas na TCP portech 6101 a 10000.

Seznam modulů, ze kterých se Multipot skládá:

- clsServer: řídí připojení socketů a datový tok,
- clsUpload: řídí zápis do logů a ukládání získaných dat (je možné propojit logování s databází MS Access),
- CAntiHammer: zabezpečovací systém,
- Modul oznamování na události.

6.4.1 Shrnutí

Multipot patří mezi honeypoty s nízkou mírou interakce. Je zaměřen na zachycení výše uvedených exploitů a tedy nepoužitelný pro tzv. zero-exploity. V dnešní době lze nalézt mnoho kvalitnějších a lépe konfigurovatelných honeypotů.

Cena	zdarma
Systém	Windows
Míra interakce	nízká
Stránky projektu	http://www.idefense.com

Tab. 9: *Základní Informace o Multipot v0.3*

6.5 Honeywall - honeynet

Honeywall je koncepce honeypotu s vysokou mírou interakce ze stejnojmenného projektu, který je dostupný na [14]. Jedná se o vlajkovou loď projektu Honeynet [4]. Tento nástroj umožňuje zachycení, kontrolu, analýzu útoků a především vytváří architekturu, která poskytne možnosti k rozmístění honeypotů s nízkou i vysokou mírou interakce. Celá koncepce Honeywallu a Honeynetu je popsána v kapitole 4.

6.5.1 Popis

Celý nástroj je dostupný ve formě bootovacího CD. Je postavený na upravené instalaci operačního systému Fedora Core 3 (s přidanými balíčky zaměřenými na bezpečnost). Tato Honeywall brána filtruje a sleduje veškerou příchozí a odchozí komunikaci na druhé vrstvě. Postup instalace je dostupný na [16]. Po instalaci honeywallu je samozřejmě nutné nainstalovat jednotlivé honeypoty - už reálné systémy s definovanými službami.

Honeywall obsahuje dva hlavní nástroje na získávání a ukládání dat:

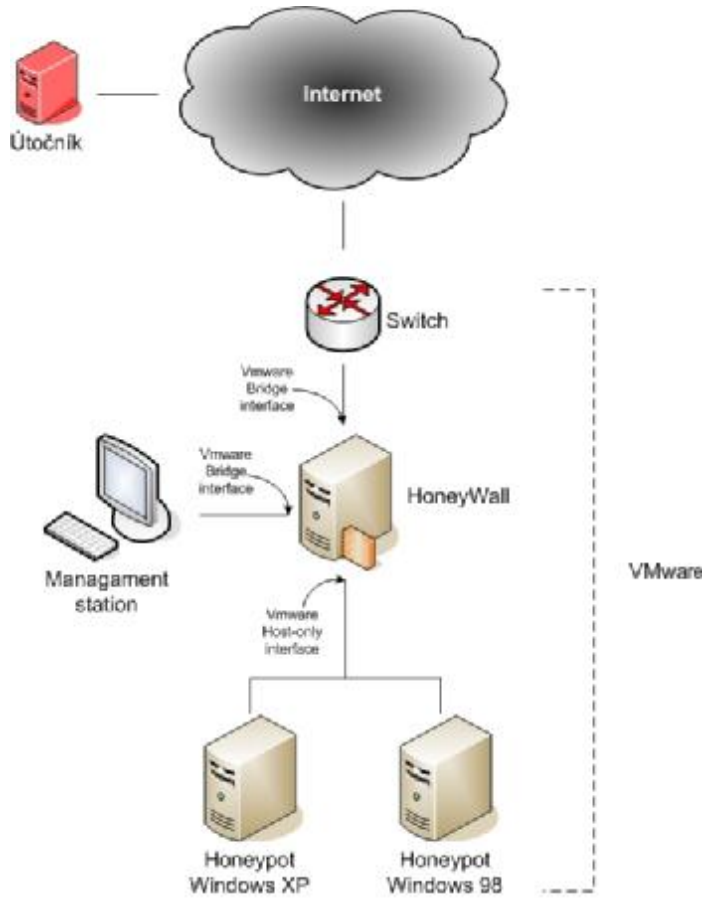
- Sebek: nástroj umožňuje zachytávat aktivitu útočníka na honeypotu, aniž by o tom věděl. Skládá se z klientské části, která běží na honeypotu a zachytává veškerou útočníkovu činnost na honeypotu a následně posílá data na serverové části. Serverová část sbírá data z jednotlivých klientů a poté je ukládá do DB, nebo případně do jiného skladiště. Serverová část je umístěna většinou na honeywall bráně.
- Snort: Jedná se open source software určený k prevenci a detekci proniknutí do síťové infrastruktury. Využívá soubor pravidel k odhalení těchto průniků. Jedná se o jeden z nejpoužívanější IDS systémů na světě.

Způsoby administrace:

- Hwctl: utilita, která umožňuje měnit systémové hodnoty z příkazové řádky (jednotlivé změny se zapisují do /etc/honeywall.conf) a následně nabídne restart jedné služby nebo celé Honeywall.
- Dialogové menu: klasické rozhraní Honeywallu (dostupná nastavení jsou popsána na *http://www.honeynet.org/tools/cdrom/roo/manual-1.1/txt/dialog-menu.txt*).
- webové rozhraní: umožňuje vzdálenou a přehlednou administrativu honeywallu se všemi funkcemi jako předchozí dva způsoby včetně analýzy dat.

6.5.2 Vlastní řešení

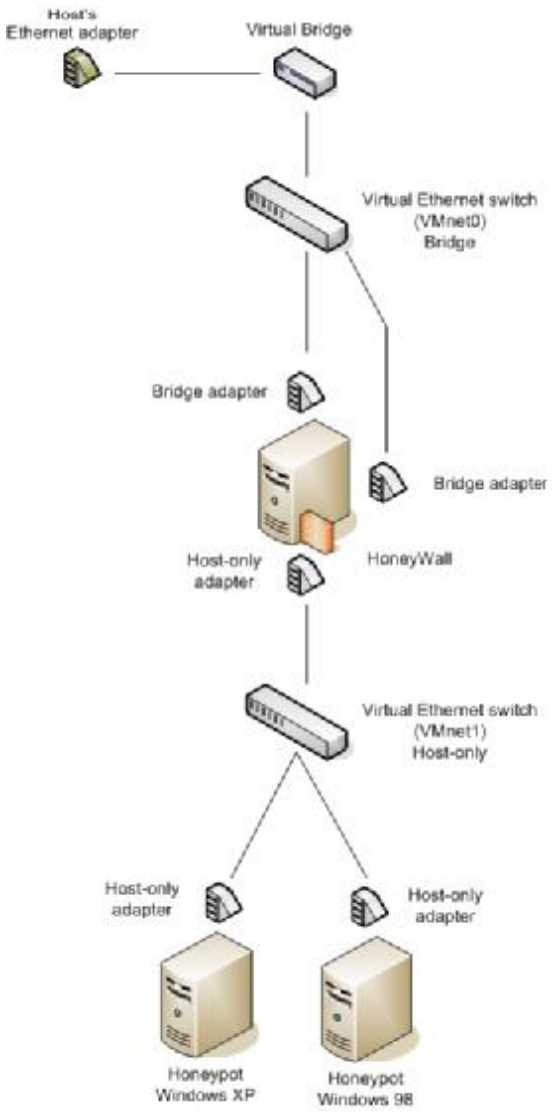
Vzhledem k nedostatku HW, celá koncepce byla testována ve virtuální síťové topologii na lokální stanici pomocí komerčního programu VmWare [15]. Z vnějšku by mnou vytvořené řešení vypadalo jako na obrázku 12.



Obr. 12: *Nasazení HoneyWall pod VMware*

Všechny virtuální stroje běžely na jediném fyzickém PC většinou s 1,5 GB RAM a s operačním systémem Windows XP, následně byly nainstalovány různé OS pro virtuální honeypoty (Windows XP a Windows 98). Virtuální honeypoty jsou směrovány přes HoneyWall pomocí VMwareNetwork. Podrobný návod je možné nalézt na [17].

Konfiguraci sítě uvnitř VMware přibližuje obrázek 13.



Obr. 13: *Vnitřní upořádání Honeywallu pomocí VMware*

Celý honeynet je připojen k jednomu reálnému síťovému adaptéru. Virtuální směrovač umožňuje připojit virtuální stroje k síti pomocí reálného adaptéru dvěma hlavními způsoby:

- Host-only: je virtuální síťový adaptér, který nemá vlastní IP adresu, ale sdílí ji s reálným strojem (tzn. jedná se privátní síť. adaptér dostupný z hostovaného stroje).
- Bridged: připojuje přímo pomocí vlastní IP adresy k reálné síti.

6.5.3 Shrnutí

Honeywall je výbornou ukázkou honeypotu s vysokou mírou interakce. Jsou nasazeny reálné systémy "za" honeywallem, který je schopen zachytit veškerou komunikaci a případně ji vhodným způsobem omezit, aby útočník nemohl naše systémy využít k další nekalé činnosti. Nasazení a správná konfigurace honeywallu je obtížnější, než u honeypotů s nízkou mírou interakce. Na druhou stranu je tato obtíž vyvážena hodnotou získaných dat.

Cena	free
Systém	Unix
Míra interakce	vysoká
Stránky projektu	http://www.honeynet.org/tools/cdrom/

Tab. 10: *Základní Informace o Honeywall*

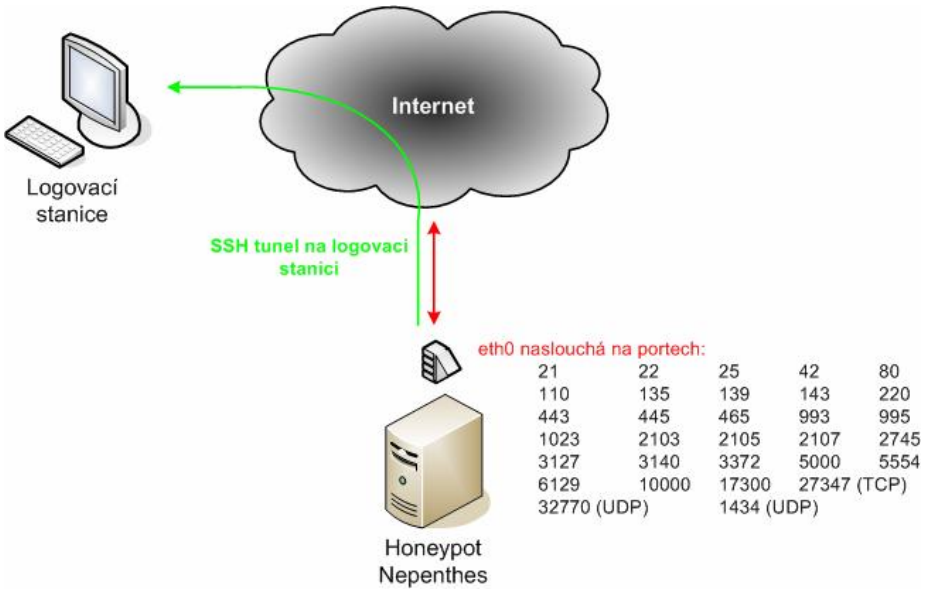
7 Nasazení honeypotu

Honeypot byl provozován na stroji s nefiltrovaným přístupem k Internetu pod operačním systémem ArchLinux [19] s níže uvedenou konfigurací:

i686 Intel(R) Pentium(R) 4 CPU 2.40GHz
Mem: 524288 kB
Intel Corporation 82562EZ 10/100 Ethernet Controller (rev 02)
Linux tmachine 2.6.20-ARCH

Pro nasazení honeypotu byl vybrán program Nepenthes (popsaný v kapitole 6.1). Jedná se honeypot s nižší úrovní interakce, proto nebylo možné plně zachytit a analyzovat všechny útoky na honeypot. Tato nevýhoda je ovšem vyvážena nepříliš náročnou instalací, konfigurací programu a možností velmi rychlé obnovy.

Dohled nad síťovým provozem na honeypotu zajišťoval program tcpdump [18]. Slabinou honeypotu bylo ukládání získaných dat, která se ukládala lokálně a jednou za čtyři dny byla automatickým nástrojem přenesena na logovací stanici.



Obr. 14: *Znázornění nasazeného honeypotu*

7.1 Instalace Nepenthes

Před samotnou instalací bylo nutné nainstalovat balíčky, které Nepenthes využívá:

- Adns: program umožňující na asynchronní překlad DNS.
- Curl: konzolová aplikace určená pro přenos souborů pomocí FTP, FTPS, HTTP, HTTPS, ...
- GeoIP: program zjišťující geografické údaje z IP adresy.
- Libcap: knihovna určená k čtení a nastavování capabilities podle podle návrhu 15 standardu POSIX.1e (původně POSIX 6).
- Pcre: jedná se o knihovnu umožňující práci s regulárními výrazy kompatibilními s Perlem

Velkou výhodou při instalaci honeypotu byl už sestavený balíček programu Nepenthes pro Archlinux. Instalace se zjednodušila na instalaci závislých balíčků, stažení souboru *pkgbuild aur.archlinux.org/packages/nepenthes/nepenthes/pkgbuild* a spuštění příkazu `makepkg –i` v dresáři s buildovacím souborem.

Soubor *pkgbuild* (dostupný na *http://aur.archlinux.org/packages/nepenthes/nepenthes/*) je skript, který na základě parametrů provádí mimo jiné tyto činnosti:

`depends=('adns' 'curl' 'geoip' 'libcap' 'pcre')`

Kontrola závislostí.

`source=(http://dl.sourceforge.net/sourceforge/nepenthes/nepenthes-$pkgver.tar.bz2 nepenthes nepenthes.conf nepenthes.confd)`

Stažení soboru z repozitoře.

`./configure--prefix=/usr --sysconfdir=/etc --localstatedir=/var/lib/nepenthes --enable-capabilities`

Konfigurace instalace. Informace k jednotlivým parametrům jsou dostupné po zapsání příkazu `configure –h`.

`make && make install`

Finální instalace.

Při instalaci je vytvořen uživatel Nepenthes ze skupiny Nepenthes, bez shellu a bez možnosti se pomocí hesla přihlásit. Pod tímto účtem se spouští démon Nepenthes.

7.2 Spuštění a konfigurace Nepenthes

7.2.1 Adresářová struktura

Pro lepší představu o fungování honeypotu je nutné alespoň z části přiblížit adresářovou strukturu programu Nepenthes. Řetězec `<dir>` označuje adresář s instalací programu.

`<dir>/etc/nepenthes/`

Adresář s konfiguračními soubory.

`<dir>/lib/nepenthes/`

Adresář s knihovnami, rozšířeními a jednotlivými moduly.

`<dir>/var/log/`

Adresář se soubory logů.

`<dir>/var/binaries/`

Adresář se staženými binárními, spustitelnými soubory.

`<dir>/var/hexdumps/`

Adresář se staženými škodlivými kódy.

7.2.2 Nastavení honeypotu

Konfigurace honeypotu je rozdělena do dvou částí, na konfiguraci hlavního programu a konfiguraci jednotlivých modulů. V hlavní konfigurační soubor `<dir>/etc/nepenthes/nepenthes.conf` byl nastaven následujícím způsobem (výchozí hodnoty nejsou uváděny):

- Povolení modulu pro stahování pomocí `"tftp: downloadtftp.so"`.
- Povolení modulu pro odesílaní stažených binárních souborů na analýzu: `"submitnorman.so"`.
- Povoleny všechny moduly zranitelností.
- Zakázáno logování pomocí IRC: `"logirc.so"`.
- Nastavení cesty pro stahované soubory: `"filesdir var/binaries/"`
- Nastavení cessty pro stahované skripty: `"hexdump_path var/hexdumps/"`
- Nastavení socket manageru a ip adresy: `"bind_address 147.229.7.3"`

Ostatní konfigurační soubory jednotlivých modulů se nacházejí v adresáři `<dir>/etc/nepenthes/`. Z 39 souborů byly upraveny konfigurační soubory:

- Submit-norman.conf: nastavuje automatické odesílání souborů k analýze.

```
submit-norman
{
        // this is the adress where norman sandbox reports will be sent
        email   "<email prijemce analyzy>@email.cz";
        urls    ("http://sandbox.norman.no/live_4.html",
        "http://luigi.informatik.uni-
        mannheim.de/submit.php?action=verify");
};
```

- Submit-gotek.conf: nastavuje automatické odesílání souborů k analýze.

```
submit-gotek
{
        host    "alliance.mwcollect.org";
        port    "34109";
        communitykey = "<klic>";
        user    "<uzivatelske jmeno>";
        spool
        {
               enable =        "1";
        // spool directory where submissions are saved until sent
        / you can also manually place files here to submit them and
        // restart nepenthes
               directory =     "var/spool/nepenthes/gotek";
         };
};
```

Další možnosti nastavení honeypotu Nepenthes je možné najít manuálových stránkách nebo na stránkách projektu.

7.2.3 Spuštění honeypotu

Před prvním spuštěním honeypotu bylo nezbytné zavést do jádra systému standardní bezpečnostní modul *capability*, příkazem: *modprobe capability*. Nutné je rovněž změnit vlastníka adresáře s nainstalovaným programem: *chown –R nepenthes.nepenthes /opt/nepenthes*

Program se spouští následujícím přikazem:

```
/opt/nepenthes/bin/nepenthes --user=nepenthes --group=nepenthes -C -R –D
```

Honeypot má při spuštění implicitně nastaveno využívání většiny balíčků uvedených v kapitole 7.1., proto není není nutné je při spouštění programu uvádět. Přehled jednotlivých parametrů, se kterými byl honeypot spouštěn:

`/opt/nepenthes/bin/nepenthes`

Cesta k binárnímu souboru.

`--user=nepenthes --group=nepenthes`

Nastaví uživatele a skupinu pod kterým se bude program spouštět

`-C`

Umožní využívat bezpečnostní modul jádra.

`-R`

Nastavení přesnějšího logování.

`-D`

Program běží jako démon na pozadí.

7.3 Režie honeypotu

Pro správnou analýzu dat, bylo nevyhnuté monitorovat síťový provoz na honeypotu. K tomuto účelu byl využit program tcpdump, který logoval veškerý provoz na síťové kartě, mimo určených stanic.

Zálohování logů na vzdálenou stanici bylo řešeno pomocí skriptu, který byl spouštěn z cronu v pondělí a ve čtvrtek. Skript fungoval následujícím zpusobem:

1. zkopíruj adresář `<dir>/var/log/` do `/tmp/download/`
2. zkopíruj výstup z tcpdumpu do `/tmp/download/`
3. zkomprimuj adresář `/tmp/download/` do `/tmp/download<date>.tar.gz`
4. odešli soubor `/tmp/download<date>.tar.gz` pomocí scp na logovací stanici (přístup pomocí klíčů)

7.4 Shrnutí

Instalace a konfigurace programu Nepenthes probíhala bez větších obtíží. Pouze některým modulům by slušela propracovanější dokumentace. Bohužel odesílání stažených souborů k analýze nefungovalo vždy bezchybně. Pokud podařilo soubor odeslat, mail s informacemi o daném souboru přišel se zpožděním několika dnů, nebo nepřišel vůbec. Proto bylo nutné analyzovat soubory ručně např. na ***www.honeynet.cz***.

8 Analýza získaných dat

Jedná se o analýzu získaných logů pomocí honeypotu popsaného v kapitole *7*. Honeypot byl provozován po dobu jednoho měsíce s minimálními odstávkami. První připojení na honeypot následovalo po dvaceti minutách od spuštění (na honeypot neukazoval žádný DNS server)

Jenom pro představu o velikosti provozu na honeypotu: výpis z logu tcpdumpu obsahoval přes čtyři sta tisíc řádků a výpis z logu programu Nepenthes přibližně tři tisíce řádků.

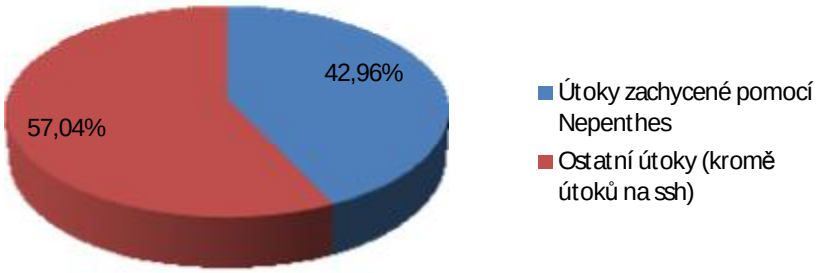
Několik dalších statistických údajů:

- Počet stažení škodlivého kódu: 629
- Počet stažení různého škodlivého kódu: 99
- Celkový počet stažení binárních souborů: 45
- Počet různých stažených binárních souborů: 8
- Počet přístupů z různých IP adres : 709

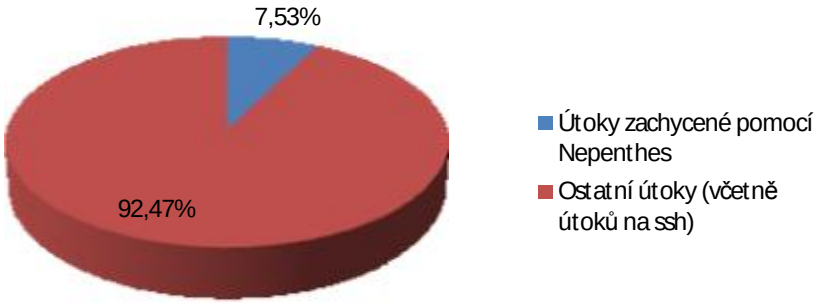
Při provozování honeypotu byly detekovány tři hlavními způsoby útoků na honeypot.

- skenování otevřených portů (útočník skenuje postupně všechny porty a následně zaměřuje útok),
- útoky silou na různé služby,
- útoky ”škodlivým kódem” na jednotlivé služby.

8.1 Procentuální vyjádření úspěšnosti zachycení útoků pomocí Nepenthes



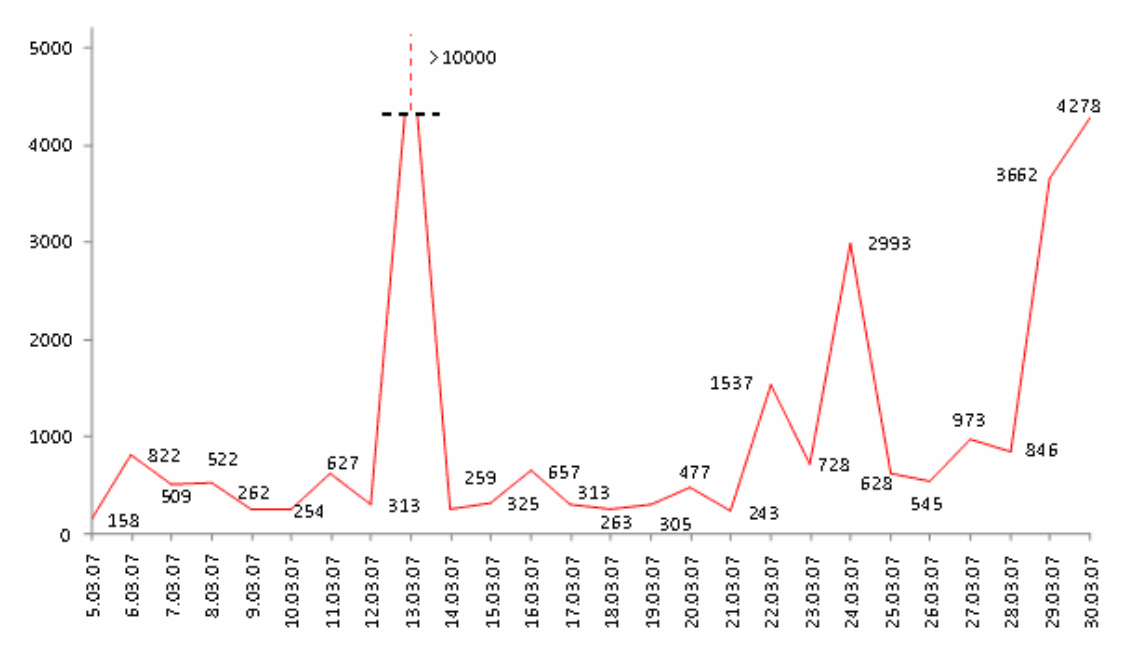
Obr. 15: *Poměr úspěšnosti zachycení útoků honeypotem Nepenthes (mimo útoků na ssh démona)*



Obr. 16: *Poměr úspěšnosti zachycení útoků honeypotem Nepenthes*

Obrázky 15 a 16 zachycují úspěšnost zachycení útoků honeypotem Nepenthes. Relevantní je především obrázek 16, k útokům na službu ssh se vrátíme v dalším textu. Mohlo by se zdát, že 43% úspěšnost detekce průniků je nízká, ale je nutné brát v potaz, že velké množství útoků/konexí na honeypot byl postupný sken portů. Bohužel nemám možnost tuto úspěšnost srovnat s jiným (obdobným) honeypotem,.

8.2 Počty spojení na honeypot v jednotlivých dnech

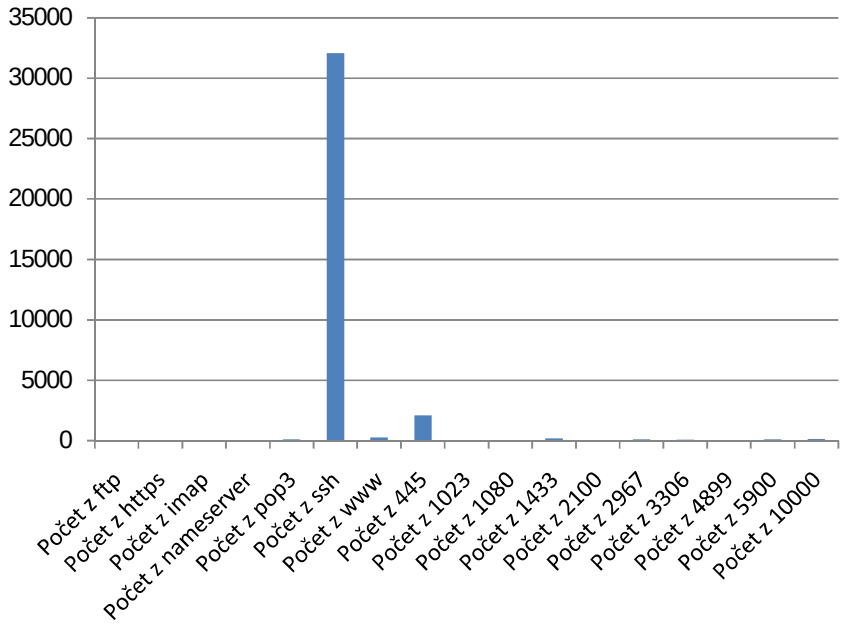


Obr. 17: *Počty spojení na honeypot v jednotlivých dnech*

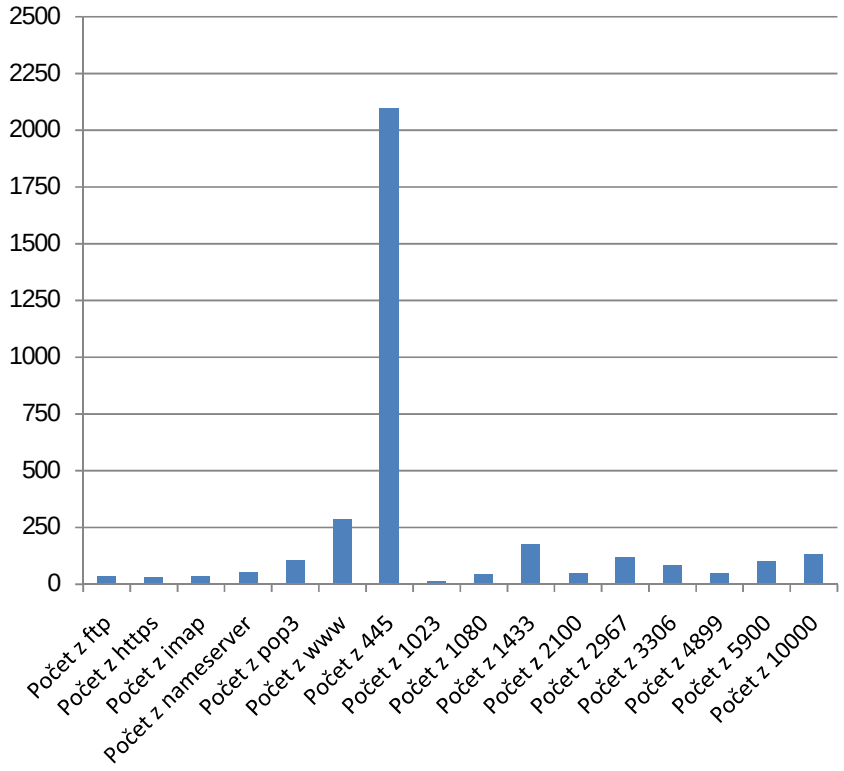
Graf znázorňuje počet spojení v průběhu jednoho měsíce. Je zřejmé, že počet konexí s časem postupně narůstá. Je nutné si odmyslet odchylku z 13. 3. 2007, kdy byl proveden masivní útok silou na ssh (pro zabránění úplné degradaci grafu, byla hodnota uměle zmenšena). Delší sledování by pravděpodobně odhalilo postupný nárůst počtu konexí až do ”maximálního stavu” – křivka by měla podobu první čtvrtiny sinusoidy. Abych tento předpoklad dokázal, je nutné provést měření v delším časovém intervalu.

8.3 Počty spojení na honeypot - rozdělení podle portů

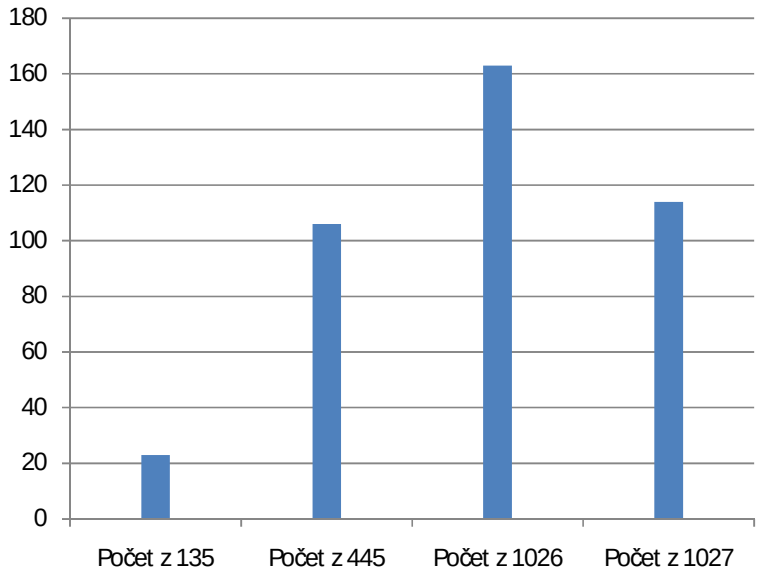
Graf na obrázku číslo 18 ukazuje počet spojení na porty TCP včetně portu 22, na kterém naslouchá ssh démon. Kvůli již zmíněnému útoku hrubou silou na tohoto démona je počet spojení na tento port nejvyšší. Graf na obrázku číslo 19 ukazuje shodné hodnoty jako graf z obrázku 18, ale není již zahrnut port ssh. Na obou grafech jsou pouze porty, na kterých byly počty spojení větší než 26. Největší počet spojení byl na port 445, na kterém běží MS-DS a které jsou cílem většiny útoků.



Obr. 18: *Celkový počet spojení (TCP) včetně portu 22 - ssh*



Obr. 19: *Celkový počet spojení (TCP) mimo port 22*

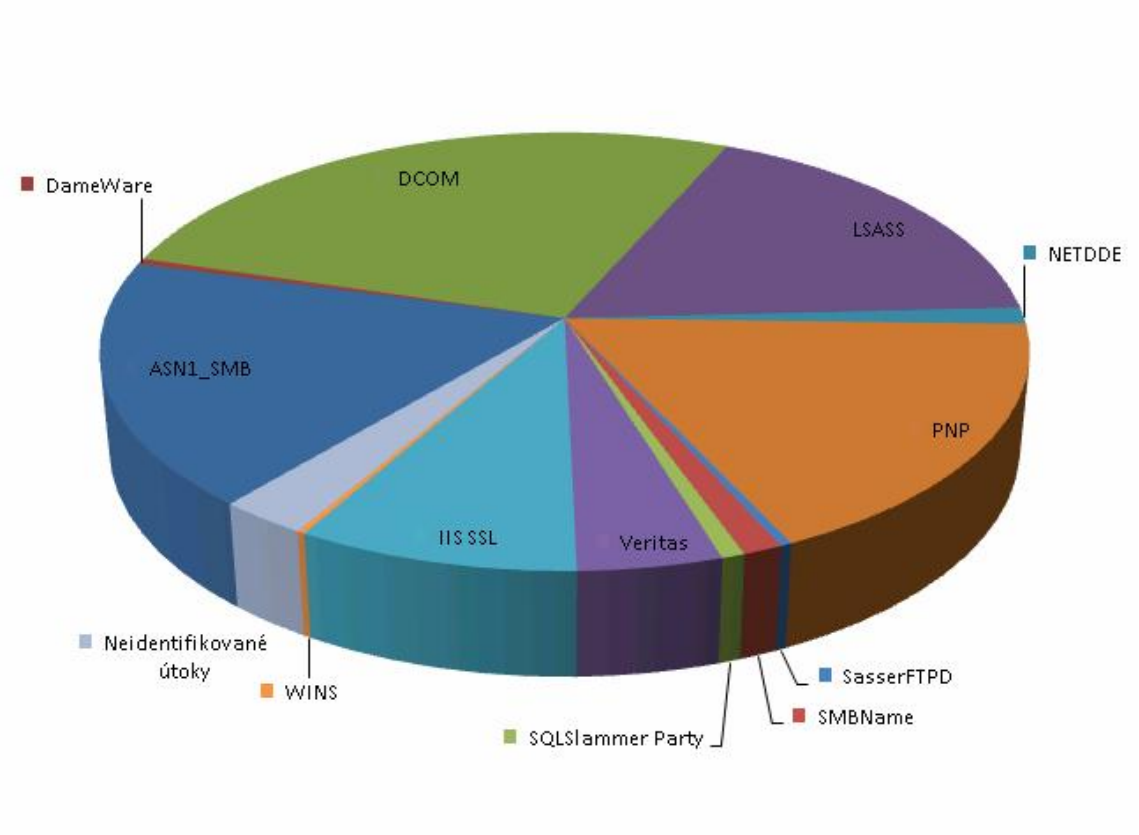


Obr. 20: *Celkový počet spojení (UDP)*

Obrázek 20 ukazuje počet spojení na honeypot pomocí UDP protokolu. Na všech portech většinou naslouchají nepříliš dobře zabezpečené služby Windows (sdílení souboru, messenger , ...)

8.4 Rámcový popis jednotlivých útoků zachycených programem Nepenthes

Informace k jednotlivým útokům jsou blíže popsány v příloze 4. Obrázek 21 zobrazuje poměr zachycených útoků na honeypot programem Nepenthes. Pokud tento poměr útoků srovnáme s analýzou počtu útoků na *www.honeynet.cz* za měsíc březen zjistíme, že hodnoty z projektu a námi testovaného honeypotu příliš nekorespondují. Srovnání hodnot lze nalézt v tabulce 11. Příčin může být několik., např. na testovaný honeypot neexistoval žádný "zajímavý" DNS záznam, který by útočníky mohl upoutat. Hlavním důvodem, proč se hodnoty odlišují, je počet a umístění senzorů, kterým výše uvedený projekt disponuje. Je zřejmé, že počty útoků na jednotlivé senzory v projektu *www.honeynet.cz* se mohou od průměrných hodnot diametrálně lišit.



Obr. 21: *Poměr útoků na honeypot zachycených programem Nepenthes*

Analyzovaný útok	Poměr na testovaném honeypotu	Poměr na projektu honeynet.cz
ASN1_SMB	18,13%	31,11%
DameWare	0,43%	0,79%
DCOM	26,93%	13,92%
LSASS	17,59%	9,07%
NETDDE	1,24%	0,24%
PNP	17,59%	5,90%
SasserFTPD	0,35%	0,68%
SMBName	1,20%	6,90%
SQLSlammer Party	0,70%	11,71%
Veritas	4,38%	2,83%
IIS SSL	8,52%	2,01%
WINS	0,23%	8,65%
Neidentifikované útoky	2,71%	6,19%

Tab. 11: *Srovnání poměrů útoků*

8.5 Útok silou na ssh

V průběhu nasazení honeypotu došlo k velkému množství útoků silou (brutal force attack), nebo slovníkovým útokům na ssh-démona. K nejmasivnějším pokusům o prolomení hesla roota došlo:

13.03.2007 z IP 202.107.217.121 (CN – Zhejiang)
22.03.2007 z IP 66.175.118.170 (US – Florida)
24.03.2007 z IP 193.188.106.202 (Bahrain)
29.03.2007 z IP 212.160.143.210 (PL – Wroclaw)
30.03.2007 z IP 69.13.230.137 (US – California)

03.04.2007 z IP ???

V úterý 3.4.2007 se pomocí těchto útoku podařilo útočníkovi získat práva superuživatele a ovládnout honeypot, který byl okamžitě použit k dalším útokům. Nepenthes svou povahou (honeypot s nižší úrovní interkace, který nefiltruje ssh) nedokázal útok zachytit a nezabránil útočníkovi získat vládu nad celým systémem. Po pokusech získat vládu nad honeypotem zpět, útočník po sobě zametl veškeré stopy, proto se lze jenom domnívat jak útok probíhal a jaké nastaly změny v sytému.

Z níže uvedeného logu (který byl zaslán z ids@cesnet.cz) je patrné, že systém byl použit ke skenování otevřeného ssh portu na 1021 IP adresách. Po tomto skenu by pravděpodobně následoval útok silou nebo slovníkový útok na sshd.

```
Data od `Út  3.04.2007, 12:05:06' do `Út  3.04.2007, 14:05:01'.

1021 pokusů o připojení z 147.229.7.3 (buslab-2.fit.vutbr.cz)

Začátek útoku: 1175606960 = Út  3.04.2007, 13:29:20
1175606960 147.229.7.3 1179 -> 195.113.aaa.2 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.3 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.4 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.5 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.6 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.7 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.8 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.9 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.10 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.11 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.12 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.13 22
1175606960 147.229.7.3 1179 -> 195.113.aaa.14 22
.. vynecháno 995 řádek …
1175606962 147.229.7.3 1179 -> 195.113.aaa.242 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.243 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.244 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.245 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.246 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.247 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.248 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.249 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.250 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.251 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.252 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.253 22
1175606962 147.229.7.3 1179 -> 195.113.aaa.254 22
Konec útoku: 1175606962 = Út  3.04.2007, 13:29:22.
Útok trval: 0:00:02 [h:m:s]. Frekvence: 30630.000 [pokusů/min].
```

Po obnovení systému následovalo několik pokusům o přihlášení na různé účty z IP 89.137.189.2 a 219.235.231.103 viz. zkrácený výpis logu – auth.log

```
Apr  3 18:11:42 tmachin sshd[4567]: Invalid user test from 89.137.189.2
Apr  3 18:11:42 tmachin sshd(pam_unix)[4567]: • ilu pass; user unknown
Apr  3 18:11:42 tmachin sshd(pam_unix)[4567]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2
Apr  3 18:11:44 tmachin sshd[4567]: Failed password for invalid user test from 89.137.189.2 port 57651 ssh2
Apr  3 18:11:44 tmachin sshd[4569]: Invalid user guest from 89.137.189.2
Apr  3 18:11:44 tmachin sshd(pam_unix)[4569]: • ilu pass; user unknown
Apr  3 18:11:44 tmachin sshd(pam_unix)[4569]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2
Apr  3 18:11:46 tmachin sshd[4569]: Failed password for invalid user guest from 89.137.189.2 port 57668 ssh2
Apr  3 18:11:47 tmachin sshd[4571]: Invalid user admin from 89.137.189.2
Apr  3 18:11:47 tmachin sshd(pam_unix)[4571]: • ilu pass; user unknown
Apr  3 18:11:47 tmachin sshd(pam_unix)[4571]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2
Apr  3 18:11:49 tmachin sshd[4571]: Failed password for invalid user admin from 89.137.189.2 port 57680 ssh2
Apr  3 18:11:49 tmachin sshd[4573]: Invalid user admin from 89.137.189.2
Apr  3 18:11:49 tmachin sshd(pam_unix)[4573]: • ilu pass; user unknown
Apr  3 18:11:49 tmachin sshd(pam_unix)[4573]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2
Apr  3 18:11:53 tmachin sshd[4575]: Failed password for invalid user user from 89.137.189.2 port 57755 ssh2
Apr  3 18:11:54 tmachin sshd(pam_unix)[4577]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2  user=root
Apr  3 18:11:56 tmachin sshd[4577]: Failed password for root from 89.137.189.2 port 57772 ssh2
Apr  3 18:11:56 tmachin sshd(pam_unix)[4579]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=89.137.189.2  user=root
…..
Apr  3 22:50:14 tmachin sshd[4617]: Invalid user test from 219.232.59.181
Apr  3 22:50:14 tmachin sshd(pam_unix)[4617]: • ilu pass; user unknown
Apr  3 22:50:14 tmachin sshd(pam_unix)[4617]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.232.59.181
Apr  3 22:50:16 tmachin sshd[4617]: Failed password for invalid user test from 219.232.59.181 port 54710 ssh2
Apr  3 23:09:05 tmachin sshd[4620]: Invalid user test from 219.235.231.103
Apr  3 23:09:05 tmachin sshd(pam_unix)[4620]: • ilu pass; user unknown
Apr  3 23:09:05 tmachin sshd(pam_unix)[4620]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103
Apr  3 23:09:13 tmachin sshd[4622]: Failed password for invalid user guest from 219.235.231.103 port 38491 ssh2
Apr  3 23:09:21 tmachin sshd[4624]: Invalid user admin from 219.235.231.103
Apr  3 23:09:21 tmachin sshd(pam_unix)[4624]: • ilu pass; user unknown
Apr  3 23:09:21 tmachin sshd(pam_unix)[4624]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103
Apr  3 23:09:22 tmachin sshd[4624]: Failed password for invalid user admin from 219.235.231.103 port 38561 ssh2
Apr  3 23:09:27 tmachin sshd[4626]: Invalid user admin from 219.235.231.103
Apr  3 23:09:27 tmachin sshd(pam_unix)[4626]: • ilu pass; user unknown
Apr  3 23:09:27 tmachin sshd(pam_unix)[4626]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103
Apr  3 23:09:35 tmachin sshd[4628]: Failed password for invalid user user from 219.235.231.103 port 38706 ssh2
Apr  3 23:09:39 tmachin sshd(pam_unix)[4630]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103  user=root
Apr  3 23:09:41 tmachin sshd[4630]: Failed password for root from 219.235.231.103 port 38782 ssh2
Apr  3 23:09:46 tmachin sshd(pam_unix)[4632]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103  user=root
Apr  3 23:09:47 tmachin sshd[4632]: Failed password for root from 219.235.231.103 port 38842 ssh2
Apr  3 23:09:51 tmachin sshd(pam_unix)[4634]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103  user=root
Apr  3 23:09:54 tmachin sshd[4634]: Failed password for root from 219.235.231.103 port 38904 ssh2
Apr  3 23:09:58 tmachin sshd[4636]: Invalid user test from 219.235.231.103
Apr  3 23:09:58 tmachin sshd(pam_unix)[4636]: • ilu pass; user unknown
Apr  3 23:09:58 tmachin sshd(pam_unix)[4636]: authentication • ilure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=219.235.231.103
Apr  3 23:10:00 tmachin sshd[4636]: Failed password for invalid user test from 219.235.231.103 port 38977 ssh2
```

Bohužel tento výpis logu jako důkaz, že průnik byl veden z jedné nebo druhé IP, nestačí. Je pravděpodobné, že oba systémy spravuje stejný útočník, který po úspěšném průniku vytvořil výše uvedené účty.

Z události je zřejmé, jak je nebezpečné nechávat nedostatečně zabezpečený stroj volně přístupný z internetu a jak je nutné požívat alespoň základní možnosti zabezpečení vzdáleného přístupu - např. přístup pouze z daných IP adres, omezení přihlašování na administrátorský účet. Bezpochyby také platí, že pokud účty, ke kterým je možné vzdáleně přistupovat, nejsou zabezpečeny robustními hesly, je každé další zabezpečení bez výsledku.

Pozn. *Root100* není robusní heslo.

8.6 Analýza binárních souborů

Za dobu provozování honeypotu bylo staženo 45 binárních souborů, které obsahovaly sedm různých virů/mallware. Jednotlivé soubory byly zaslány k analýze automatickým nástrojem do společnosti http://www.norman.com/microsites/nsic/Submit/en-us, případné výsledky byly porovnány s "hashem" stažených/analyzovaných souborů na ***www.honeynet.cz***. Seznam analyzovaných binárních souborů, lze nalézt v příloze 3.

Seznam stažených virů:

- TR/Crypt.PCMM.Gen 29
- TR/Crypt.PCMM.Gen* 7
- WORM/Sdbot.391168 4
- W32/Parite 1
- BDS/Agent.aew.1 1
- Trojan.Mybot-7508 1
- WORM/Sdbot.391168 1
- BDS/Agent.aew.1 1

*jiný binární soubor

Podrobná analýza stažených binárních souborů je dostupná v příloze 3. Z důvodů uvedených v kapitole 8.4 se ani jeden ze stažených virů/červů neumístil v první desítce detekovaných virů u antivirových společností za měsíc duben.

8.7 Shrnutí

Analýza logů přináší přehled událostí, které se na honeypotu za jeden měsíc staly.

Z informací, které jsme získali je patrné, že nejnebezpečnější byl slovníkový útok na ssh démona, který vedl ke ztrátě kontroly nad celým systémem. Následné útočníkovi akce na kompromitovaném systému mohly odpovídat informacím uvedeným v článku na *http://www.honeynet.cz/?news_id=57*.

Pozoruhodným zjištěním je postupné narůstání počtu spojení na honeypot. Bylo by velmi zajímavé zjistit, proč počty útoků narůstají a kdy se nárůst počtu útoků zastaví. Tyto otázky by si jistě vyžádaly rozsáhlejší výzkum.

Pokud se zamyslíme nad jednotlivými útoky z hlediska míry nebezpečnosti, je velmi těžké určit, které útoky zachycené honeypotem představují větší a které menší hrozbu. Obecně platí, že jakýkoliv i nepovedený útok spotřebovává systémové zdroje, proto je každý útok nežádoucí. Rámcová analýza útoků zachycených honeypotem Nepenthes včetně jejich rozboru je k dispozici v příloze 4. Analýza zachycených binárních souborů se nachází v příloze 3.

9 Závěr

Cílem práce bylo seznámení s jednotlivými typy honeypotů, způsobem jejich činnosti a jejich následné testování. Největší pozornost byla věnována získání a především analýze dat z honeypotu s nižší úrovní interakce, který byl spuštěn na delší časové období.

Z analýzy získaných dat vyplývá, že v drtivé většině stojí za útoky z internetu automatické nástroje, které využívají chyb v operačních systémech (nebo jiném softwaru). Pokud automatický nástroj uspěje s nalezením a proniknutím do nezabezpečené služby, velmi pravděpodobně se na napadnutý systém připojí už živý útočník, který zařízení využije k další nekalé činnosti.

Význam honeypotu spočívá v zachycení a analýze útoků. Především jsme schopni zachytit ještě nepopsaný útok. Začleněním honeypotu do provozní sítě získáme nástroj, jehož smysl spočívá v jeho neautorizovaném využití. Sledování této nástrahy nám umožní analyzovat bezpečnostní incidenty a na základě získaných informací se přiměřeně bránit.

Incident, kdy útočník použil slovníkový útok na démona ssh, nás přesvědčil o nutnosti využívat alespoň základní zabezpečení přístupu ke vzdáleným službám. Bezpochyby také platí, že pokud účty, ke kterým lze vzdáleně přistupovat, nejsou zabezpečeny robustními hesly, je každé další zabezpečení neúčinné.

Velmi zajímavým rozšířením této práce by byla možnost pozorovat určitý segment (virtuální) sítě a vysledovat, jakým způsobem útočníci postupují při vyhledávání zranitelných stanic a služeb. Na základě zjištěných dat by teoreticky bylo možné filtrovat nežádoucí provoz přímo na routeru. Podobný způsob by se pravděpodobně uplatnil i při filtrování samotných útoků na směrovači, tzn. filtrování nežádoucích signatur, IP adres, aj. Na druhou stranu by to znamenalo snížení rychlosti routeru a možnost vytváření chyb při nastavování pravidel a tím komplikace při běžném provozu.

Literatura

[1] Lance Spitzner: Honeypots tracking hackers. Boston, Addison-Wesley, 2003.

[2] Roger A. Grimes: Honeypots for Windows. Apress © 2005

[3] Popis Honeynetu je dostupný na URL: http://project.honeynet.org/papers/honeynet/index.html

[4] Informace k Honeynet Project jsou dostupné na URL: http://www.honeynet.org/

[5] Amir Alsbih: Honeypots – How to seek them out. 5 April 2006, dostupné na URL: http://www.it-observer.com/articles/1101/honeypots_how_seek_them_out/

[6] Informace k hping, jsou dostupné na URL: http://www.hping.org/

[7] Mapping Internet Sensors With Probe Response Attacks, dostupné na URL: http://www.usenix.org/events/sec05/tech/bethencourt/bethencourt_html/index.html

[8] Právní aspekty využívaní honeypotů jsou dostupné na URL: http://www.cryptoworld.info/ casop7/crypto78_05.pdf

[9] Informace k IRC botům, jsou dostupné na URL: http://en.wikipedia.org/wiki/IRC_bot

[10] Přehled o projektu Nepenthes je dostupný na URL: http://nepenthes.mwcollect.org/

[11] Informace k Nepenthes je možné získat na adrese: http://honeyblog.org/junkyard/paper/ collecting-malware-final.pdf

[12] Informace o HoneyD lze nalézt na URL: http://www.honeyd.org/general.php

[13] Informace o Internet Storm Center, jsou dostupné na URL: http://isc.sans.org/

[14] Informace k nástrojům honeynetu, jsou dostupné na URL: http://www.honeynet.org/ tools/cdrom/

[15] Informace k programu VMware, jsou dostupné na URL: http://www.vmware.com/

[16] Manuálové stránky projektu Honeynet jsou dostupne na URL: http://www.honeynet.org/tools/ cdrom/roo/manual-1.1/

[17] Informace k bráně Hoenywall jsou dostupné na URL: http://www.honeynet.org.pk/ honeywall/ eeyore/page3.htm

[18] Informace k programu tcpdump, jsou dostupné na URL: http://www.tcpdump.org/

[19] Informace k operačnímu systému Archlinux, jsou dostupné na URL: http://www.archlinux.org/

[20] Informace k honeypotu Multipot jsou dostupné na URL: http://www.idefense.com/

[21] Vysvětlení pojmů je dostupné na URL: http://www.soom.cz/articles/print.php?aid=316

[22] Informace k jednotlivým útokům jsou dostupné na URL: http://www.actinet.cz/

Seznam příloh

Příloha 1

Ukázky útoků zachycené honeypotem Nepenthes

HTTP Directory Traversal (Windows)
dirtr_shadow.htdump 10000/tcp Directory traversal attack to retreive /etc/shadow via http.

Možný útočník, může využívat tuto bezpečnostní díru ke čtení libovolných souborů na vzdálené stanici s právy http procesu.
/var/lib/nepenthes/nepenthes.0.log

[05032007 19:05:40 info module] Traffic for VERITAS (520 bytes)
[05032007 19:05:40 warn module] Stored Hexdump /var/lib/nepenthes/hexdumps/2fe30f34ab1bc19bc7799bf86cf1bd55.bin (0x0808a280 , 0x00000208).
[05032007 19:05:42 warn module] Unknown VERITAS exploit 520 bytes State 0
[05032007 19:05:42 module] Stored Hexdump /var/lib/nepenthes/hexdumps/2fe30f34ab1bc19bc7799bf86cf1bd55.bin (0x0808a280 , 0x00000208).
[05032007 19:05:43 info module] Traffic for VERITAS (142 bytes)
[05032007 19:05:43 warn module] Stored Hexdump /var/lib/nepenthes/hexdumps/7c6ed9fdb9b2d8b26f4ee274acb31dba.bin (0x08089e00 , 0x0000008e).
[05032007 19:05:44 warn module] Unknown VERITAS exploit 142 bytes State 0
[05032007 19:05:44 module] Stored Hexdump /var/lib/nepenthes/hexdumps/7c6ed9fdb9b2d8b26f4ee274acb31dba.bin (0x08089e00 , 0x0000008e).

/var/lib/nepenthes/hexdumps/
....
-rw-r--r-- 1 root root 520 Mar 6 17:20 2fe30f34ab1bc19bc7799bf86cf1bd55.bin
...

GET
/unauthenticated/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/.
.%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01
/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%0
1/..%01/..%01/..%01/..%01//etc/passwd HTTP/1.1
Host: 147.229.7.3:10000
Pragma: no-cache
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*

Další informace:
http://www.vulnerabilityscanning.com/HTTP-Directory-Traversal-Windows--Test_14229.htm#Scan-HTTP-Directory-Trave

Remote root exploit for a buffer overflow in NetBios NTLMSSP
http://www.microsoft.com/technet/security/bulletin/MS04-007.mspx

[06032007 08:35:05 warn module] Unknown NETDDE exploit 0 bytes State 0
[06032007 08:35:05 module] Ignoring zero-length hexdump.
[06032007 08:35:05 warn module] Unknown SMBName exploit 0 bytes State 0
[06032007 08:35:05 info module] 72 4
[06032007 08:35:05 info module] SMB Session Request 72
D CKFDENECFDEFFCFGEFFCCACACACACACA
[06032007 08:35:15 warn module] Unknown NETDDE exploit 4500 bytes State 1
[06032007 08:35:15 module] Stored Hexdump /var/lib/nepenthes/hexdumps/fb8118c9d7919e3addf0f55bc9d63f05.bin (0x0808b2c0 , 0x00001194).
[06032007 08:35:15 warn module] Unknown SMBName exploit 4 bytes State 1

tcpdump

08:35:05.917923 IP comp1.1514 > buslab-2.fit.vutbr.cz.netbios-ssn: S 529454124:529454124(0) win 64240 <mss 1460,nop,nop,sackOK>
08:35:05.917959 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1514: S 2020218585:2020218585(0) ack 529454125 win 5840 <mss 1460,nop,nop,sackOK>
08:35:05.918118 IP buslab-2.fit.vutbr.cz.32779 > kazi.fit.vutbr.cz.domain: 61549+ PTR? 157.17.229.147.in-addr.arpa. (45)
08:35:05.918182 IP comp1.1514 > buslab-2.fit.vutbr.cz.netbios-ssn: . ack 1 win 64240
08:35:05.918263 IP comp1.1514 > buslab-2.fit.vutbr.cz.netbios-ssn: F 1:1(0) ack 1 win 64240

08:35:05.918508 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1514: F 1:1(0) ack 2 win 5840
08:35:05.918724 IP comp1.1514 > buslab-2.fit.vutbr.cz.netbios-ssn: . ack 2 win 64240
08:35:05.919191 IP kazi.fit.vutbr.cz.domain > buslab-2.fit.vutbr.cz.32779: 61549 1/3/4 (206)
08:35:05.919684 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: S 529512570:529512570(0) win 64240 <mss 1460,nop,nop,sackOK>
08:35:05.919696 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: S 2018789187:2018789187(0) ack 529512571 win 5840 <mss 1460,nop,nop,sackOK>
08:35:05.919912 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: . ack 1 win 64240
08:35:05.919974 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: P 1:73(72) ack 1 win 64240 NBT Session Packet: Session Request
08:35:05.919988 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: . ack 73 win 5840
08:35:05.920245 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: P 1:65(64) ack 73 win 5840 NBT Session Packet: Session Granted
08:35:05.920530 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: P 73:77(4) ack 65 win 64176 NBT Session Packet: Session Message
08:35:05.920958 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: P 77:1537(1460) ack 65 win 64176 NBT Session Packet: Unknown packet type 0xFFData: (41 bytes)
[000] 53 4D 42 72 00 00 00 00 18 53 C8 00 00 00 00 00 SMBr\000\000\000\000 \030S\310\000\000\000\000\000
[010] 00 00 00 00 00 00 00 00 00 37 13 00 00 00 00 00 \000\000\000\000\000\000\000\000 \0007\023\000\000\000\000\000
[020] 62 00 02 50 43 20 4E 45 54   b\000\002PC NE T

08:35:05.920966 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: . ack 1537 win 8760
08:35:05.921090 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: . 1537:2997(1460) ack 65 win 64176 NBT Session Packet: Unknown packet type 0x6FData: (41 bytes)
[000] 20 75 73 65 72 20 31 20 31 20 3E 3E 20 6F 20 26   user 1 1 >> o &
[010] 65 63 68 6F 20 67 65 74 20 77 69 6E 64 6E 73 2E echo get windns.
[020] 65 78 65 20 3E 3E 20 6F 20   exe >> o

08:35:05.921560 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: . 2997:4457(1460) ack 65 win 64176 NBT Session Packet: Unknown packet type 0x43Data: (41 bytes)
[000] 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCC CCCCCCCC
[010] 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCC CCCCCCCC
[020] 43 43 43 43 43 43 43 43 43   CCCCCCCC C

08:35:05.921573 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: . ack 4457 win 14600
08:35:05.921579 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: P 4457:4501(44) ack 65 win 64176 NBT Session Packet: Unknown packet type 0x44Data: (41 bytes)
[000] 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 DDDDDDDD DDDDDDDD
[010] 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 DDDDDDDD DDDDDDDD
[020] 44 44 44 44 44 00 00 00 00   DDDDD\000\000\000 \000

08:35:05.959753 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: . ack 4501 win 14600
08:35:15.989206 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: F 4501:4501(0) ack 65 win 64176
08:35:15.989579 IP buslab-2.fit.vutbr.cz.netbios-ssn > comp1.1515: F 65:65(0) ack 4502 win 14600
08:35:15.989802 IP comp1.1515 > buslab-2.fit.vutbr.cz.netbios-ssn: . ack 66 win 64176

Přetečení bufferu proti Microsft LSASS
http://www.microsoft.com/technet/security/bulletin/ms04-011.mspx

 [05032007 19:38:51 warn dia] Unknown ASN1_SMB Shellcode (Buffer 4291 bytes) (State 1)
[05032007 19:38:51 dia] Stored Hexdump /var/lib/nepenthes/hexdumps/3dbde032431f08e3559fd60de4e1b498.bin (0x0808c4e0 , 0x000010c3).
[05032007 19:38:51 warn module] Unknown PNP Shellcode (Buffer 137 bytes) (State 0)
[05032007 19:38:51 module] Stored Hexdump /var/lib/nepenthes/hexdumps/cf1fd621e305cce07e76c214df9673b3.bin (0x0808a618 , 0x00000089).
[05032007 19:38:51 warn module] Unknown LSASS Shellcode (Buffer 137 bytes) (State 0)
[05032007 19:38:51 module] Stored Hexdump /var/lib/nepenthes/hexdumps/cf1fd621e305cce07e76c214df9673b3.bin (0x0808a1b0 , 0x00000089).
[05032007 19:38:51 warn handler dia] Unknown DCOM Shellcode (Buffer 4 bytes) (State 0)
[05032007 19:38:51 handler dia] Stored Hexdump /var/lib/nepenthes/hexdumps/d760b4f5cca6cc4fe2bed30a6814145d.bin (0x08089da8 , 0x00000004).

tcpdump
19:38:20.300524 arp who-has buslab-2.fit.vutbr.cz tell bd-boz.net.vutbr.cz
19:38:20.300544 arp reply buslab-2.fit.vutbr.cz is-at 00:11:2f:bd:5e:ba (oui Unknown)
19:38:30.957928 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: S 1748370312:1748370312(0) win 65535 <mss 1460,nop,nop,sackOK>
19:38:30.957964 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: S 69263757:69263757(0) ack 1748370313 win 5840 <mss 1460,nop,nop,sackOK>
19:38:30.958176 IP buslab-2.fit.vutbr.cz.32778 > kazi.fit.vutbr.cz.domain: 5348+ PTR? 192.214.229.147.in-addr.arpa. (46)
19:38:30.958311 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: . ack 1 win 65535
19:38:30.959106 IP kazi.fit.vutbr.cz.domain > buslab-2.fit.vutbr.cz.32778: 5348 1/3/3 (192)
19:38:30.959115 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: P 1:5(4) ack 1 win 65535
19:38:30.959125 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 5 win 5840
19:38:30.959462 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: P 5:138(133) ack 1 win 65535
19:38:30.959478 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 138 win 6432

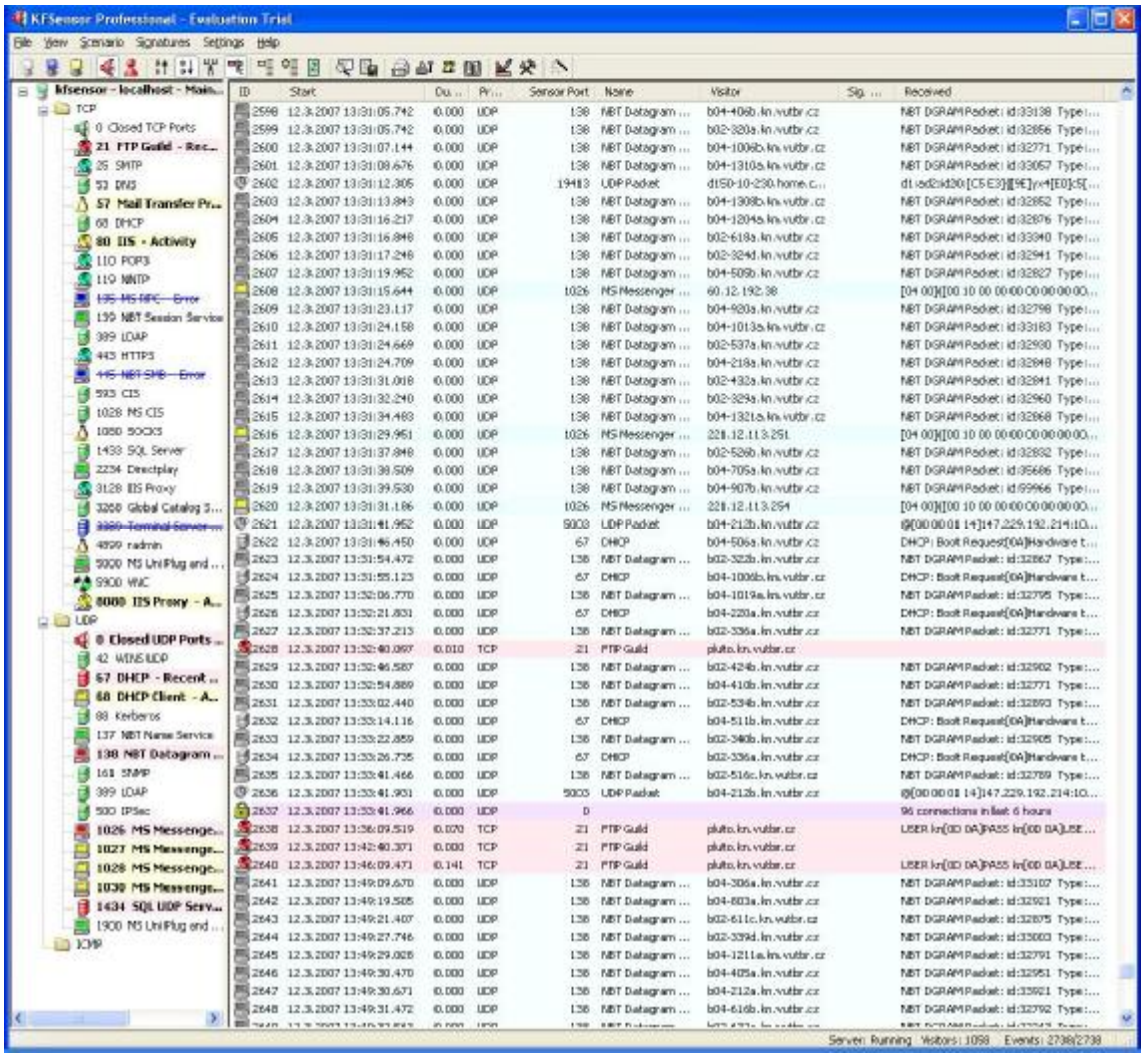
```
19:38:41.475419 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: P 138:142(4) ack 1 win 65535
19:38:41.475434 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 142 win 6432
19:38:41.475806 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: . 142:1602(1460) ack 1 win 65535
19:38:41.475813 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 1602 win 8760
19:38:41.475929 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: . 1602:3062(1460) ack 1 win 65535
19:38:41.475941 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 3062 win 11680
19:38:41.476045 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: P 3062:4429(1367) ack 1 win 65535
19:38:41.476054 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: . ack 4429 win 14600
19:38:51.992188 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: F 4429:4429(0) ack 1 win 65535
19:38:51.992936 IP buslab-2.fit.vutbr.cz.445 > a03-0235a.kn.vutbr.cz.2955: F 1:1(0) ack 4430 win 14600
19:38:51.993235 IP a03-0235a.kn.vutbr.cz.2955 > buslab-2.fit.vutbr.cz.445: . ack 2 win 65535
```

Příloha 2

Log programu KFSensor

Ppokus serveru pluto.kn.vutbr.cz o nalogovaní na lokální ftp server

<event sensorid="kfsensor" id="129" type="Connection" action="SimStdServer" name="FTP Guild" simname="FTP Guild" protocol="TCP" severity="High">
  <start>2007-03-11 13:36:13:987</start>
  <end>2007-03-11 13:36:14:027</end>
  <client domain="pluto.kn.vutbr.cz" ip="147.229.190.133" port="58335" />
  <host ip="147.229.192.46" bindip="" port="21" />
  <connection closedby="Server" />
  <recBytes>46</recBytes>
  <received size="46" coding="kf">
    <![CDATA[USER kn%0D%0A
PASS kn%0D%0A
USER purk%0D%0A
PASS purk%0D%0A
QUIT%0D%0A
]]>
  </received>
  <sentBytes>221</sentBytes>
  <sent size="295" coding="kf">
    <![CDATA[>>>>220-networksforu.com%0D%0A
>>>>220 Please enter your name:%0D%0A
USER kn%0D%0A
>>>>331 User name okay, Need password.%0D%0A
PASS kn%0D%0A
>>>>530 Password not accepted.%0D%0A
USER purk%0D%0A
>>>>331 User name okay, Need password.%0D%0A
PASS purk%0D%0A
>>>>530 Password not accepted.%0D%0A
QUIT%0D%0A
>>>>221 Goodbye. Control connection closed.%0D%0A
]]>
  </sent>
</event>



Obr. 22: *Hlavní okno KFSensor se sledovanými porty a ukázkou událostí*

Příloha 3

Seznam stažených binárních souborů

jméno: dload.exe
MD5: 84491b5246f3dce0a8f270edca6a6c30
velikost: 66845
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 131
první výskyt: 2007-03-19 16:06:08
poslední výskyt: 2007-04-21 01:29:25
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: TR/Crypt.PCMM.Gen
clamav antivirus: Trojan.Small-1671

jméno: dload.exe
MD5: 16a15dd4e6636d38adfcc911577a09ac
velikost: 65968
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 131
první výskyt: 2007-03-11 18:13:15
poslední výskyt: 2007-04-13 00:24:34
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: TR/Crypt.PCMM.Gen
clamav antivirus: Trojan.Small-1671

jméno: dload.exe
MD5: 24229ba136f97ce47bbc3a0018872063
velikost: 68575
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 131
první výskyt: 2007-03-16 01:45:58
poslední výskyt: 2007-04-17 16:56:04
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: TR/Crypt.PCMM.Gen
clamav antivirus: Trojan.Small-1671

jméno: recsl.exe
MD5: e1735e398d3124237c2787e727214219
velikost: 76885
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 58
první výskyt: 2007-01-03 17:12:44
poslední výskyt: 2007-01-05 16:13:53
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: TR/Crypt.PCMM.Gen

jméno: x.exe
MD5: da0354ed776994e44586de4e75364e34
velikost: 189398
operační systém: Windows
protokol použitý pro stažení: http
Botnet ID: 0
první výskyt: 2007-04-03 22:47:58
poslední výskyt: 2007-04-14 23:14:08
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit, UPX compressed
avira antivirus: W32/Parite
clamav antivirus: Worm.Padobot.N
fprot antivirus: W32/Korgo.S

jméno: wulogin.exe

MD5: cad6fa541387f6c2748f31e30df5ccfc velikost: 1259520
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 0
první výskyt: 2006-09-17 19:52:05
poslední výskyt: 2007-04-15 12:38:08
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: BDS/Agent.aew.1
clamav antivirus: Trojan.Agent-683
fprot antivirus: Backdoor.NYI
avast antivirus: Win32:Agent-CAH

jméno: dload.exe
MD5: 0599b3f41ada7225caf40447f2265122
velikost: 391168
operační systém: Windows
protokol použitý pro stažení: ftp
Botnet ID: 0
první výskyt: 2007-03-18 17:00:12
poslední výskyt: 2007-03-24 19:59:21
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: WORM/Sdbot.391168
avast antivirus: Win32:SdBot-gen44

jméno: +uWQEQ==MD5: 96837ddfcc76cc352d67c116343d0194
velikost: 196473
operační systém: Windows
protokol použitý pro stažení: tftp
Botnet ID: 0
první výskyt: 2006-07-25 17:45:40
poslední výskyt: 2007-05-02 16:29:17
unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit
avira antivirus: Backdoor-Server/Hupigon.buw
clamav antivirus: Trojan.Mybot-7508
fprot antivirus: Sdbot.STM
avast antivirus: Win32:Hupigon-NF
bitdefender antivirus: Backdoor.Rbot.AEM
norman antivirus: Spybot.AOIC.dropper

Podrobná analýza jednoho ze stažených souborů

Analyzováno:
http://www.honeynet.cz/?mmenu=malware&smenu_int=0&lang=cz&vmetr=1&part=0&tab=0&order_w=tstamp&order_h=desc&search_e=1&search_name=&search_md5=84491b5246f3dce0a8f270edca6a6c30&search_tdate=&search_ldate=&search_avira=&mal_id=12166

jméno: dload.exe **MD5:** 84491b5246f3dce0a8f270edca6a6c30 **velikost:** 66845

operační systém: Windows **protokol použitý pro stažení:** ftp **Botnet ID:** 131

první výskyt: 2007-03-19 16:06:08 **poslední výskyt:** 2007-04-21 01:29:25

unix file: MS-DOS executable PE for MS Windows (GUI) Intel 80386 32-bit

avira antivirus: TR/Crypt.PCMM.Gen

clamav antivirus: Trojan.Small-1671

cWsandbox:

Analysis Summary:Analysis Date 19.04.2007 18:58:00
Sandbox Version 1.107
Filename 84491b5246f3dce0a8f270edca6a6c30.exe

Technical Details:Analysis Number 1
Parent ID 0
Process ID 520
Filename c:\84491b5246f3dce0a8f270edca6a6c30.exe
Filesize 66845 bytes
MD5 84491b5246f3dce0a8f270edca6a6c30
Start Reason AnalysisTarget

Termination Reason NormalTermination
Start Time 00:00.265
Stop Time 00:19.375
Detection Trojan.Small-1671 (ClamAV)
OK (BDC/Linux-Console)
TR/Crypt.PCMM.Gen] (AntiVir Workstation)
DLL-Handling Loaded DLLs
c:\84491b5246f3dce0a8f270edca6a6c30.exe
C:\WINDOWS\system32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\WS2_32.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\WS2HELP.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\user32.dll
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\oleaut32.dll
C:\WINDOWS\system32\ole32.dll
C:\WINDOWS\system32\comctl32.dll
C:\WINDOWS\system32\wsock32.dll
C:\WINDOWS\system32\pstorec.dll
C:\WINDOWS\system32\ATL.DLL
C:\WINDOWS\system32\Wship6.dll
C:\WINDOWS\system32\Secur32.dll
KERNEL32.DLL
WS2_32.dll
user32.dll
ws2_32.dll
advapi32.dll
kernel32.dll
comctl32.dll
wininet.dll
icmp.dll
netapi32.dll
dnsapi.dll
iphlpapi.dll
mpr.dll
shell32.dll
C:\WINDOWS\system32\odbcint.dll
odbc32.dll

Filesystem New Files
\Device\Tcp
\Device\Ip
\Device\Ip
C:\WINDOWS\system32\svcchosst.exe
Opened Files
\\.\Ip
C:\WINDOWS\explorer.exe
C:\WINDOWS\system32\svcchosst.exe
\SystemRoot\AppPatch\sysmain.sdb
\SystemRoot\AppPatch\systest.sdb
\Device\NamedPipe\ShimViewer
C:\WINDOWS\system32\svcchosst.exe
Chronological order
Create/Open File: \Device\Tcp (OPEN_ALWAYS)
Create/Open File: \Device\Ip (OPEN_ALWAYS)
Create/Open File: \Device\Ip (OPEN_ALWAYS)
Open File: \\.\Ip (OPEN_EXISTING)
Get File Attributes: C:\WINDOWS\system32\svcchosst.exe Flags: (SECURITY_ANONYMOUS)
Copy File: c:\84491b5246f3dce0a8f270edca6a6c30.exe to C:\WINDOWS\system32\svcchosst.exe
Open File: C:\WINDOWS\explorer.exe (OPEN_EXISTING)
Open File: C:\WINDOWS\system32\svcchosst.exe (OPEN_EXISTING)
Set File Time: C:\WINDOWS\system32\svcchosst.exe
Set File Attributes: C:\WINDOWS\system32\svcchosst.exe Flags:
(FILE_ATTRIBUTE_HIDDEN,FILE_ATTRIBUTE_READONLY,FILE_ATTRIBUTE_SYSTEM,SECURITY_ANONYMOUS)
Open File: \SystemRoot\AppPatch\sysmain.sdb (OPEN_EXISTING)
Open File: \SystemRoot\AppPatch\systest.sdb (OPEN_EXISTING)
Open File: \Device\NamedPipe\ShimViewer (OPEN_EXISTING)
Open File: C:\WINDOWS\system32\svcchosst.exe ()
Find File: svcchosst.exe

Mutexes Creates Mutex: lastyboyhot
Registry Reads

HKEY_LOCAL_MACHINE\SYSTEM\WPA\MediaCenter 'Installed'

Process Management Creates Process - Filename (C:\WINDOWS\system32\svcchosst.exe) CommandLine: (C:\WINDOWS\system32\svcchosst.exe 1444 'c:\84491b5246f3dce0a8f270edca6a6c30.exe') As User: () Creation Flags: (DETACHED_PROCESS)
Kill Process - Filename () CommandLine: () Target PID: (520) As User: () Creation Flags: ()
System Info Get System Directory
Network Activity

The following process was started by process: 1 Analysis Number 2
Parent ID 1
Process ID 1720
Filename C:\WINDOWS\system32\svcchosst.exe C:\WINDOWS\system32\svcchosst.exe 1444 c:\84491b5246f3dce0a8f270edca6a6c30.exe
Filesize 66845 bytes
MD5 84491b5246f3dce0a8f270edca6a6c30
Start Reason CreateProcess
Termination Reason Timeout
Start Time 00:16.000
Stop Time 02:00.797
Detection Trojan.Small-1671 (ClamAV)
OK (BDC/Linux-Console)
TR/Crypt.PCMM.Gen] (AntiVir Workstation)
DLL-Handling Loaded DLLs
C:\WINDOWS\system32\svcchosst.exe
C:\WINDOWS\system32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\WS2_32.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\WS2HELP.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\user32.dll
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\oleaut32.dll
C:\WINDOWS\system32\ole32.dll
C:\WINDOWS\system32\comctl32.dll
C:\WINDOWS\system32\wsock32.dll
C:\WINDOWS\system32\pstorec.dll
C:\WINDOWS\system32\ATL.DLL
C:\WINDOWS\system32\Wship6.dll
C:\WINDOWS\system32\Secur32.dll
KERNEL32.DLL
WS2_32.dll
user32.dll
ws2_32.dll
advapi32.dll
kernel32.dll
comctl32.dll
wininet.dll
icmp.dll
netapi32.dll
dnsapi.dll
iphlpapi.dll
mpr.dll
shell32.dll
C:\WINDOWS\system32\odbcint.dll
odbc32.dll
RASAPI32.DLL
RTUTILS.DLL
RASMAN.DLL
secur32.dll
C:\WINDOWS\system32\msv1_0.dll
SHELL32.dll
USERENV.dll
shlwapi.dll

Filesystem New Files
\Device\Tcp
\Device\Ip
\Device\Ip
\Device\RasAcd
Opened Files
\\.\Ip

\\.\PIPE\srvsvc
\\.\PIPE\ROUTER
\\.\PIPE\lsarpc
c:\autoexec.bat
C:\Dokumente und Einstellungen\All Users\Anwendungsdaten\Microsoft\Network\Connections\Pbk\rasphone.pbk
Deleted Files
c:\84491b5246f3dce0a8f270edca6a6c30.exe
Chronological order
Create/Open File: \Device\Tcp (OPEN_ALWAYS)
Create/Open File: \Device\Ip (OPEN_ALWAYS)
Create/Open File: \Device\Ip (OPEN_ALWAYS)
Open File: \\.\Ip (OPEN_EXISTING)
Delete File: c:\84491b5246f3dce0a8f270edca6a6c30.exe
Open File: \\.\PIPE\srvsvc (OPEN_EXISTING)
Open File: \\.\PIPE\ROUTER (OPEN_EXISTING)
Open File: \\.\PIPE\lsarpc (OPEN_EXISTING)
Get File Attributes: c:\autoexec.bat Flags: (SECURITY_ANONYMOUS)
Open File: c:\autoexec.bat (OPEN_EXISTING)
Find File: C:\Dokumente und Einstellungen\All Users\Anwendungsdaten\Microsoft\Network\Connections\Pbk*.pbk
Find File: C:\Dokumente und Einstellungen\All Users\Anwendungsdaten\Microsoft\Network\Connections\Pbk\rasphone.pbk
Open File: C:\Dokumente und Einstellungen\All Users\Anwendungsdaten\Microsoft\Network\Connections\Pbk\rasphone.pbk
(OPEN_EXISTING)
Find File: C:\WINDOWS\system32\Ras*.pbk
Find File: C:\Dokumente und Einstellungen\nepenthes\Anwendungsdaten\Microsoft\Network\Connections\Pbk*.pbk
Create/Open File: \Device\RasAcd (OPEN_ALWAYS)

Mutexes Creates Mutex: lastyboyhot
Creates Mutex: RasPbFile
Network Shares Delete Share - Host: () Network Ressource: (IPC$?_CHAR(0x04)_;) Filename: () As User: ()
Delete Share - Host: () Network Ressource: (ADMI?_CHAR(0x01)_;) Filename: () As User: ()
Delete Share - Host: () Network Ressource: (C$) Filename: () As User: ()
Enum Network Shares - Network Ressource: () Host: ()
Registry Changes
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run 'msvccc66' = svcchosst.exe
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices 'msvccc66' = svcchosst.exe
HKEY_LOCAL_MACHINE\Software\Microsoft\OLE 'EnableDCOM' = N
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa 'restrictanonymous' = [REG_DWORD, value: 00000001]
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa 'restrictanonymoussam' = [REG_DWORD, value:
00000001]
Reads
HKEY_LOCAL_MACHINE\Software\Microsoft\Rpc\SecurityService '10'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurityProviders 'SecurityProviders'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'Name'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'Comment'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'Capabilities'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'RpcId'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'Version'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'Type'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msapsspc.dll 'TokenSize'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'Name'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'Comment'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'Capabilities'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'RpcId'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'Version'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'Type'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\digest.dll 'TokenSize'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'Name'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'Comment'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'Capabilities'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'RpcId'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'Version'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'Type'
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SspiCache\msnsspc.dll 'TokenSize'
HKEY_LOCAL_MACHINE\Software\Microsoft\Rpc\SecurityService 'DefaultAuthLevel'

Service Management Open Service Manager - Name: 'SCM'
Open Service - Name: 'RASMAN'
System Info Get System Directory
Get Computer Name
Get System Time
User Management Impersonate User - Domain: () User: (nepenthes)
Window Find Window - Class Name (mIRC) Window Name ()
Network Activity DNS Lookup
Host Name IP Address
llasty.dsaku72830.info 81.95.148.234

C&C Server: 81.95.148.234:16666
Server Password:
Username: DEU|780945
Nickname: DEU|780945

Analysis Number 3
Parent ID 0
Process ID 664
Filename services.exe
Filesize 108544 bytes
MD5 edb6b81761bd60f32f740bbc40afb676
Start Reason SCM
Termination Reason Timeout
Start Time 00:29.984
Stop Time 02:00.968

Příloha 4

Popis jednotlivých útoků

ASN1_SMB	Počet útoků	249
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	445, 80
	Další informace	http://www.microsoft.com/technet/security/bulletin/ms04-007.mspx
DameWare	Počet útoků	5
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	6129, 445
	Další informace	V serverové aplikaci DameWare Mini Remote Control určené ke vzdálené správě systému, zasláním upravených paketů serveru poslouchajícímu na portu TCP 6129 může útočník docílit přetečení bufferů a spuštění libovolného kódu.
DCOM	Počet útoků	695
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	135
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS03-039.mspx http://support.microsoft.com/kb/823980/ http://www.microsoft.com/technet/security/bulletin/MS04-012.mspx
NETDDE	Počet útoků	16
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	139
	Další informace	http://www.microsoft.com/technet/security/Bulletin/MS04-031.mspx
LSASS	Počet útoků	227
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	445
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS04-011.mspx

PNP	Počet útoků	219
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	445,5000
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS01-059.mspx
SasserFTPD	Počet útoků	4
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	445, 1023, 5000, 5454
	Další informace	http://www.viry.cz/go.php?p=viry&t=novinka&id=2355 http://www.microsoft.com/technet/security/bulletin/ms04-011.mspx
SMBName	Počet útoků	16
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	445, 1023, 5000, 5454
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS04-045.mspx
SQLSlammer Party	Počet útoků	9
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	1433
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS02-039.mspx http://www.avast.com/cze/win32sqlslammer.html
Veritas	Počet útoků	56
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	10000
	Další informace	Ve Veritas Backup Exec Remote Agent byly nalezeny chyby způsobující Buffer Overflow, které mohou způsobit pád systému nebo dovolují spuštění libovolného kódu útočníkem. Další chyba v této aplikaci umožňuje útočníkovi získat administrátorská práva k postiženému systému. A nakonec DoS vyvolaný na NetBackup pro NetWare Media Servers může způsobit pád systému[4]. Podrobnosti o postižených verzích a aktualizacích naleznete v jednotlivých oznámeních výrobce. http://www.actinet.cz/bezpecnost_informacnich_technologii/l8/n92/st9/j1/newsletter.htm
IIS SSL	Počet útoků	110
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	433
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS03-007.mspx http://www.microsoft.com/technet/security/bulletin/MS03-

		051.mspx http://www.microsoft.com/technet/security/bulletin/MS04-011.mspx
Wins	Počet útoků	3
	Výsledek útoku	Vzdálené spuštění kódu
	Obvyklý port	42
	Další informace	http://www.microsoft.com/technet/security/bulletin/MS04-006.mspx http://www.microsoft.com/technet/security/Bulletin/MS04-045.mspx

Neidentifikované útoky	Počet útoků	70
	Výsledek útoku	???
	Obvyklý port	Různé

Další informace k útokům jsou uvedny např. na [22].