

Systemy detekce neautorizovaného průniku do informačního systému firemní sítě Fatra, a.s.

Systems of detection of unauthorized intrusion into information
system and network corporate Fatra, a.s.

Bc. Jaroslav Sup

Diplomová práce
2011



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2010/2011

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jaroslav SUP**
Osobní číslo: **A09721**
Studijní program: **N 3902 Inženýrská informatika**
Studijní obor: **Informační technologie**

Téma práce: **Systémy detekce neautorizovaného průniku do
informačního systému firemní sítě Fatra, a.s.**

Zásady pro vypracování:

1. Provedte literární rešerši na zvolené téma.
2. Analyzujte současný stav firemní sítě a vhodnost realizace IDS a Honey pot systémů.
3. Formou projektu navrhnete vhodné řešení.
4. Provedte implementaci a její otestování v provozu.
5. Provedte diskusi nad zvoleným řešením.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **BEALE, Jay** , et al. **Snort 2.1 intrusion detection. 2, ilustrované vydání. [s.l.] :** Syngress, 2004. 716 s. ISBN 1931836043.
2. **SCHULTZ, Eugene; MELLANDER , Jim; ENDORF , Carl.** Hacking ? detekce a prevence počítačového útoku. Praha : Grada Publishing, 2005. 355 s. ISBN 80-247-1035-8.
3. **Harrisová S., harper A., Eagle Ch., Ness.** Hacking - manuál hackera : podrobnější popis. Praha : Grada Publishing, 2008. 399 s. ISBN 9788024713465.
4. **UR REHMAN, Rafeeq.** Intrusion Detection Systems with Snort. Mary Sudul; Jill Harry. [s.l.] : [s.n.], c2003. 275 s. ISBN 0-13-140733-3.
5. **PROVOS, Niels; HOLZ, Thorsten .** Virtual Honeypots : From Botnet Tracking to Intrusion Detection. [s.l.] : Addison-Wesley Professional, 2008. 440 s. ISBN 0321336321.
6. **SPITZNER , Lance .** Honeypots : Tracking Hackers. USA : Addison-Wesley , 2002. 480 s. ISBN 0321108957.
7. **Know Your Enemy : Learning about Security Threats. 2nd Edition. [s.l.] :** Addison-Wesley Professional, 2004. 800 s. ISBN 10:0-321-16646-9.
8. **DUNNIGAN, James.** Bojiště zítřka : tváří v tvář globální hrozbě kybernetického terorismu. [s.l.] : Baronet Publishers, 2004. 356 s. ISBN 8072146424, 9788072146420.

Vedoucí diplomové práce:

doc. Mgr. Roman Jašek, Ph.D.

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

24. února 2011

Termín odevzdání diplomové práce:

18. května 2011

Ve Zlíně dne 24. února 2011

prof. Ing. Vladimír Vašek, CSc.
děkan



doc. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

ABSTRAKT

Obsahem této diplomové práce je oblast informačně bezpečnostních technologií, přesněji „detekce a prevence“ neautorizovaného průniku do lokálních sítí a pastí. Projekt si klade za cíl vytvořit systém pro odhalení nových, zatím neidentifikovaných hrozeb a pokusů o narušení či zneužití informačních zdrojů. Systém bude schopen analyzovat, upozornit na případné incidenty, vytvářet pravidla a odvádět útočnickovu pozornost od reálné sítě.

Očekávaným přínosem je zvýšení současného bezpečnostního standardu informačních a komunikačních technologií v podnikové síti Fatra, a.s.

Koncepce projektu je založena výhradně na software spadající do skupiny GNU General Public Licence.

Klíčová slova: Detekce, prevence, past, průnik, incident, honeypot, analýza, snort, honeyd

ABSTRACT

The content of this thesis is the area of information the security technologies, more "detection and prevention" unauthorized intrusion into local networks and traps. The project is a system for detecting new, yet unidentified threats and intrusion attempts or misuse of information resources. The system will be able to analyze incidents and create rules to route the attack away from the real network.

The expected benefit is to improve the safety standard of information and communication technologies in the corporate network Fatra, a.s.

The project concept is based only on the software group of the GNU General Public License.

Keywords: Detection, prevention, trap, intrusion, incident, honeypot, analysis, snort, honeyd

Poděkování bych chtěl věnovat panu doc. Mgr. Romanovi Jaškovi, Ph.D. za vedení a korekturu této práce, svému nejbližšímu okolí, pracovnímu kolektivu a především mé rodině za oddanou trpělivost a pochopení mého studijního úsilí.

Prohlašuji, že tato diplomová práce je mým autorským dílem na kterém jsem pracoval samostatně. Veškerou literaturu a prameny ze kterých jsem čerpal při zpracování řádně cituji a uvádím odkazy na příslušné zdroje.

Ve Zlíně

.....
Podpis diplomanta

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 STRUKTURA PRÁCE	11
1.1 STANOVENÍ CÍLŮ	12
2 ŠKODLIVÝ KÓD A JEHO STRATEGIE	13
2.1 MALWARE	13
2.2 TYPY ÚTOKŮ	14
2.3 SOCIÁLNÍ INŽENÝRSTVÍ	15
3 ÚVOD DO PROBLEMATIKY IDS/IPS	16
3.1 IDS (INTRUSION DETECTION SYSTEM)	16
3.1.1 HIDS (Host Based Intrusion Detection System)	17
3.1.2 NIDS (Network Based Intrusion Detection System).....	17
3.2 IPS (INTRUSION PREVENTION SYSTEM).....	18
3.2.1 HIPS (Host Based Intrusion Prevention System).....	18
3.2.2 NIPS (Network Based Intrusion Prevention System).....	19
3.3 ROZDÍL MEZI IDS A IPS.....	19
3.4 DETEKČNÍ SONDY	20
3.4.1 Síťová sonda.....	20
3.4.2 Uzlová sonda.....	20
4 CO JSOU HONEYPOTY ?	21
4.1 ÚČEL HONEYPOTŮ	21
4.2 CO JE HONEYNET ?	21
4.3 ROZDĚLENÍ HONEYPOTŮ.....	22
4.3.1 Serverové honeypoty	22
4.3.2 Klientské honeypoty	22
4.3.3 Nenativní honeypoty	23
4.3.4 Nativní honeypoty	23
5 ANALÝZA ZABEZPEČENÍ SÍTĚ VE SPOLEČNOSTI FATRA, A.S.	25
5.1 BEZPEČNOSTNÍ SLOŽKY V SÍTI FATRA, A.S.	25
5.2 SHRNUÍ A NEDOSTATKY.....	28
6 NÁVRH ŘEŠENÍ	29
6.1 VIRTUALIZACE	29
6.2 OPERAČNÍ SYSTÉM LINUX UBUNTU.....	30
6.3 IDS SNORT	31
6.3.1 Rozbor aplikace Snort	31
6.3.2 Definice jednotek Snort.....	32
6.3.3 Analytické nástroje logovacích souborů Snort.....	33

6.4	HONEYPOT HONEYD.....	34
6.4.1	Proč použít Honeyd ?	34
6.4.2	Architektura Honeyd	35
II	PRAKTICKÁ ČÁST	38
7	IMPLEMENTACE SNORT	39
7.1	INSTALACE LINUX UBUNTU SERVER	39
7.1.1	Postup instalace	39
7.1.2	Definice pojmů	40
7.2	POSTUP INSTALACE A KONFIGURACE INTERNÍ SONDY SNORT 2.8.....	40
7.2.1	Konfigurace HW operačního systému.....	41
7.2.2	Snort.conf	42
7.2.3	Snort	42
7.2.4	MySQL.....	42
7.2.5	BASE a ABODB	43
7.2.6	PHP a Apache2.....	43
7.2.7	Barnyard	44
7.2.8	Testování Snort 2.8	45
7.2.9	Export virtual appliance	45
7.3	KLONOVÁNÍ INTERNÍ SONDY SNORT 2.8	45
7.3.1	Import virtual appliance	45
7.3.2	Konfigurace HW operačního systému.....	45
7.4	POSTUP INSTALACE EXTERNÍ SONDY SNORT 2.9 A SNORT REPORT	46
7.4.1	Konfigurace HW operačního systému.....	47
7.4.2	JpGraph	47
7.4.3	Snort Report	47
7.4.4	DAQ	48
7.4.5	Libdnet	48
7.4.6	Instalace Snort	48
7.4.7	MySQL.....	49
7.4.8	Snort.conf	49
7.4.9	Barnyard2	49
7.4.10	Snort.conf	50
7.4.11	Testování Snort 2.9	51
7.4.12	Automatické spuštění Snort 2.9	51
7.4.13	Nastavení výjimek.....	52
7.4.14	Příklad nastavení výjimky	52
7.4.15	Vytváření pravidel	52
7.4.16	Logický pohled na pravidlo Snortu	53
7.4.17	Definice adres.....	53
7.4.18	Definice portů.....	53
7.4.19	Směr toku dat	54
7.4.20	Nastavení rolí	54
7.4.21	Příklad vytvoření pravidla	54
7.4.22	Automatická aktualizace Oinkmaster.....	55
8	IMPLEMENTACE HONEYPOTU HONEYD	56

8.1	KONFIGURACE SYSTÉMU	56
8.1.1	Kompilace Honeyd 1.5c	57
8.1.2	Předkompilovaná verze - Honeyd Kit	57
8.1.3	APT balíček Honeyd	58
8.1.4	Nastavení emulace síťové struktury a služeb v souboru „honeyd.conf“	59
8.1.5	Spuštění a test démona Honeyd (verze „APT balíček“)..	61
8.2	UTILITY PRO HONEYD	61
8.2.1	Honeydstats	61
8.2.2	Honeydsum.....	62
8.2.3	Testování Honeyd v provozu	65
8.2.4	Automatické reporty.....	66
9	DISKUZE	67
9.1.1	Zhodnocení praktické části.....	67
9.1.2	Řešené problémy a nedostatky	68
	ZÁVĚR	69
	ZÁVĚR V ANGLIČTINĚ.....	70
	SEZNAM POUŽITÉ LITERATURY.....	71
	SEZNAM INTERNETOVÝCH ZDROJŮ	72
	OSTATNÍ ZDROJE	73
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	74
	SEZNAM OBRÁZKŮ	76
	SEZNAM TABULEK.....	77
	SEZNAM PŘÍLOH.....	78

ÚVOD

Jedním ze symbolů současné moderní společnosti jsou informační a komunikační technologie, díky kterým dosahujeme lepších technických i hospodářských výsledků ve všech oborech lidské činnosti. Dostupnost internetových komunikací nám umožňuje budovat a udržovat dnešní svět takový jaký je. Vytváří naprosto ojedinělé a nové příležitosti uplatnění lidského potenciálu, umožňuje rychlejší přenos znalostí a vědeckých poznatků do praxe. Obecně lze říci že chod naší společnosti je závislý na komunikacích či sdílení zdrojů. Kolaps těchto systémů si nemůžeme dovolit neboť omezení schopnosti komunikovat znamená ohrožení chodu procesů a ekonomické ztráty.

Přímo úměrně s počtem sdílených internetových zdrojů se současně zvyšuje kyberkriminalita. Četnost incidentů nabývá na rozměrech a dochází k masovému skenování sítí a prolamováním hesel pomocí slovníkových útoků. Nezabezpečené systémy způsobují zneužití důvěrných dat či jejich neoprávněné manipulaci, zneužití identit, hesel, klíčů či platebních karet.

V agresivním prostředí internetu dochází k narušení datové bezpečnosti často. Příčinou jsou nedostatečná opatření v souvislosti s neznalostí problematiky. Tyto hrozby lze eliminovat nastavením souboru bezpečnostních pravidel a postupů. Základem je důsledná osvěta uživatelů, omezení sociálního inženýrství, vytvoření vhodné architektury síťového prostředí, údržba systémů, stanovení politik, šifrování služeb, komunikace na zabezpečených portech a používání silných hesel. Techniky vetřelců se neustále zdokonalují a proto musíme počítat s rafinovanějšími útoky. Je nutno tomuto trendu přizpůsobit naše obranné mechanismy a sledovat trendy, vytvářet a dodržovat komplexní bezpečnostní strategie neboť špatně udržovaný systém je snadnou kořistí a současně větším rizikem pro celou společnost.

Úkolem této práce je identifikace známých i neznámých rizik na nejvyšších síťových vrstvách, snížení pravděpodobnosti průniku do lokálních sítí a metoda jak problému čelit.

Pokusme se tedy věnovat tématu více pozornosti, neboť „šťěstí přeje připraveným“.

I. TEORETICKÁ ČÁST

1 STRUKTURA PRÁCE

V tomto pojednání budou objasněny terminologie z oblasti škodlivých kódů, detekčních systémů, dále otázka bezpečnosti v podnikové síti Fatra, a.s., možnosti zlepšení a téma monitorování, detekce průniku a pastí. Jádrem projektu jsou podrobné postupy, instalace a popis integrace detekčního systému a pastí do bezpečnostní infrastruktury sítě Fatra, a.s. Na závěr je provedena analýza incidentů s přehledy možných rizik a grafickými výstupy, tvorba statistik a pravidel pro Intrusion Detection System.

- Ve 2.kapitole se seznámíme s typy možných rizik, se kterými se budeme v tomto tématu potýkat.
- V 3.kapitole se seznámíme s problematikou a terminologiemi IDS/IPS, prezentace mechanismů, které jsou schopny analyzovat a vyhodnocovat datový tok.
- V 4. kapitole se zaměříme na technologii „honeypot“, což je koncept nástrah na internetové útočníky. Honeypoty slouží pro zmatení a odvrácení pozornosti od reálné sítě a získání informací o něm samotném.
- V 5. kapitole bude provedena analýza stavu komunikační bezpečnosti v podnikové síti Fatra, a.s., odhalení nedostatků vnější i vnitřní části podnikové sítě a možnosti zlepšení.
- V 6. kapitole je návrh řešení na zvýšení bezpečnostního potenciálu informačních systémů vůči vnějším i vnitřním útokům. Koncept kontroly a vyhodnocení komunikace nad úrovní střeženou firewallem¹. Očekávaný přínos pro správce systému a pro společnost Fatra, a.s.
- V 7. a 8. kapitole se budeme zabývat konfigurací IDS „Snort“ a honeypotu „Honeyd“, bude podrobně popsán praktický postup při integraci systému do podnikové sítě Fatra, a.s.

¹ Firewall - síťový prvek s pravidly pro komunikaci mezi sítěmi s různou mírou bezpečnosti. Zdroj: [http://cs.wikipedia.org/wiki/Firewall].

- V 9. kapitole zhodnotíme předchozí kroky a splnění stanovených cílů, provedeme diskuzi o úspěšnosti a přínosu projektu.

1.1 Stanovení cílů

- Vypracování studie na téma IDS/IPS a honeypots.
- Oživení a umístění NIDS senzorů v demilitarizované zóně a před firewallem.
- Integrace uzlové NIDS sondy v lokální síti.
- Implementace honeypotů před firewallem, v demilitarizované zóně a v lokální síti.
- Emulace virtuálních podsítí s aktivními prvky a počítači různých operačních systémů.
- Zprovoznění podpůrných utilit pro generování reportů z detekčních systémů.
- Zajištění automatické aktualizace NIDS Snort.
- Automatické generování výstupu Honeyd.

2 ŠKODLIVÝ KÓD A JEHO STRATEGIE

Pokud bychom se měli zamýšlet nad důvody proč někdo věnuje čas a energii tvorbě zákeřností, dospěli bychom ke slovům „peníze a moc“. Ovládnutí přístupů k cenným a důvěrným informacím jsou hlavním důvodem tvorby škodlivého kódu. Útočníci mají snahu vyloučit z uživatelů citlivé data a využívají k tomu systémových zranitelností nebo sociálního inženýrství. Dobrým příkladem můžou být podvržené stránky internetového bankovníctví, spam, keyloggery² zachycující hesla nebo špatně záplatovaný systém. ^[12]

Než se začneme zabývat kontrolou bezpečnosti a vytvářet obrannou koncepci, je vhodné znát kategorie a metody útoků. Popíšeme si jakými trendy se tvůrci zákeřného software v současnosti orientují a určíme typy útoků na počítačové systémy.

2.1 Malware

Malware ^[1] je obecný výraz pro veškeré škodlivé kódy vytvořené k získání neoprávněného přístupu do počítačového systému bez vědomí majitele. Je složen zkrácením slov „malicious (zákeřný)“ a software. Útočníci využívají různé metody k oklamání uživatele a prolomení slabin v systému. Podle současných statistik denně vzniká 30 až 40 tisíc nových variant malware, přitom původci se těžko stopují. Příkladem můžou být exploits³ z Číny, které využívají ke spuštění skriptů servery z Austrálie přičemž napadají zdroje v Americe.

- **Červi** - programy ze schopností vlastní replikace bez asistence dalšího softwaru nebo uživatele. Šíří se pomocí chyb aplikací a komunikačních kanálů např. ICQ.
- **Viry** - aktivují se spuštěním hostitelského infikovaného souboru s infiltrací škodlivého kódu. Zdánlivě bezpečná aplikace umožňuje virům přístup do paměti, vykonání vlastní replikace a další šíření.

² Keylogger - hardware nebo software, který snímá stisky jednotlivých kláves. Zdroj: [<http://cs.wikipedia.org/wiki/Keylogger>].

³ Exploit - speciální program, data nebo sekvence příkazů, které využívají programátorskou chybu. Obvykle se jedná o ovládnutí počítače nebo nežádoucí instalaci software, která dále provádí činnost, o které uživatel počítače neví (např. nějaký druh malware). Zdroj: [<http://cs.wikipedia.org/wiki/Exploit>].

- **Trojské koně** – tváří se jako užitečné programy, které po spuštění provedou v hostitelském počítači záškodnickou akci.
- **Zhoubný mobilní kód** – programy stažené ze vzdálených uzlů za účelem interakce s webovými servery.
- **Backdoory** – Programy, které obchází bezpečnostní mechanismy.
- **Uživatelské rootkity** – programy, které nahrazují nebo mění aplikace spouštěné pod uživatelskými oprávněními.
- **Jádrové rootkity** – programy, které nepozorovaně změní operační systém.

2.2 Typy útoků

- **Neautorizované skenování** - cílená akce mapování dostupných portů u veřejně přístupných adres. Jsou k tomu používány nástroje jako Nmap⁴, Ettercap⁵, atd. Sběrem informací hacker odhalí potenciální slabiny a poté provede pokus o průnik.
- **Průnik pomocí směrování portů** - napadení a ovládnutí veřejně přístupného serveru v demilitarizované zóně. Dalším krokem je nastavení provozu z vnější strany firewallu dovnitř sítě na přesměrovaných portech.
- **Slovníkový útok** - lze použít u nezabezpečených služeb FTP, HTTP nebo Telnet. **Man-in-the-Middle** ^[19] - metoda, kdy je útočník v pozici prostředníka mezi uživatelem a serverem. Jako příklad můžeme uvést změněné záznamy na Proxy serveru nebo v DNS, kdy je oběť směrována na podvržené stránky hackera.
- **DoS, DdoS** ^[18] - (Denial of Service, Distributed Denial of Service) jsou útoky typu odepření služby serverem z důvodu přetížení, zahlcení požadavky. Důsledkem

⁴ Nmap ("Network Mapper") - open source nástroj pro zkoumání nebo bezpečnostní auditu v síti. Správcům sítě poskytuje užitečné informace o běžících službách, operačních systémech. Je často využíván pro správu přehledů, sledování hostitelů a služeb v síti. Zdroj: [<http://cs.wikipedia.org/wiki/Nmap>].

⁵ Ettercap - významný sniffovací nástroj, obsahující množství propracovaných v některých případech těžko odhalitelných sniff metodik. Často se používá k útokům typu "man in the midle", pro odposlech komunikace u protokolů (http, https icq, pop3, nfs, ssh1, telnet, ftp, atd.). Zdroj: [<http://www.abclinuxu.cz>].

takového útoku je nedostupnost internetových služeb. Rozdíl mezi metodami je v počtu hráčů. Zatímco DoS je spouštěn z jednoho PC, DDoS je hromadná akce prováděná zotročenými stanicemi „zombie“.

- **Zero-day-exploit** ^[1] – útok který využívá chyby v programu pro vytvoření zranitelnosti systému. Zero-day-exploit je neznámá hrozba na kterou ještě neexistuje ochrana.

2.3 Sociální inženýrství

Úvod této kapitoly asi nejlépe vystihne známý citát: "Jen dvě věci jsou nekonečné - vesmír a lidská hloupost. Tím prvním si ovšem nejsem tak jist" Albert Einstein.

Sociální inženýrství je považováno za jakési umění přimět lidi, aby plnili Vaše přání. Ovládním psychologických triků na oprávněné uživatele systému lze získat jejich přístupy. Obecně se jedná o zneužití nejslabšího článku, postavené na lidské důvěřivosti a hlouposti. Na světě neexistuje žádný počítačový systém, který by nebyl závislý na lidech. Tato bezpečnostní slabina je tedy univerzální, nezávislá na platformě, síti nebo vybavení.^[5]

Je to fenomén, který se v počítačovém světě vyskytuje odedávna ve všech podobách. Sociální inženýrství je nechvalně známým pomocník při šíření virů, poplašných zpráv a obecně kybernetické nákazy. Uživatelé jsou nepoučitelní, nedbají varování a neustále podceňují riziko, neprověřují obsah příloh, pravost www stránek nebo formulářů při vkládání citlivých dat. Velký vliv mají sociální sítě s hojností důvěřivců. Nabízením falešných skutečností nebo podvržením obsahu dat si novodobí kriminálníci přijdou na slušný výdělek. Zarážející je především bezmezná důvěra sdílení informací a odhalování identity.

Obranou je dobře zvládnutá bezpečnostní politika ve firemním prostředí. Vše začíná důsledným školením a informovaností uživatelů jak bezpečně používat firemní zdroje a prostředky, jak nakládat s daty a technikou. Sepsáním závazných bezpečnostních směrnic dojde k upozornění uživatelů na možná rizika. Nastavením striktních pravidel jako jsou www filtry, šifrování komunikace, intervaly na změnu hesla, počet znaků, sílu hesla. Jmenované patří mezi základní prvky obrany, kterou lze obohatit o detekční systémy IDS/IPS, viz. následující kapitoly.^[5]

3 ÚVOD DO PROBLEMATIKY IDS/IPS

V prvních obranných liniích podnikové sítě se obvykle nachází standardní firewall, který čte ve svých definicích, propustí pouze žádaný provoz a ostatní pakety zahodí. Je to v podstatě jednoduché zařízení, které nezkoumá obsah komunikace, pouze chrání podnikovou síť na transportní vrstvě. Pokud jsou limitující pravidla dodržována, dosáhneme na této vrstvě referenčního modelu ISO/OSI⁶ vysoké úrovně bezpečnosti. Přesto však dochází ke zneužití služeb a neoprávněnému průniku do sítě. Důvodem je charakter prováděných útoků. Odehrávají se mimo rámec rozlišovacích schopností firewallu. Je proto potřeba jít ještě dále a nahlížet do struktury proudících dat. K tomuto účelu jsou vyvinuty systémy IDS/IPS. ^[13]

Abychom byli schopni nebezpečí rozpoznat, musíme do síťové struktury začlenit detekční sondy a systém schopný kontrolovat datový tok. Oproti firewallu patří detekční a prevenční systémy mezi ofenzivní mechanismy, které umí rozebrat pakety ve sledovaném síťovém uzlu. Provádí inspekci paketů, zkoumají je, testují, porovnávají a vyhodnocují. Využívají přitom kombinaci technologie signatur a dekódování protokolů. Snaží se hledat podobnosti ve znakových sekvencích datového toku a detekují jejich stav. Probíhá dekódování a defragmentace paketů. Tyto metody umožňují nalézt útoky a správně je rozlišovat od falešných poplachů. Signatury musí být automaticky aktualizovány, aby se minimalizoval počet falešných hlášení. ^[17]

3.1 IDS (Intrusion Detection System)

IDS ^[1] můžeme definovat jako soubor nástrojů a pravidel, který je schopen detekovat nechtěné a nežádoucí změny v systémech pomocí pravidel a signatur (unikátní sekvence znaků). Tyto systémy pasivně sbírají data a informují administrátora o nebezpečí. Bývají součástí jiného bezpečnostního zařízení a pracují jako alarm systém potenciálního rizika.

⁶ ISO/OSI – standardizovaný síťový model zpracovaný organizací ISO. Úlohou referenčního modelu je poskytnout základnu pro vypracování norem pro účely propojování systémů. Zdroj: [http://cs.wikipedia.org/wiki/Referen%C4%8Dn%C3%AD_model_ISO/OSI].

Připojují se k síťovému rozbočovači HUB nebo v přepínaných sítích na sledovaný zrcadlený (SPAN) port.

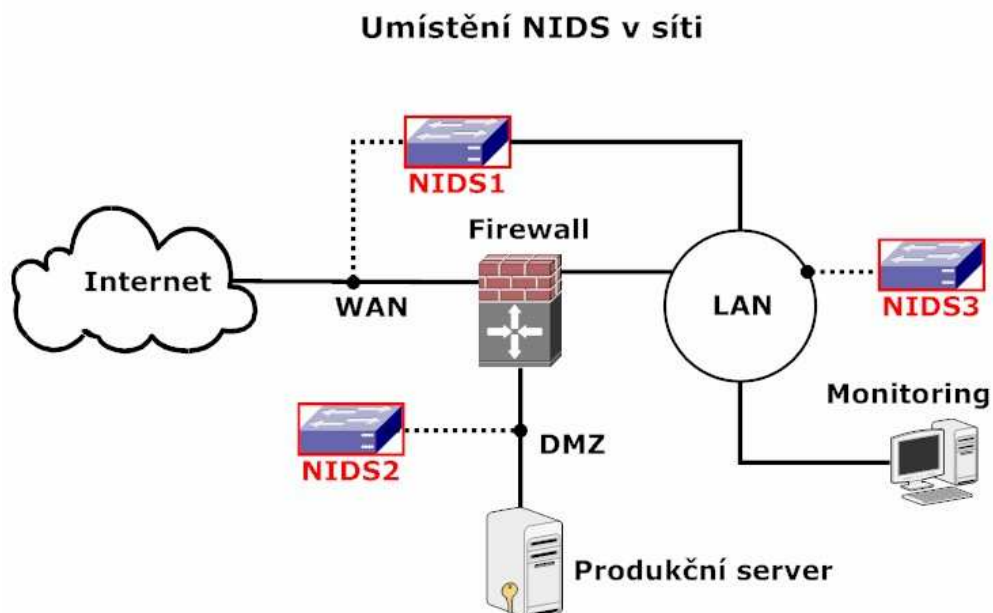
Systémy IDS rozlišujeme na **HIDS** a **NIDS** podle způsobu a místa sledování síťového provozu.

3.1.1 HIDS (Host Based Intrusion Detection System)

HIDS jsou nasazovány na koncových stanicích a serverech. Kontrolují lokální aktivity pouze na sledovaném hostiteli, což jim umožňuje kontrolovat šifrované kanály. Nevýhodou je jejich napadnutelnost při DoS útoku.

3.1.2 NIDS (Network Based Intrusion Detection System)

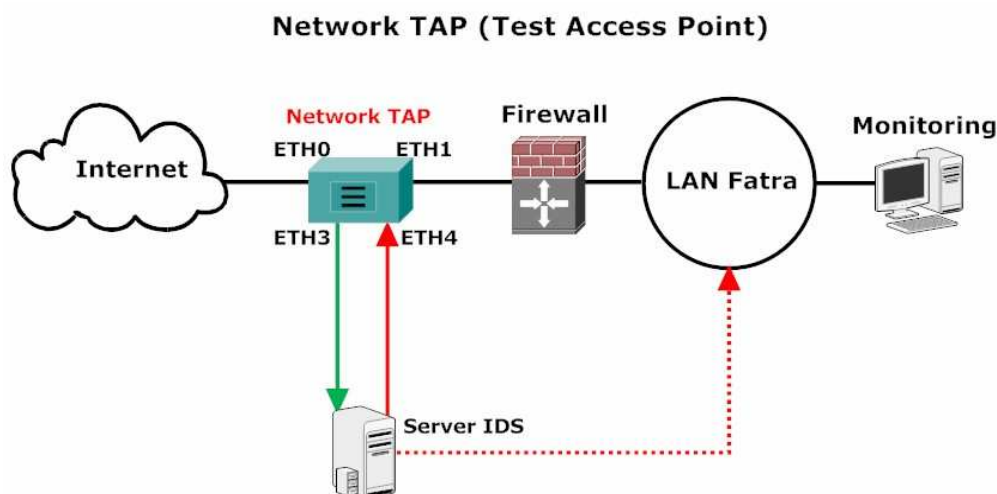
NIDS je SW systém nebo zařízení na principu sledování provozu sítě v jednotlivých segmentech. Při správném nastavení dokáže odhalit průnik podle aktivity na síti. Do spojení fyzicky nezasahuje, pouze informuje poplachem na určená místa. Obvykle jej instalujeme za firewall do vnitřní sítě. Pro lepší vyhodnocování útoků může další detektor umístit před firewallem a v DMZ. Problémem těchto systémů je vysoká míra falešných poplachů.



Obr. 1: Příklad umístění více zařízení NIDS v lokální síti ^[20]

3.2 IPS (Intrusion Prevention System)

Dnešní IPS^[1] je technologie, která překračuje možnosti firewallu a posouvá hranice bezpečnosti o kousek dále. V podstatě je to nadstavba IDS obohacená o schopnosti firewallu spolu s antivirem. IPS je schopen aktivně reagovat na příchozí útoky, bývá nasazen „in-line“ v síťovém provozu a přímo blokuje narušení nebo spolupracuje s místním firewallem. Dokáže zasáhnout do pravidel nebo odepřít přístup zdrojové adrese útočníka. Systémy IPS mají podobné rozlišení jako IDS, můžou být uzlově orientované **HIPS** nebo síťové **NIPS**. Existují samostatné systémy zapojené z bezpečnostních důvodů v clusteru (dvě stejné jednotky paralelně) nebo pomocí síťového „TAP⁷“ (při poruše IPS data proudí přes „TAP“). Mezi nejznámější výrobce patří Cisco Systems a CheckPoint.



Obr. 2: Příklad zapojení Network TAP před firewall^[20]

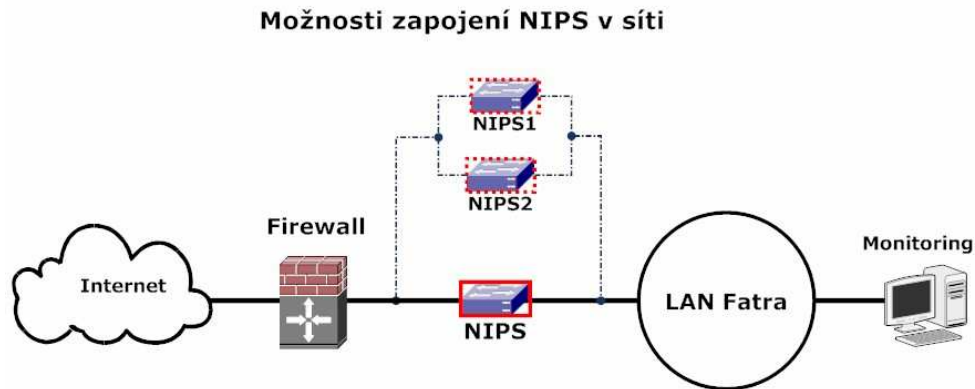
3.2.1 HIPS (Host Based Intrusion Prevention System)

HIPS detekuje útoky na aplikace lokálního systému (viry, červy, škodlivé kódy), dokáže kontrolovat šifrované spojení na aplikační úrovni.

⁷ TAP (Test Access Point) - pasivní síťový prvek určený pro sledování provozu mezi sítěmi. Pro své okolí však není viditelný.

3.2.2 NIPS (Network Based Intrusion Prevention System)

NIPS je řazen nejčastěji za firewall, průchozí data jsou v NIPS vyhodnocena. Odráží útoky ze sítě (Portscan, DoS, DDoS, atd.).



Obr. 3: Volba zapojení NIPS v lokální síti ^[20]

3.3 Rozdíl mezi IDS a IPS

IDS provádí především monitoring s následným vyvoláním poplachu a sám do komunikace nikterak nezasahuje. Výjimkou jsou Firewally a IDS schopné spolupracovat a dynamicky měnit pravidla na základě standardu OPSEC⁸. Toto řešení však nemá všeobecnou podporu a vyžaduje si kompatibilní kombinaci firewallu a IDS.

Systémy IPS jsou složitější zařízení, které jsou vyvíjeny komerčně. Svou funkcionalitou přímo ovlivňují komunikaci a stejně tak jako firewally či antiviry je nelze provozovat jako Open Source. IPS je nezávislý, pracuje sám a je schopen aktivně zabránit útoku ještě dříve než dojde k navázání komunikace. Může rozhodnout o přístupu nejen podle adresy a portu,

⁸ „OPSEC (Open Platform for Security) - otevřené sdružení dodavatelů bezpečnostních řešení. Se svými více než 270 partnery OPSEC zákazníkům nabízí nejširší nabídku integrovaných aplikací a platform, které podporují Check Point architekturu Secure Virtual Network. Certifikace OPSEC zaručuje, že daný produkt je plně interoperabilní a je možné jej bezproblémově integrovat s produkty Check Point. Kompletní informace o sdružení OPSEC, seznam partnerů a certifikovaných bezpečnostních produktů a zkušební verze těchto produktů jsou k dispozici na [http://www.op-sec.com]. Na stejné adrese je bezplatně dostupné také OPSEC SDK (software development kit)“. Zdroj: [http://www.lupa.cz].

ale i podle obsahu dat. Oproti IDS zde však hrozí riziko odepření komunikace kvůli špatně vyhodnocenému incidentu.

Systemy IDS/IPS využívají podobnou filozofii rozpoznání malware a tím se podílí na zvyšování úrovně bezpečnosti v síťovém provozu. V textu jsou popsány jejich základní principy a rozdíly. Výhody či nevýhody jednotlivých systémů závisí na konkrétní situaci v dané firmě. V oblasti IPS existují velmi kvalitní produkty renomovaných výrobců používající složité mechanismy pro odhalení útočníka. Náročnost jejich konfigurace a správy si však žádá hodně pozornosti a důkladné vyškolení obsluhy. ^[8]

3.4 Detekční sondy

Nejčastější způsob sběru dat je zrcadlení (mirroring, spanning) zdrojového portu v přepínači na jiný port, kde je umístěna sonda. Funkce zrcadlení portu se nachází v novějších aktivních prvcích s managementem. V rámci jednoho zařízení lze takto definovat porty a pohodlně filtrovat pakety příchozí, odcházející nebo obojí současně. ^[8]

Dalším způsobem sledování komunikace je větvení sítě (network „TAP“). Pakety prochází skrz, zkopírují se a pak pokračují dále.

Nasazením síťových a uzlových sond získáme rozhodující informace pro analýzy použitelné v IDS/IPS.

3.4.1 Síťová sonda

Jedná se o HW zařízení nebo účelový SW, který zachytává přes síťové rozhraní data procházející všemi uzly sítě. Obsahuje obvykle dvě síťové rozhraní. Jedno je monitorovací (Smíšený režim, bez IP adresy a TCP/IP zásobníku) a druhé pro management. V sítích s ethernetovými přepínači je nutno využít funkce zrcadlení (přemostění) portů. Správně navržený systém je pro své okolí neviditelný a nikterak nezatěžuje provoz. ^[8]

3.4.2 Uzlová sonda

V unixovém prostředí pracují tyto sondy jako démoni monitorující jen daný uzel. Posílají sesbírané data do centrálních aplikací, což jsou většinou analytické nástroje pro sledování událostí. Uzlová sonda zaznamenává pouze události vznikající v daném uzlu. Dává tedy přesnější informace o pokusech nebo napadení tohoto místa. ^[8]

4 CO JSOU HONEYPOTY ?

Myšlenka honeypotů se zrodila počátkem 90. let s nárůstem aktivit škodlivého kódu neboli malware. „Honeypot“ (hrnec medu) je výstižné pojmenování pasti na vetřelce (jail-technology⁹), která je založena na principu polapení útočníka do nástražného systému. Představuje mimikry virtuální síťové infrastruktury včetně předstíraných zdrojů, kde dochází ke zdržení útočníka a záznamu jeho činností. ^[9]

4.1 Účel honeypotů

Úkolem honeypotů je tedy poskytnutí dostatečného množství informací administrátorovi, aby získal čas, pochopil strategie útoků a aby mohl provést preventivní opatření. Honeypoty nemají žádnou jinou produkční funkci, proto lze považovat každý pokus o jejich kontaktování za útok. Této skutečnosti pak odpovídá i jednoduchá správa logů, ve kterých nejsou obsaženy falešné popluchy, jako tomu bývá u systémů IDS/IPS, ale jen platné pokusy o průnik. Pasti rozdělujeme do skupin podle několika kritérií. Podle intenzity spolupráce s protivníkem jsou to honeypoty s nízkou nebo vysokou mírou interakce, podle směru působení na serverové a klientské, podle platformy virtuální či fyzické, podle místa využití na produkční a výzkumné. Virtuální honeypoty patří mezi nejvíce rozšířené. Mají malou režii a dokáží imitovat na jednom serveru stovky služeb nebo produkčních sítí. Fyzické servery jsou většinou výkonné profesionálně a vědecky zaměřené stroje pro výzkum a vývoj bezpečnostního software.

4.2 Co je honeynet ?

Honeynet je vlastně síť honeypotů, umístěná za firewallem s implicitně instalovanými počítači bez jakékoliv bezpečnostní ochrany. Je to vysoce interaktivní soustava, ve které je klíčovým prvkem „gateway Honeywall¹⁰“. Tyto systémy slouží k zachycení velkého množství informací o potencionálních hrozbách. Jsou provozovány především jako servery pro výzkumné účely nebo v laboratořích vědeckých uskupení. Mezi nejznámější neziskové

⁹ Jail-technology - pojmenování technologie přitahující potenciální útočníky. ^[15]

¹⁰ Gateway Honeywall (obdoba firewallu) - brána, která odděluje síť honeynet od ostatních sítí. ^[15]

organizace, které se touto tematikou zabývá je „The honeynet Project“¹¹, ve které od roku 1999 odborná skupina programátorů shromažďuje odkazy na softwarové zdroje a informace o taktikách útočnicků. Význam honeynetů nabývá na rozměrech zejména pro softwarové společnosti, které vyvíjí komplexní bezpečnostní řešení. Evidují malware aktivity a sledují nové trendy. Informace získané pomocí honeynetů mají pro ně hlubší význam a jsou komplexnější nežli data běžně odesílaná klientskými aplikacemi.^[9]

4.3 Rozdělení honeypotů

Honeypoty můžeme rozdělit podle několika základních kritérií. Podle místa svého působení se dělí na serverové nebo klientské, podle úrovně spolupráce na honeypoty s vysokou nebo nízkou mírou interakce. Volba typu honeypotu a metodiky imitace služeb záleží na tom, co si od monitorování útočnicka slibujeme. Rozsah nabízené spolupráce s útočníkem je logicky přímo úměrný s počtem získaných informací. Plnohodnotný honeypot s vysokou mírou interakce je pro útočnicka důvěryhodnější a nabízí mu větší volnost v jeho aktivitách. Z tohoto hlediska zde ale existuje jisté nebezpečí ovládnutí honeypotu a vystavení se riziku zneužití.^[7]

4.3.1 Serverové honeypoty

Patří mezi historicky nejvíce rozšířené honeypoty, které jsou charakteristické pasivním chováním, čekají na útočnickovu aktivitu a poté navazují interakci. Monitorují zejména nové útočné aktivity na zranitelnosti systémů nebo neznámé internetové červy. Serverové honeypoty jsou umístěny v klíčových uzlech sítě a patří mezi výkonné nástroje pro sběr informací o útocích. Mezi nejpoužívanější patří Honeyd a Nephentes viz. kapitola 4.3.3.

4.3.2 Klientské honeypoty

Od serverových honeypotů se liší především aktivitou. Jsou to virtuální stanice, které simulují chování uživatele na síti. Příkladem je HPC Capture, vytvářející iluzi skutečného

¹¹ The honeynet Project – mezinárodní výzkumná organizace pro zlepšení internetové bezpečnosti a vývoje bezpečnostních Open Source nástrojů. Zdroj: [<http://www.honeynet.org/>].

uživatelé. Aplikace má instrukce k automatickému procházení definovaných internetových odkazů. Tím aktivně vyhledává údaje o škodlivých kódech, které jinak nelze získat. ^[7]

4.3.3 Nenativní honeypoty

Jsou honeypoty s nízkou mírou interakce. Patří mezi nejvíce používané typy honeypotů. Jejich popularitu můžeme přisuzovat snadné instalaci, jednoduché konfiguraci a nízké míře rizika zneužití. Tyto výhody jsou samozřejmě vykoupeny limitovaným množstvím získaných dat. Způsob prezentace je vlastně emulace částí operačních systémů s malým počtem příkazů. Útočníkovi je podstrčena velmi reálná imitace známých zranitelností systémů a soustav sítí. Nejsou to skutečné systémy, pouze skripty napodobující služby nebo celé sítě, přepínače a směrovače. Z průběhu záznamu v logu se dozvíme intenzitu, čas, službu a internetovou adresu vetřelce. Pro splnění našich požadavků a provedení včasného zásahu jsou získané údaje plně dostačující.

- **Honeyd** – asi nejpopulárnější v kategorii nástrah s nízkou mírou interakce. Jedná se o démon běžící v Linuxovém prostředí, který emuluje operační systémy, aktivní prvky a služby sítě. Zdrojové kódy jsou volně ke stažení na domovské stránce [<http://www.honeyd.org/>]. Podporované platformy: BSD, Linux, Solaris, Windows.
- **LaBrea** – tato nástraha vznikla jako reakce na řádění červa „Code Red“ v r.2001. Účelem nástrahy je postupné zpomalení odpovědí až do úplného zastavení aktivity útoku. Dostupný na [<http://labrea.sourceforge.net/>]. Podporované platformy: FreeBSD, Linux, Solaris, Windows 98/2000.
- **HoneyWeb** – aplikace v Perlu, která vytváří emulace web serverů.
- **Nepenthes** – láká útočníky na známé zranitelnosti emulované na virtuálních klientech. Je určen k detekci nových škodlivých kódů, zdroje jsou dostupné na domovské stránce [<http://www.nepenthespharm.com/>]. Podporované platformy: Windows, Linux.

4.3.4 Nativní honeypoty

Jsou náročnější honeypoty s vysokou mírou interakce. Jsou tvořeny kompletním operačním systémem a nabízí útočníkovi daleko více možností. Dosahují lepších výsledků v získání

podrobnějších údajů o metodách útoků, použitých algoritmech nebo způsobech šifrování. Jejich provozování je ale určitým rizikem pro hostitele, neboť může dojít k jejich zneužití.

- **The Honeynet Project** – v současnosti nejvíce aktivní projekt viz. kapitola 4.2.
- **Argos** – detekce a zachycení útoků typu „zero-day-exploit“, založený na QUEMU¹². Domovská stránka se zdrojovými kódy Argos: [<http://www.few.vu.nl/argos/>].
- **Potemkin** - použití pro velký počet virtuálních honeypotů. Technologie využívá virtualizační platformu XEN¹³ [<http://cseweb.ucsd.edu/~savage/papers/Sosp05.pdf>].
- **HPC Capture** – patří do skupiny klientských honeypotů. Jeho architektura je složena ze serveru na platformě linux a klientů nejčastěji na operačním systému Windows [<https://projects.honeynet.org/capture-hpc>].

¹² QUEMU - open source emulátor procesů a operačních systémů, který pracuje ve dvou režimech. V plné emulaci, kdy umožňuje přímo nabootovat virtuální počítač a provozovat na něm libovolný operační systém nebo v uživatelské emulaci, kdy umožňuje spouštět aplikace určené pro jiný procesor. Zdroj: [<http://www.abclinuxu.cz/software/system/emulatory/qemu>].

¹³ XEN - open-source virtualizační systém jehož hypervizor umožňuje současný běh více virtuálních strojů na jednom hardware. Xen jde přitom cestou paravirtualizace, kdy se virtuální stroje netváří jako samostatné fyzické, neboť neemulují úplný hardware. Výhodou tohoto přístupu je vyšší výkon paravirtuálních strojů, než jaký mohou poskytnout stroje plně virtuální. Zdroj: [<http://www.abclinuxu.cz/software/system/virtualizace/xen>].

5 ANALÝZA ZABEZPEČENÍ SÍTĚ VE SPOLEČNOSTI FATRA, A.S.

Podmínkou fungování dobře prosperující organizace je důsledné zabezpečení ICT. K vytvoření takového prostředí je potřeba stanovit pravidla provozování informačních systémů a zajistit materiální i personální zdroje.

Společnost Fatra, a.s. jasně vymezuje základní pravomoci a zodpovědnosti při pořizování, používání, správě a údržbě prostředků informačních a komunikačních technologií ve vypracovaných podnikových směrnících. Mezi nejdůležitější oblasti patří postupy při udělování přístupových práv, dodržování licenčních politik, zálohování, archivace, obnova dat a fyzické zabezpečení hardware. Za tímto výčtem pojmů stojí dokumentace se závaznými postupy nebo činnostmi správců. Postupy zobecňují první fázi firemní strategie informační bezpečnosti a jsou základem pro vytvoření komplexních obranných struktur.

Koncept bezpečnostních politik je řetěz návazností, měl by zohledňovat obecné a sociotechnické zákonitosti:

- Každý systém je nějakým způsobem napadnutelný.
- Vlastníme-li sebelepší technologii, bez know-how zůstane vždy jen technologií.
- Za úroveň bezpečnosti nesou zodpovědnost pouze lidé.
- Obranný systém bude jen tak silný, jako jeho nejslabší článek.
- Bezpečnost je proces, který si žádá neustálou pozornost.
- Opatření spojená s ochranou dat budou uživatele omezovat vždy.

5.1 Bezpečnostní složky v síti Fatra, a.s.

- **Firewally**

Ve správě ICT Fatra, a.s. se nachází 5 firewallů na více úrovních. Nejdůležitější je centrální CheckPoint NGX R71 oddělující lokální síť VLAN od internetu. Jeho rozlišovací schopnosti však končí na transportní vrstvě standardu ISO/OSI. Dále poskytuje služby VPN IPSec a QoS. Mezi VLANy jsou vloženy firewally Cisco ASA 5505 a CheckPoint Safe@office 500.

- **VPN přístupy**

Vzdálené připojení k firemním zdrojům je již dnes samozřejmostí. Pokud to chceme učinit, musíme také samozřejmě zaručit, že spojení nebude možno zneužít a zachytit přenášená data. VPN v síti Fatra, a.s. je součástí firewallu CheckPoint a k vytvoření tunelu používá

metod certifikovaných klíčů a šifrování. Připojení k cíli je bezpečné a současnými metodami nenapadnutelné.

- **Doménové a skupinové politiky**

Nejčastější příčinou ztráty nebo zneužití dat jsou samotní uživatelé lokální počítačové sítě. Pravidlem číslo jedna je zakázat vše a poté povolit jen nutné, schválené minimum. Vysoká hodnota firemních zdrojů vede k ostražitosti, proto musí být data chráněna doménovými politikami. Přidělování oprávnění je podrobena schvalovacímu procesu s odpovědností hierarchií. Organizační jednotky obsahují skupiny uživatelů podle stupně rizika zneužití výpočetní techniky. Snahou je minimalizace veškerých rizik spojených s jejich neoprávněnou manipulací. Většina uživatelů má zamezeny přístupy k internetu, výměnným médiím, ke změnám nastavení systému a instalacím.

- **Lokální firewally**

Lokálním firewally jsou centrálně řízeny doménovými politikami z Active Directory¹⁴, kde jsou definovány bezpečné služby a aplikace, ostatní je zakázáno. Běžný uživatel nemá oprávnění nastavení měnit.

- **Antispam**

Hrubý odhad současných statistik uvádí, že reálný podíl skutečné, vyžádané pošty tvoří pouze 20% z celkové komunikace. Přitom značná část zákeřného software se šíří pomocí spamu. Filtrování e-mailů je proto nevyhnutelné. Antispam ve firmě Fatra, a.s. probíhá na serveru Symantec Messaging Gateway.

- **Antivirus**

Každá koncová stanice, připojená k síti je vybaven centrálně spravovaným klientem Symantec Endpoint Protection. Mimo virovou detekci je klient schopen zachytit škodlivý kód na základě signatur a inteligentního rozpoznávání přijatelného chování. Úroveň ochrany stanic je po této stránce na přijatelné úrovni.

¹⁴ Active Directory – adresářové služby v doméně Microsoft Windows

- **Aktualizace software Microsoft**

Většina PC stanic v doméně Fatra, a.s. používá operační systém Microsoft Windows , který vyžaduje pravidelnou aktualizaci bezpečnostních záplat. K tomuto účelu je v síti spuštěna automatická služba WSUS. Na serveru probíhá denní kontrola a stahování servisních balíčků. Ty jsou serverem WSUS distribuovány po síti pomocí skupinových politik Active Directory.

- **Detekce a prevence narušení**

Licenční politika CheckPoint umožňuje rozšiřovat balík centrálního firewallu NGX R71 o další „blade“ software. Modul Check Point IDS/IPS Software Blade patří mezi vysoce sofistikované nástavby. Jedná se o profesionální nástroj k detekci a prevenci průniku. Velkým problémem je ovšem jeho vysoká pořizovací cena a udržovací poplatek, což obvykle způsobuje managerům vrásky při obhajobě ročního ICT rozpočtu.

- **Zálohování a archivace**

K pravidelnému zálohování většího množství dat z rozličných zdrojů jsou nejčastěji využívány pásková zařízení s knihovnou záznamových médií. Tuto ověřenou a cenově dostupnou technologii využívá i společnost Fatra, a.s. Nastupujícím trendem jsou však rychlé sofistikované diskové zálohovací centra s deduplikací, redundancí, snapshotováním¹⁵ a rychlou obnovou. Obnova poškozených dat nebo celých serverů je pak otázkou několika minut.

- **Vysoká dostupnost aktivních prvků**

Jedna ze složek bezpečnostních opatření síťové infrastruktury se týká požadavků na vysokou dostupnost přenosového média. Z pohledu uživatelů má výpadek aktivních prvků stejně nepříjemný dopad jako kritická závada celého informačního systému. Přichází pak o neuložená data nebo mají nedostupné služby. Vytváření bezpečnostních politik by mělo zahrnovat vysokou dostupnost sítě. Strukturovaná síť ve Fatra, a.s. obsahuje zhruba 1800 přípojných míst ve dvou vzdálených lokalitách. V obou lokalitách se jedná o přepínanou síť ethernet v topologii „Dual Homing“. Obsahuje redundanci centrálních přepínaných

¹⁵ Snapshot – kopie, datový snímek běžícího souborového systému.

prvků i optických spojů a zajišťuje trvalý provoz při výpadku jednoho z center. Dále je síť rozčleněna na více VLAN segmentů z důvodu eliminace kolizních domén. Celkově se v síti nachází 600 až 700 zařízení s aktivním připojením do 10 virtuálních sítí.

- **Virtualizace**

Virtualizace serverů i desktopů patří mezi přelomové technologie tohoto století, která staví oblast ICT do úplně jiných dimenzí. Po stránce bezpečnosti nabízí uživateli vytváření serverových svazků, snímků souborového systému, replikace a migrace operačních systémů za chodu. Vysokou variabilitou zálohování přispívá ke zlepšení stavu softwarové bezpečnosti. V síti Fatra, a.s jsou již částečně servery virtualizovány na platformě VMware ESXi. Tento stav není konečný a plnohodnotného řešení je stěžejním úkolem nejbližších dnů.

5.2 Shrnutí a nedostatky

Ochrana koncových stanic podléhá centrální správě doménových politik, aktualizací služeb a politik antivirového systému s trvalým přísunem aktuálních definic. Použitá verze antiviru patří mezi komplexní řešení s dobrou referencí. Lze tedy konstatovat relativně dobrou úroveň zabezpečení koncových stanic.

Datová síť Fatra, a.s. vykazuje bezpečnostní nedostatky v oblasti odhalování průniku do informačního systému. Schází integrace složek, které by prováděly korelaci datového toku z obsahového hlediska. Tyto hrozby nelze podceňovat neboť množství a záludnost útoků malware ze světových hackerských velmocí neustále roste. Správci nejsou informováni o možných rizicích, incidenty jsou maskovány za autorizovaná pravidla a v rámci logovacích záznamů na firewallu jsou naprosto legitimní. Administrátor se o této skutečnosti vůbec nemusí dozvědět, pokud na sebe útočník sám neupozorní. Za úspěšný útok je mezi zkušenými hackery považován právě ten, který zůstane pro oběť utajen a on může dlouhodobě čerpat cenná data na „dobitém území“.

Ke zlepšení současného stavu zabezpečení zdrojů společnosti by významně přispěla instalace systému detekce průniku a honeypotů. Obsah síťového provozu pak bude zviditelněn a administrátor bude o případném útoku včas informován.

6 NÁVRH ŘEŠENÍ

V předchozí kapitole byly odhaleny nedostatky v oblasti detekce a prevence před neoprávněným vniknutím do infrastruktury ICT. Z hlediska vysoké pořizovací ceny přídatného modulu „Blade IPS“, což je doplněk ke stávajícímu systému CheckPoint firewall“, se poohlédneme po dostupnějším řešení.

Nabízí se možnost využití doporučených Open Source zdrojů a vytvoření vlastního systému na nekomerční bázi. Vhodným řešením je kombinace operačního systému „Linux Ubuntu server 10.10“, aplikace IDS „Snort 2.9“ spolu s honeypotem s nízkou mírou interakce „Honeyd 1.5c“. Tyto aplikace spadají pod skupinu software GNU GPL, mají širokou podporu a dosahují velmi dobrých praktických výsledků.

Instalací IDS Snort a nástrah Honeyd spolu s vhodnou konfigurací analytických nástrojů lze dosáhnout vysoké úrovně aktuálních přehledů o podezřelých aktivitách a dění na kritických místech sítě.

6.1 Virtualizace

Jedním z důležitých aspektů úspěšné aplikace softwarového projektu je jeho stabilita a udržitelnost. V tomto projektu bude zapotřebí zprovoznit desítku skenovacích sond IDS a několik serverů pro emulaci pastí Honeyd. Snahou je sestavit softwarový systém na dostupném, spolehlivém a přitom snadno spravovatelném hardware. Vzhledem k počtu sond je provoz takového systému na běžném hardware z časových a technických důvodů náročný.

Snadnější cestou je využití dostupných virtualizačních platform (např. VMware vSphere Hypervisor ESXi 4.1, VMware Server 2, VMware Player 3.0, Microsoft Hyper-V Server 2008 R2, Microsoft Virtual PC 2007, Oracle VirtualBox 4.0, Xen Hypervisor 4.1, Citrix XenServer 5.5.), které usnadní provoz, zálohy, údržbu nebo vytváření ladících verzí.

Firma Fatra, a.s. disponuje systémem IBM Blade Centrum s volnou HW kapacitou serveru IBM HS21XM, který ideálně poslouží pro vytvoření monitorovacího systému. Blade centrum obsahuje 4 ethernetové rozhraní, použitelné při implementaci sond do různých částí sítě. Operační systém Ubuntu bude instalován na virtualizační platformě VMware vSphere ESXi, která nabízí bezplatný provoz i pro komerční účely. Oproti plnohodnotné

placené verzi VMware vSphere™ má určité omezení, které však pro stávající projekt nejsou klíčové.

6.2 Operační systém Linux Ubuntu

Historie Ubuntu sahá do roku 2004, kdy Mark Shuttleworth¹⁶ založil společnost Canonical Ltd. a skupinu softwarových vývojářů Ubuntu. Zanedlouho vydal první distribuci operačního systému Ubuntu 4.10 "Warty Warthog". Filozofie systému spočívá ve vývoji „svobodného software“ pro libovolné použití. Systém je z větší části založen na původní otevřené distribuci Linux Debian a prostředí Gnome. V současné době je dostupná poslední verze Ubuntu 10.10 "Maverick Meerkat" pro servery a 10.04 „Lucid Lynx“ pro osobní počítače a notebooky.

Linux Ubuntu můžeme instalovat na téměř jakoukoliv HW architekturu (x86, AMD64, UltraSPARC T1 a OpenPower (Power5)). Pokud si však nejsme jisti, existuje seznam kompatibilního HW (<https://wiki.ubuntu.com/HardwareSupport>). Instalační balíky Ubuntu server (obrazy instalačních disků s příponou „iso“) jsou k dispozici na <http://www.ubuntu.cz/ziskejte/stahnout>. Instalace je rychlá a přehledná. Před spuštěním je vhodné zajistit přístup do internetu. Během instalace systém provádí konfiguraci balíčkovacího nástroje a stahuje si aktualizace (např. LAMP - Linux, Apache, MySQL, PHP). Pokud nebudeme mít přístup k internetu, čas instalace se může protáhnout.

Operační systém Ubuntu má širokou celosvětově udržovanou členskou základnu programátorů a nadšenců podporujících vývoj systému i uživatele. V naší republice je aktivní nezisková organizace „Občanské sdružení Ubuntu pro Českou republiku“, která zajišťuje rozšíření systému, lepší dostupnost, distribuci a lokalizaci do českého jazyka.

¹⁶ Mark Shuttleworth – úspěšný jihoafrický podnikatel, zakladatel společnosti Canonical Ltd. a projektu Linux Ubuntu, majitel společnost Thawte, sponzor mnoha open source projektů. Zdroj: [<http://www.abclinuxu.cz/kdo-je/mark-shuttleworth>].

6.3 IDS Snort

Snort patří mezi nejpopulárnější nekomerční produkty IDS. Tento produkt vyvinul v 90. letech Marty Roesch¹⁷. Snort je stále zdokonalován (aktuální verze 3) a doposud se těší velké oblibě. Můžeme jej naléznout v mnoha odnožích (např. IDS - Snorby), které čerpají z jeho jádra. Jinde je v softwarových distribucích coby doplňkový bezpečnostní prvek.

Snort je automatický nástroj založený na pravidlech a signatury používá až při identifikaci útoku. Lze jej použít jako NIDS i jako HIDS. Při rozboru signalizovaného poplachu probíhající spojení nepřerušuje a poskytuje mnoho přídavných funkcionalit pro identifikaci odkud útok přichází. Velkou výhodou Snortu je možnost vytváření vlastních pravidel a signatur. Tento nástroj je schopen fungovat na mnoha systémech např. Linux, Windows, Solaris, FreeBSD, OpenBSD, NetBSD, MacOS.^[14]

Sondy se nasazují v hlavních uzlech provozu, na každém rozhraní firewallu a na jednotlivých segmentech sítě. Takové zařízení může být například virtuální linuxová stanice se dvěma sítovkami. První monitorovací rozhraní je napojeno na uzel (SPAN port) a nemá IP adresu, je tedy pro útočníka neviditelné. Druhé funguje jako spojení s managementem, kde se vytváří statistiky a grafy incidentů.

Zpracování datového toku si vyžaduje značné hardwarové nároky. Zvláště na páteřních linkách, kde data dosahují vysokých přenosů jsou záznamové a detekční služby dost přetíženy. Na to bychom měli pamatovat při výběru HW. Větší efektivity v prevenci dosáhneme použitím samostatných senzorů se společným vyhodnocováním výsledků v centrální databázi.

6.3.1 Rozbor aplikace Snort

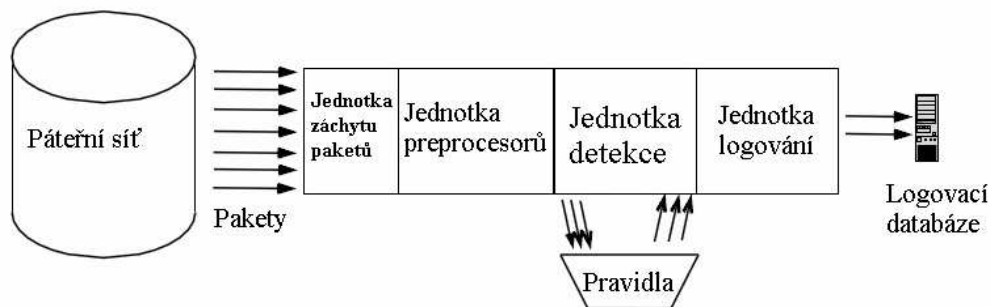
V systému spolupracuje několik komponent současně. Jejich úkolem je data zachytit, upravit, analyzovat a vyhodnotit. Výsledkem jsou grafické reporty a získané informace o útočnickovi.

¹⁷ Marty Roesch - zakladatel a v současnosti technický ředitel Sourcefire, autor a hlavní vývojář software Snort, je uznávanou autoritou v oblasti inteligentních technologií detekce a prevence narušení. Zdroj: [http://en.wikipedia.org/wiki/Martin_Roesch].

Snort běží ve třech základních režimech:

- Sniffer - zachytává a zobrazuje datový tok.
- Packet logger - provádí záznam do logovacího souboru.
- Network Intrusion Detection System (IDS) - analýza odchycených dat.

6.3.2 Definice jednotek Snort



Obr. 4: Architektura IDS Snort ^[2]

- **Jednotka záchytu paketů** - sbírá a formuje pakety z různých síťových rozhraní pro detekční jednotku.
- **Jednotka preprocesorů** - jsou to moduly, které upravují pakety a připravují je k analýze v IDS. Preprocesory zkoumají hlavičky paketů, provádí jejich defragmentaci, přeskupuje a převádí řetězce, kterým IDS nerozumí (např. hexadecimální a unicode znaky), převádí je do srozumitelného tvaru.
- **Jednotka detekce** - hlavní komponenta Snortu, která systematicky porovnává zachycené data, zda odpovídají některému z pravidel. Pravidla jsou načtena ve vnitřních datových strukturách, oproti kterým se provádí rozbor každého paketu. Operace detekční jednotky jsou časově náročné a žádají si velký výpočetní výkon. Při nedostatečném výkonu počítače jsou některé pakety zahazovány a to znamená zkreslení výsledků.
- **Jednotka logování** - zapisuje do souboru události a výstrahy vyhodnocené detekční jednotkou jako podezřelé. Výsledkem je textový soubor ve tvaru tcp-dump ze kterého čerpá následující jednotka výstupního modulu.
- **Jednotka výstupního modulu** – definujeme, jak a kam budeme hlášení ukládat nebo posílat. Můžeme je vkládat do databáze, posílat SNMP trapy do jiných

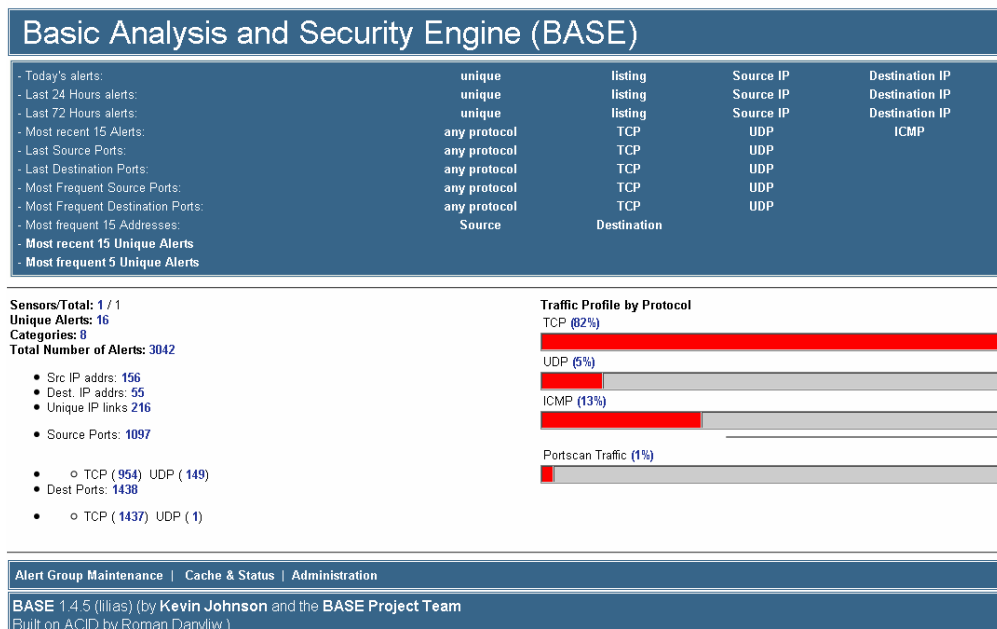
monitorovacích systémů, exportovat do XML formátu nebo upravovat pravidla na podporovaném zařízení. ^[14]

6.3.3 Analytické nástroje logovacích souborů Snort

Využitím vhodného analytického nástroje získáme souhrnné statistické informace o incidentech. Data z jednotky výstupního modulu jsou uložena v databázi a obsahují základní informace (IP adresy, porty a typy útoků). Doplnění původu útoku provedeme prostřednictvím vestavěných utilit (Whois, DNS, GeoIP).

- **BASE**

Mezi nejznámější analytické nástroje Snort patří BASE (Basic Analysis and Security Engine). Je to aplikační grafické rozhraní pro analýzy incidentů. Je renovací původní verze ACID (Analysis Console for Intrusion Databases) a slouží jako dotazovací nástroj nad databází Snortu. Vytváří přehledy v datech, upravuje je do svých tabulek a v menu nabízí nástroje pro lepší identifikaci zdrojových adres. Přehledy můžeme poskytnout i ostatním uživatelům na základě autentizační funkce - rule - base, kde vytvoříme účty s přístupy. ^[2]



Obr. 5: Base - nástroj pro analýzu databáze Snort verze 2.8 ^[21]

- **Snort report**

Snort Report je přídatný modul IDS, který provádí rozbor nad Snort - MySQL databází v realime režimu. Události jsou rozděleny podle příznaků signatur v časovém sledu. V detailu jsou reference k signaturám, odkazy na pomocné nástroje a rozbor paketů.

Signature: SHELLCODE x86 inc ecx NOOP					
Earliest Such Alert: 2011-04-19 12:44:33					
Latest Such Alert: 2011-04-19 20:52:25					
Sources Triggering This Attack Signature					
Source IP	FQDN	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
77.75.73.38	download.seznam.cz	32	32	1	1
217.163.21.37	mpr4.ngd.vip.ch1.yahoo.com	20	20	1	1
64.236.124.228	64.236.124.228	4	4	1	1
194.228.62.32	mail10.agrofert.cz	4384	5054	1	1
98.136.75.194	mpr1.ngd.vip.ac4.yahoo.com	2	2	1	1
76.13.220.49	mpr8.ngd.vip.ac4.yahoo.com	1	1	1	1
217.163.21.41	mpr8.ngd.vip.ch1.yahoo.com	13	13	1	1
77.75.77.38	download.seznam.cz	30	30	1	1
98.136.76.244	mpr4.ngd.vip.ac4.yahoo.com	4	4	1	1
209.85.149.102	ber01s02-in-f102.1e100.net	1	1	1	1
74.125.39.139	fx-in-f139.1e100.net	6	6	1	1
194.79.52.199	c8.idnes.cz	2	2	1	1
Destinations Receiving This Attack Signature					
Dest IP	FQDN	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
88.81.91.130	rt.fatra.cz	115	899	11	127
88.81.91.139	88.81.91.139	4384	5072	1	9

Obr. 6: SnortReport - nástroj pro analýzu databáze Snort verze 2.9 ^[21]

6.4 Honeygot Honeyd

Projekt Honeyd vzniknul v roce 2002 pro podporu výzkumu útočných strategií hackerů a boje proti internetové kriminalitě. Honeyd je zkráceným názvem dvou slov honeypot, daemon a jeho tvůrcem je PhD. Niels Provos¹⁸.

6.4.1 Proč použít Honeyd ?

Důvodem nasazení Honeyd je především jeho flexibilita, jednoduchost konfigurace a implementace. Patří mezi nejvíce podporované systémy nástrah a zainteresované

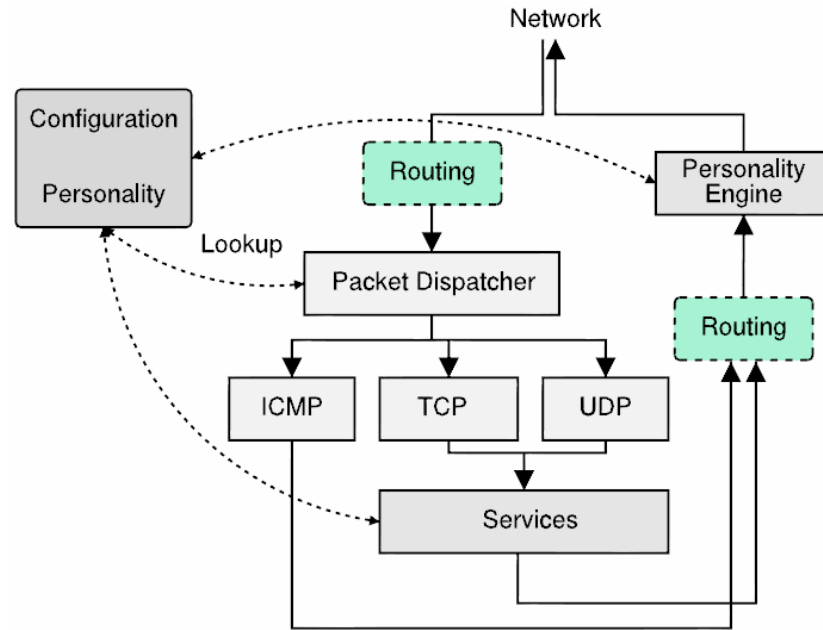
¹⁸PhD. Niels Provos - německý softwarový inženýr, tvůrce aplikace Honeyd a publikací v oblasti internetové bezpečnosti. Zdroj: [http://www.honeyd.org/].

internetové zdroje jej velmi dobře hodnotí. Lze jej doplnit o další podpůrné utility pro zlepšení přehledů logovacích záznamů. Honeyd je emulátor operačních systémů na volných IP adresách a podle potřeby na nich spouští simulace služeb, např. HTTP, IIS, POP3, SMTP, FTP, SSH. Napodobuje chování síťových směrovačů, např. ztráty, zpoždění nebo zahození paketů. Upravuje odpovědi tak, aby odpovídaly skutečným systémům. Jeho podstatou je vytvoření maximální iluze reálných systémů.

Přestože autor tohoto produktu ukončil vývoj již v roce 2007, Honeyd se těší stále velké oblibě a je hojně využíván a doplňován. Patří mezi nástrahy s nízkou mírou interakce, jeho démon běží pod operačním systémem a může reprezentovat až 65535 hostů (honeypotů) současně. Existují variace pro Windows [<http://www.netvigilance.com/winhoneyd>], Linux [<http://www.honeyd.org/release.php>], FreeBSD a Solaris [<http://www.citi.umich.edu/u/provos/honeyd/>]. Honeyd je prezentován dostatečnou interakcí s útočníkem a kvalitou zpracování požadovaných výsledků, splňuje tedy podmínky k provedení jednoho z úkolů této práce.

6.4.2 Architektura Honeyd

Pro zvýšení reálného dojmu musí být Honeyd schopen simulovat více síťových topologií najednou. V tomto ohledu je prostředí Honeyd ideálním nástrojem pro vytvoření dojmu rozsáhlých síťových struktur. V konfiguračním souboru můžeme jednoduše emulovat sítě a podsítě s aktivními prvky, které jsou odděleny směrovači s definovanou latencí. Pokud dojde ke kontaktování honeypotu Honeyd, může spustit libovolný operační systém a libovolný počet služeb. Na rozdíl od fyzického stroje je virtuální honeypot simulovaný systém s modelovaným chováním. Virtuální honeypoty jsou lákavější kořistí neboť poskytují více variací systémů a služeb s minimálními požadavky na údržbu a finance. ^[15]



Obr. 7: Architektonický přehled systému Honeyd ^[7]

- **Configuration personality** (konfigurace osobnosti honeypotu)

Základní nastavení honeypotu se nachází v šabloně konfiguračního souboru, ve kterém definujeme jeho správné fungování a přizpůsobení určenému prostředí.

- **Routing** (směrování)

Honeyd virtuálně nabízí libovolný počet směrování do různých sítí. Dokáže napodobovat latenci a ztrátu paketů jako je tomu u skutečných sítí.

- **Packet dispatcher** (plánovač paketů)

Příchozí pakety jsou zpracovávány centrálním paketovým dispečerem, který nejprve prověří jejich délku, kontrolní součet a typ. Dispečer akceptuje pouze TCP, UDP a ICMP pakety, ostatní jsou zahozeny. Pakety jsou pak tříděny podle typu protokolu a odesílány na správnou jednotku.

- **ICMP, TCP, UDP**

V základním nastavení Honeyd jsou povoleny odpovědi na požadavky ICMP. U služeb TCP a UDP protokolů jsou data zasílána virtuálním službám, které požadavek zpracují a popř. odpovídají. Veškeré odpovědi jsou modifikovány fiktivním otiskem (fingerprint) k oklamání nepřítele a věrnému napodobení chování síťového zásobníku operačního systému.

- **Services** (emulované služby)

Emulované služby Honeyd jsou zajištěny externími aplikacemi nad rámci protokolů TCP nebo UDP. O spuštění služby a její chování se stará interní skript napsaný obvykle v jazyce Python, Perl nebo Bash.

- **Personality engine** (nástroj tvorby osobnosti operačního systému)

Protivníci mají snahu odhalit identitu skenovaného systému pomocí nástrojů např. Nmap¹⁹ nebo Xprobe²⁰. Získávají tím informace o povaze systémů pro přesnější zacílení útoků.

Každému honeypotu lze přiřadit určitou osobnost (personality) podle druhu simulovaného systému. Generátor osobností (Personality engine) simuluje síťový zásobník operačních systémů k oklamání protivníků, kteří kontrolují správné otisky operačních systémů. [15]

Odpověď	TCP	UDP	ICMP
Open	Dojde k navázání standardního spojení SYN/ACK	Neodpoví - je zalogováno	Odpoví ICMP správným paketem
Block	Ignoruje paket, nepošle odpověď	Neodpoví -není zalogováno	Ignoruje paket, neodpoví
Reset	Odpoví paketem RST/ACK	Odpoví paketem ICMP Port unreachable (std. chování)	Není definováno
Tarpit	Dochází ke zpoždění až k zastavení spojení	Není definováno	Není definováno

Tab. 1: Tabulka chování jednoduchých portů v konfiguraci Honeyd [7] [22]

¹⁹ Nmap ("Network Mapper") Open Source nástroj pro zkoumání nebo bezpečnostní auditu v síti. Správcům sítě poskytuje užitečné informace o běžících službách, operačních systémech. Je často využíván pro správu přehledů, sledování hostitelů a služeb v síti. Zdroj: [http://cs.wikipedia.org/wiki/Nmap].

²⁰ Xprobe - účelem tohoto nástroje je provádět snímání otisků prstů na základě protokolu ICMP. Umožní zjistit, jaký operační systém na vzdáleném počítači běží, provádí. Zdroj: [http://sourceforge.net/projects/xprobe/].

II. PRAKTICKÁ ČÁST

7 IMPLEMENTACE SNORT

Tato kapitola bude si klade za cíl podrobně zdokumentovat postupy při instalaci a integraci Ubuntu server, IDS Snort a Honeyd do ostrého provozu sítě Fatra, a.s.

Upozornění pro čtenáře:

Některé údaje týkající se citlivých informací budou z bezpečnostních důvodů pozměněny. Týká se to především IP adres, TCP protokolů a přihlašovacích údajů do systémů. Tyto změny však žádným způsobem neovlivní konstrukční podobu skutečného systému. Posloupnost kapitol bude odpovídat reálnému postupu při instalaci a každá kapitola bude ukončena zhodnocením a testem.

7.1 Instalace Linux Ubuntu server

Volba stabilního HW a operačního systému patří mezi nejdůležitější kroky projektu. V kapitole 6.1 jsou uvedeny možnosti a výhody použití virtualizačních platforem v implementační části, které nepochybně přispějí k větší stabilitě a usnadnění realizace projektu. Operační systém Linux Ubuntu je vhodná Open Source platforma pro spolehlivý chod aplikací Snort a Honeyd.

Ve VMware vSphere HypervisorTM (ESXi) vytvoříme vzorovou instalaci Linux Ubuntu Server 10.04 "Lucid Lynx". Hardwarové požadavky uspokojí jeden virtuální CPU, 512MB RAM, 12GB HDD. Po instalaci exportujeme vzorovou instalaci „virtual appliace“ na externí úložiště. Vzor použijeme k urychlení implementace vytvářením klonů.

7.1.1 Postup instalace

- Prvním dotazem instalátoru je volba jazykové mutace, zvolíme stabilní anglickou verzi, Install Ubuntu Server, language English, United States, detect keyboard – No, main menu USA.
- Důležitým momentem je volba adresy, kdy máme možnost odmítnout adresu z DHCP a definujeme vlastní nastavení IP.
- Název serveru, časové pásmo.
- Rozdělení pevného disku.

- Zadáme uživatele a heslo.
- Ve volbě „software selection“ přidáme „LAMP“ a „OpenSSH“ server.
- Přihlásíme jako root („sudo su“).
- Editaci konfiguračních souborů a práci s adresáři nám usnadní (Midnight Commander) „apt-get install mc“.
- Editujeme soubor „/etc/sudoers“ a přidáme uživatele, jeho práva a specifikaci „ALL= (ALL) ALL“.
- Povolíme seznam internetových zdrojů v souboru „sources.list“. Odstraníme komentáře z řádků končících slovem „universe“.
- Provedeme aktualizaci zdrojů zadáním (sudo apt-get update) do příkazové řádky.
- Ustavíme terminálové spojení na SSH server pomocí např. „PuTTY“.
- Instalujeme kompilátor „apt-get install gcc“.
- Provedeme aktualizace „apt-get update“.
- Spustíme upgrade serveru Ubuntu „apt-get upgrade“ a provedeme reboot.

7.1.2 Definice pojmů

VMware vSphere Hypervisor™ (ESXi) – virtualizační architektura společnosti VMware.

LAMP – Zkratka aplikací Linux, Apache, MySQL, PHP, Perl, Python.

OpenSSH - svobodná verze SSH komunikačních nástrojů.

Sudo – (Super User DO), příkaz UNIX pro vykonání úkonů pod nejvyšším oprávněním.

PuTTY – svobodný SSH klient pro operační systémy Microsoft Windows.

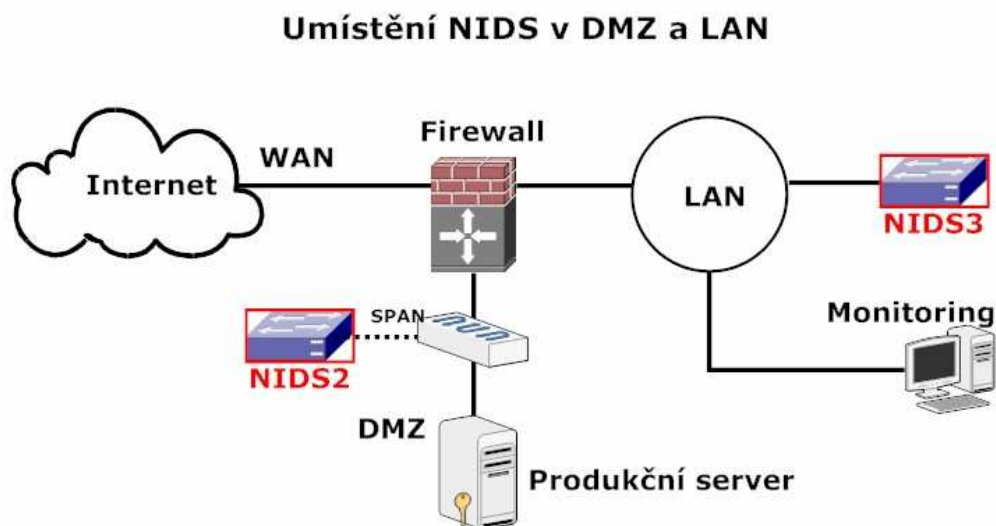
aptitude - vysokoúrovňové rozhraní k balíčkovacímu systému Debian – Ubuntu.

7.2 Postup instalace a konfigurace interní sondy Snort 2.8

Jak již bylo řečeno v kapitole 6.2 NIDS Snort je určen k detekci průniků a pracuje na principu porovnávání reálného toku se svou databází signatur. Snort může pracovat v režimech sniffer, packet logger a IDS. V našem případě využijeme režim NIDS, několik

síťových sond s vlastní databází záznamů o útocích. K tomu je zapotřebí dostatečný HW potenciál, zejména diskový prostor alespoň 8GB na NIDS. Sondy NIDS zpravidla obsahují dvě síťové rozhraní, kde jedno slouží pro správu systému a druhé pracuje v promiskuitním režimu viz obr.1 kap.3.2.2. Tento režim umožňuje být „neviditelným“, neboť rozhraní nemá IP adresu a zároveň zachytává veškerý provoz v síťovém uzlu. Avšak vzhledem k tomu, že se již výlučně používá přepínaná síť a pakety neproudí všemi směry jako tomu bývalo u zařízení HUB, musíme v monitorovaném uzlu instalovat network TAP viz obr.2, kap.3.3.2 nebo nastavit na přepínači zrcadlený port (SPAN).

Interní NIDS lze umístit v libovolné části podnikové sítě, kde bude monitorovat nežádoucí jevy. Jedna sonda zajistí přehled dění v adresním rozsahu 10.13.0.0/16 a druhá v DMZ. Orientační názvy jsou snortlan1 s IP adresou 10.13.1.142 a snortdmz1 s veřejnou IP adresou 89.81.91.150.



Obr. 8: Rozmístění detekčních sond v DMZ a LAN ^[20]

7.2.1 Konfigurace HW operačního systému

Přejmenování Ubuntu serveru provedeme ve dvou souborech: /etc/hosts, hostname.

- Editujeme konfigurační soubor /etc/network/interfaces a nastavíme IP adresu a rozhraní.
- Editujeme konfigurační soubor /etc/hosts, /etc/hostname a nastavíme IP adresu a jméno serveru.

7.2.2 Snort.conf

- V příkazovém řádku serveru spustíme instalaci podpůrných knihoven pro Snort: `apt-get install libpcap0.8-dev libmysqlclient15-dev bison flex libapache2-mod-php5 php5-gd php5-mysql libtool libpcre3-dev php-pear.`
- Vytvoříme si pracovní adresář pro instalační soubory stažené z internetu. např. `/tempnort.`
- Stáhneme vizualizační utilitu BASE (Basic Anylysis and Security Engine): `wget http://downloads.sourceforge.net/project/secureideas/BASE/base-1.4.5/base-1.4.5.tar.gz?use_mirror=voxel.`
- Uložíme malou, rychlou databázovou knihovnu ADOdb 4991 pro PHP4 a 5: `wget http://downloads.sourceforge.net/project/adodb/adodb-php-4-and-5/adodb-4991-for-php/adodb4991.tgz?use_mirror=voxel.`
- Stáhneme nástroj ke zpracování výstupů z aplikace Snort, která provádí zpracování paketů: `wget http://www.securixlive.com/download/barnyard2/barnyard2-1.7.tar.gz.`

7.2.3 Snort

- Spustíme instalaci APT²¹ balíčku Snort: `apt-get install snort-mysql.`
- Po výzvě vložíme síťový rozsah, ve kterém bude sonda umístěna např. `10.13.0.0/16.`

7.2.4 MySQL

- vytvoříme databázi Snort v MySQL serveru: `mysql -u root -p, grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.* to snort@localhost;.`

²¹ APT (Advanced Packaging Tool) – balíčkovací systém Debian GNU/Linux pro usnadnění, automatizaci vyhledávání, instalací a odinstalací aktuálního software přímo z internetu. Je používán ve všech odnožích Linux Debian. Zdroj: [http://cs.wikipedia.org/wiki/Advanced_Packaging_Tool].

- Vstoupíme do adresáře s šablonou struktury databáze: `cd /usr/share/doc/snort-mysql/`.
- Importujeme schéma databáze: `zcat create_mysql.gz | mysql -u root -p snort`.
- Upravíme „snort.conf“: `nano /etc/snort/snort.conf`, najdeme „var HOME_NET any“ a nahradíme hodnotou „var HOME_NET \$eth0_ADDRESS“, vyhledáme „Output log_tcpdump: tcpdump.log“ a řádek zakomentujeme #, najdeme řádek „output log_unified“ a vložíme pod něj hodnotu „output unified2: filename snort.log, limit 128“.

7.2.5 BASE a ABODB

- Nastavíme utility BASE a ABODB pomocí PHP scriptu PEAR: `pear install --alldeps Mail`, `pear install --alldeps Mail_Mime`, `pear install --alldeps Image_Canvas-0.3.2`, `pear install --alldeps Image_Graph-0.7.2`.
- Vstoupíme do adresáře /tempnort a rozbalíme Adodb a Base: `tar -zxvf adodb4991.tgz`, `tar -zxvf base-1.4.5.tar.gz`.
- Přesuneme soubory do adresáře /var/www : `mv adodb /var/www`, `mv base-1.4.5 /var/www`.

7.2.6 PHP a Apache2

- Provedeme úpravu konfiguračního souboru /etc/php5/apache2/php.ini: Vyhledáme „Dynamic Extensions“ a přidáme 2 řádky „extension=mysql.so“, „extension=gd.so“, vyhledáme „error_reporting = E_ALL & ~E_DEPRECATED“ a nahradíme „error_reporting = E_ALL & ~E_NOTICE“.
- Editujeme „/etc/apache2/apache2.conf“ a na konec souboru přidáme řádek „servername snortlan.local“. Pak restartujeme ovládací rozhraní: „`apache2ctl restart`“.
- Vstoupíme do adresáře „/var/www“, vytvoříme link „`ln -s base-1.4.5 ./base`“, nastavíme oprávnění „`chmod a+w base`“.
- Spustíme prohlížeč a vložíme adresu odkazu `http://10.13.1.142/base` a nastavíme grafické rozhraní Base: path to adodb to /var/www/adodb, Database Name=snort,

Database Host=localhost, Database User=snort, Database Password=
"heslo_databáze_snort", nastavíme oprávnění „chmod og-w base“.

7.2.7 Barnyard

- Provedeme kompilaci Barnyard: „./configure --with-mysql && make && make install“, zkopírujeme konfigurační soubor „cp etc/barnyard2.conf /etc/snort“, vytvoříme adresář „mkdir /var/log/barnyard2“.
- Editujeme soubor /etc/snort/barnyard2.conf: Přidáme řádky „config hostname: localhost“, „config interface: eth0“, „output database: alert, mysql, user=snort password= heslo_databáze_snort dbname=snort host=localhost“.
- Spustíme Snort a Barnyard: snort -c /etc/snort/snort.conf -i eth0, odpovědí by měla být hláška „Not Using PCAP_FRAMES“.
- Otevřeme novou relaci terminálu „PuTTY“ a vypíšeme obsah logovacího adresáře: „ls -la /var/log/snort“ a vyhledáme nejnovější soubor „snort.log.1324654287“ s desetimístnou číselnou příponou, kterou zkopírujeme.
- Vytvoříme soubor „nano /var/log/snort/barnyard.waldo“ a vložíme do něj text, každý údaj na samostatný řádek: /var/log/snort, snort.log, 1324654287, 0.
- Do příkazové řádky zadáme (vše do jednoho řádku): „/usr/local/bin/barnyard2 -c /etc/snort/barnyard2.conf -G /etc/snort/gen-msg.map -S /etc/snort/sid-msg.map -d /var/log/snort -f snort.log -w /var/log/snort/barnyard.waldo“. Dojde tak ke spuštění IDS a zachytávání komunikace. Správnou funkci ověříme např. analyzátozem Nmap, skenováním serveru (nmap -T4 -A -v 10.13.1.142). Výsledek se projeví záznamem na <http://10.13.1.142/base>.
- Restartujeme server a přejmenujeme db-pending-config: „mv /etc/snort/db-pending-config /etc/snort/db-pending-config.orig“. Až poté je možno spustit démona /etc/init.d/snort start.
- Vytvoříme soubor „barnyard2.conf“ pro automatické spuštění aplikace po startu PC viz. příloha1.

7.2.8 Testování Snort 2.8

- Opět ověříme funkci na adrese `http://10.13.1.142/base`, provedeme test například analyzátozem Nmap, skenováním serveru (`nmap -T4 -A -v 10.13.1.142`).

7.2.9 Export virtual appliance

- Funkční virtuální server ve VMware vSphere exportujeme jako Virtual Appliance do formátu „nazev.ovf“ a uložíme ho na sdílené úložiště pro potřeby klonování.

7.3 Klonování interní sondy Snort 2.8

Z předchozí kapitoly vyplývá, že správné nastavení NIDS si vyžaduje značné úsilí a přesné dodržení postupů v konfiguračních souborech. Proto nyní využijeme výhod virtualizace a NIDS umístěné v DMZ si naklonujeme.

7.3.1 Import virtual appliance

- Ve VMware vSphere provedeme import Appliance a přidáme virtuální síťový adaptér, přejmenujeme server a nastavíme správné „eth“ rozhraní v souboru `70-persistent.net-rules: /etc/udev/rules.d/70-persistent.net-rules`.

7.3.2 Konfigurace HW operačního systému

- Editujeme konfigurační soubor `/etc/network/interfaces` a nastavíme novou IP adresu a rozhraní.
- Editujeme konfigurační soubor `/etc/hosts`, `/etc/hostname` a nastavíme novou IP adresu a jméno serveru a provedeme `reboot` a ověříme funkci testem: `www://nováIP/base`.

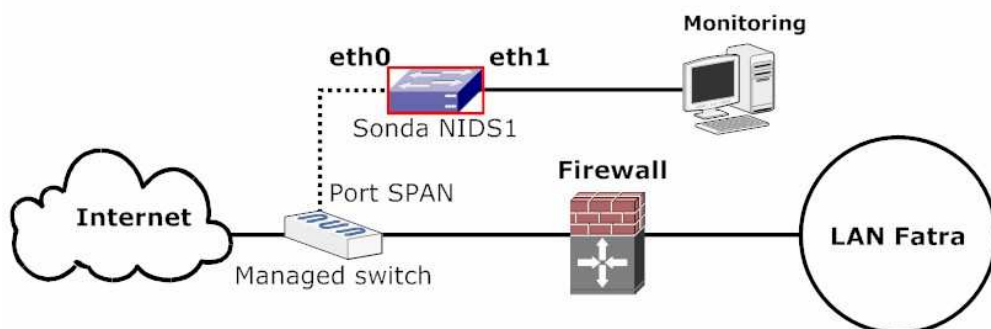
Basic Analysis and Security Engine (BASE)				
Home Search				
[Back]				
Queried on: Tue April 19, 2011 20:51:07				
Meta Criteria	any			
IP Criteria	any			
Layer 4 Criteria	none			
Payload Criteria	any			
Displaying 15 Most Frequent IP addresses				
< Src IP address >	Sensor #	< Total # >	< Unique Alerts >	< Dest. Addr. >
<input type="checkbox"/> 88.81.91.139	1	5518	3	156
<input type="checkbox"/> 91.207.208.233	1	596	1	1
<input type="checkbox"/> 74.212.251.106	1	304	1	1
<input type="checkbox"/> 84.253.142.220	1	215	1	1
<input type="checkbox"/> 88.81.235.178	1	186	1	1
<input type="checkbox"/> 213.209.196.46	1	175	1	1
<input type="checkbox"/> 217.86.132.232	1	143	1	1
<input type="checkbox"/> 77.67.57.72	1	120	1	1
<input type="checkbox"/> 201.44.84.10	1	112	1	1
<input type="checkbox"/> 88.86.122.153	1	101	2	1
<input type="checkbox"/> 61.19.252.96	1	101	2	1
<input type="checkbox"/> 58.211.72.43	1	101	2	1
<input type="checkbox"/> 76.164.231.116	1	101	2	1
<input type="checkbox"/> 118.129.154.165	1	101	2	1
<input type="checkbox"/> 178.63.20.131	1	100	2	1
<input type="button" value="{ action }"/> ACTION <input type="button" value="Selected"/> <input type="button" value="ALL on Screen"/>				
[Loaded in 0 seconds]				
Alert Group Maintenance Cache & Status Administration				
BASE 1.4.5 (llias) by Kevin Johnson and the BASE Project Team				
Built on ACID by Roman Danyšew				

Obr. 9: Ukázka grafického reportu z aplikace BASE [21]

7.4 Postup instalace externí sondy Snort 2.9 a Snort Report

Implementace NIDS si vyžaduje zvýšenou pozornost při nastavení pravidel na firewallu nebo monitoring oddělený od sítě. Externí NIDS budou umístěny v kritických částech síťové struktury před firewallem, přitom ale budou dosažitelné z vnitřní sítě. NIDS obsahuje dvě síťové rozhraní, jedno pro sběr dat a druhé pro správu a monitorování. Server NIDS je připojen v promiskuitním režimu (eth0) k vnějšímu rozhraní firewallu pomocí SPAN portu. Takové rozhraní nemá vlastní IP adresu a proto je před vetřelci skryta.

Umístění systému NIDS před firewallem



Obr. 10: Detekční sonda umístěná před firewallem [20]

7.4.1 Konfigurace HW operačního systému

- Před instalací aplikace Snort vložíme do systému druhou virtuální síťovou kartu potřebnou k vytvoření promiskuitního módu. Editujeme virtuální stanici a přidáme síťové rozhraní, které je připojeno k zrcadlenému uzlu sítě.
- V konfiguračním souboru „/etc/udev/rules.d/70-persistent.net.rules“ nastavíme rozhraní eth0 a eth1.
- Definujeme promiskuitní mód síťového rozhraní eth0, provedeme editaci souboru /etc/network/interfaces a přidáme do řádků následující záznamy: „auto eth0, iface eth0 inet, manual, ifconfig eth0 up“.
- Restartujeme síť /etc/init.d/networking restart.
- Do souboru /etc/rc.local přidáme záznam „ifconfig eth0 up“ pro spuštění síťového rozhraní eth0 po startu.
- Přejdeme k instalaci Ubuntu apt balíčků a knihoven potřebných pro chod Snort 2.9 :
“apt-get install nmap nbtscan apache2 php5 php5-mysql php5gd libpcap0.8dev libpcr3dev g++ bison flex libpcapruby mysqlserver libmysqlclient16dev“.

7.4.2 JpGraph

- Přehledy incidentů zpracovává grafická utilita Snort Report spolu s PHP utilitou pro tvorbu grafů „JpGraph“: Vstoupíme do adresáře pro dočasné soubory /tempsnort, stáhneme „JpGraph“ wget <http://hem.bredband.net/jpgraph/jpgraph-1.27.1.tar.gz>.
- Vytvoříme adresář: mkdir /var/www/jpgraph.
- Rozbalíme zdrojový kód a zkopírujeme data: tar zxvf jpgraph1.27.1.tar.gz, cp -r jpgraph-1.27.1/src /var/www/jpgraph/.

7.4.3 Snort Report

- Stáhneme aktuální verzi Snort Report z adresy <http://www.symmetrixtech.com>.
- Rozbalíme tar zxvf snortreport-1.3.1.tar.gz -C /var/www/.

- Upravíme konfigurační soubor „srconf.php“: Editujeme „/var/www/snortreport1.3.1/srconf.php“: Vyhledáme \$pass = "HesloDbMySQL"; , vyhledáme záznam: define("JPGRAPH_PATH", "../jpgraph/"); a nahradíme jej záznamem: define("JPGRAPH_PATH", "../jpgraph/src/"); , vyhledáme záznamy: define("NMAP_PATH", "/usr/local/bin/nmap -v"); define("NBTSCAN_PATH", "/usr/local/bin/nbtscan"); a nahradíme je záznamy: define("NMAP_PATH", "/usr/bin/nmap -v"); define("NBTSCAN_PATH", "/usr/bin/nbtscan");.

7.4.4 DAQ

- Instalace DAQ (Data Acquisition API) – služba odběru vzorků signálů. Stáhneme zdrojový kód <http://www.snort.org/downloads/> 860, rozbalíme do adresáře „/tempnort“ `tar zxvf daq-0.5.tar.gz`, `cd daq-0.5`, `sudo ./configure`, `make`, `make install`, `ldconfig`.

7.4.5 Libdnet

- Instalace zjednodušeného interface k několika síťovým low-level rutinám „libdnet“: Stáhneme zdrojový kód z [www](http://www.libdnet.googlecode.com/files/libdnet-1.12.tgz) „wget <http://libdnet.googlecode.com/files/libdnet-1.12.tgz>“, rozbalíme „tar zxvf libdnet-1.12.tgz“, vstoupíme do adresáře „cd libdnet1.12“ a provedeme kompilaci kódu „./configure, make, make install“.

7.4.6 Instalace Snort

- Přejdeme k instalaci Snort, stáhneme do pracovního adresáře „/tempnort“ aktuální verzi Snort: <http://www.snort.org/snort-downloads> (verze 2.9.0.5 se nachází na [www](http://www.snort.org/downloads/867) „<http://www.snort.org/downloads/867>“).
- Rozbalíme zdrojový kód: „tar zxvf snort-2.9.0.5.tar.gz“, vstoupíme do „cd snort-2.9.0.5“, zkonfigurujeme „./configure --prefix=/usr/local/snort --enable-ipv6 --enable-gre --enable-mpls --enable-targetbased --enable-decoder-preprocessor-rules --enable-ppm --enable-perfprofiling --enable-zlib --enable-active-response --enable-normalizer --enable-reload --enable-react --enable-flexresp3“.

- Zkompilujeme: „make, make install“, vytvoříme adresář „/var/log/snort“, „/var/snort“, přidáme skupinu „groupadd snort“, uživatele „useradd --g snort snort“, vlastníka „chown snort:snort /var/log/snort“.

7.4.7 MySQL

- MySQL vytvoříme databázi: „echo „create database snort;“ | mysql -u root -p“, vložíme schéma „mysql -u root -p -D snort < ./schemas/create_mysql“, nastavení oprávnění: „echo \"grant create, insert, select, delete, update on snort.* to snort@localhost identified by 'password'\" | mysql -u root -p“.

7.4.8 Snort.conf

- Konfigurujeme „/usr/local/snort/etc/snort.conf“, změníme následující záznamy: „dynamicpreprocessor directory /usr/local/lib/snort_dynamicpreprocessor/“ na „dynamicpreprocessor directory /usr/local/snort/lib/snort_dynamicpreprocessor/“, „dynamicengine /usr/local/lib/snort_dynamicengine/libsf_engine.so“ na „dynamicengine /usr/local/snort/lib/snort_dynamicengine/libsf_engine.so“, „dynamicdetection directory /usr/local/lib/snort_dynamicrules“ na „dynamicdetection directory /usr/local/snort/lib/snort_dynamicrules“, zakomentujeme „unlimited_decompress“, „preprocessor mormalize_ip,tcp...“.
- Aktuální signatury a pravidla Snort získáme ze <https://www.snort.org/snort-rules>, rozbalíme je do /usr/local/snort: „tar zxvf snortrulesnapshot-2905.tar.gz -C/usr/local/snort“, vytvoříme adresář „mkdir /usr/local/snort/lib/snort_dynamicrules“, zkopírujeme pravidla: „cp /usr/local/snort/so_rules/precompiled/Ubuntu-10-4/i386/2.9.0.5/* /usr/local/snort/lib/snort_dynamicrules“.

7.4.9 Barnyard2

- Barnyard je utilita, která čte logovací záznam snortu a vkládá jej do databáze MySQL. Snižuje celkovou zátěž Snort a urychluje zpracování výsledků. Verzi 2-1.8 stáhneme z internetu: <http://www.securixlive.com/download/barnyard2/barnyard2-1.8.tar.gz>.

- Provedeme konfiguraci a kompilaci kódu: Rozbalíme kód “tar zxvf barnyard2-1.8.tar.gz“, vstoupíme do „cd barnyard2-1.8“, konfigurujeme „./configure --with-mysql“, kompilujeme „make, make install“, kopírujeme konfigurační soubor „cp etc/barnyard2.conf /usr/local/snort/etc“, vytvoříme logovací adresář „mkdir /var/log/barnyard2“, nastavíme oprávnění „chmod 666 /var/log/barnyard2“, zpřístupníme aktualizace a modifikace času na souboru „touch /var/log/snort/barnyard2.waldo“, nastavíme vlastníka souboru „chown snort.snort /var/log/snort/barnyard2.waldo“.
- Upravíme záznamy v souboru „./usr/local/snort/etc/barnyard2.conf“, vyhledáme řádky: „config reference_file: /etc/snort/reference.config“, „config classification_file: /etc/snort/classification.config“, „config gen_file: /etc/snort/genmsg.map“, „config sid_file: /etc/snort/sidmsg.map“, „config reference_file: /usr/local/snort/etc/reference.config“, „#config hostname: thor“, „#config interface: eth0“, „#output database: log, mysql, user=root password=test dbname=db host=localhost“ a nahradíme je „config reference_file: /usr/local/snort/etc/reference.config“, „config classification_file: /usr/local/snort/etc/classification.config“, „config gen_file: /usr/local/snort/etc/genmsg.map“, „config sid_file: /usr/local/snort/etc/sidmsg.map“, „config hostname: localhost“, „config interface: eth1“, „output database: log, mysql, user=snort password=HESLOMYSQL dbname=snort host=localhost“.

7.4.10 Snort.conf

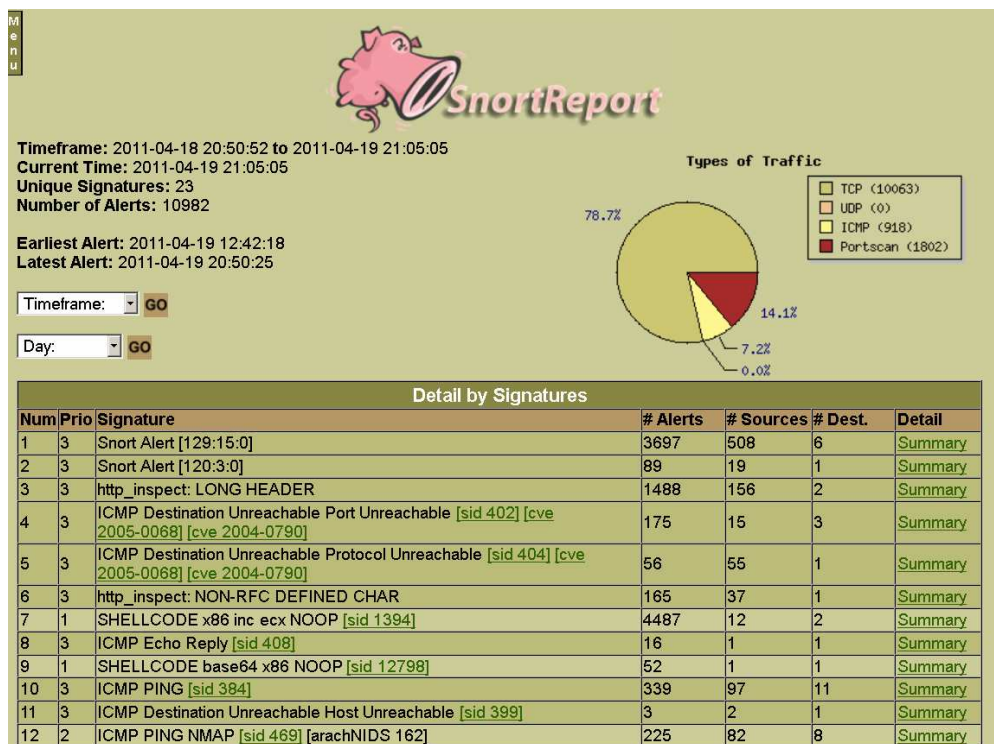
- Editujeme konfigurační soubor „./usr/local/snort/etc/snort.conf“: Vyhledáme záznamy „dynamicpreprocessor directory /usr/local/lib/snort_dynamicpreprocessor“, „dynamicengine /usr/local/lib/snort_dynamicengine/libsf_engine.so“, „dynamicdetection directory /usr/local/lib/snort_dynamicrules“ a nahradíme je za „dynamicpreprocessor directory /usr/local/snort/lib/snort_dynamicpreprocessor“, „dynamicengine /usr/local/snort/lib/snort_dynamicengine/libsf_engine.so“, „dynamicdetection directory /usr/local/snort/lib/snort_dynamicrules“. Přidáme záznam „output unified2: filename snort.u2, limit 128“.

7.4.11 Testování Snort 2.9

- Provedeme test NIDS Snort: Z příkazové řádky spustíme `./usr/local/snort/bin/snort -u snort -g snort -c /usr/local/snort/etc/snort.conf -i eth0`. Zdařilá instalace vypíše hlášku „Commencing packet processing“.

7.4.12 Automatické spuštění Snort 2.9

- Ukončíme test Ctrl+C a nastavíme automatické spuštění Snort po startu počítače: Provedeme editaci `./etc/rc.local` a pod záznam `„ifconfig eth0 up“` přidáme následující řádky: ^[11]
- `/usr/local/snort/bin/snort -D -u snort -g snort \`
- `-c /usr/local/snort/etc/snort.conf -i eth0`
- `/usr/local/bin/barnyard2 -c /usr/local/snort/etc/barnyard2.conf \`
- `-G /usr/local/snort/etc/gen-msg.map \`
- `-S /usr/local/snort/etc/sid-msg.map \`
- `-d /var/log/snort \`
- `-f snort.u2 \`
- `-w /var/log/snort/barnyard2.waldo \`
- `-D`
- Uložíme a nastartujeme script: `./etc/init.d/rc.local start`.
- Ověříme funkci přes www browser: `„http://10.13.1.135/snortreport-1.3.1/alerts.php“`.



Obr. 11: Ukázka grafického výstupu z aplikace SnortReport ^[21]

7.4.13 Nastavení výjimek

Záznamy aktivit na vnější straně firewallu se vyznačují častými falešnými poplachy. Vysoký nárůst alarmů můžeme eliminovat přidáním výjimek do konfiguračního souboru `/etc/snort/rules/local.rules`.

7.4.14 Příklad nastavení výjimky

- Pokud chceme odstranit falešné poplachy vyvolané TCP protokolem z důvěryhodného serveru s IP adresou 85.32.16.3, přidáme do záznamy do „local.rules“.
- `pass tcp 85.32.16.3 any -> any any (sid:1000001 ;)`
- `pass tcp any any -> 85.32.16.3 any (sid:1000002 ;)`

Podobným způsobem nastavíme výjimky pro UDP, ICMP protokoly.

7.4.15 Vytváření pravidel

Výhodou Snortu je možnost definice vlastních pravidel což umožní přizpůsobení konkrétnímu místu instalace. Je schopen operovat s protokoly TCP, ICMP a UDP. Pravidla

jsou tvořeny hlavičkami, nastavením volby a z popisné zprávy zapsané do logovacího souboru. V hlavičce se systém dozvídá jakou akci bude provádět, jestli se zaznamená poplach, ignoruje paket, jen zaloguje nebo zahodí viz tab.2. Nastavením portů, adres a operátorů <-> určujeme směr a podmínky komunikace. V popisu volby pravidla (nastavení rolí) se nachází podstatná obsahová část rozhodovacího mechanismu, klasifikace výstrah, příznaky paketu, upřesní formy záznamu viz tab.3.

Akce	Popis akce pravidla
Pass	ignoruje paket
log	zapiše do logu
alert	generuje výstrahu a zapiše
activate	generuje výstrahu a vyvolá další pravidlo "dynamic"
dynamic	je vyvolán pouze pravidlem "activate"
drop	přidá do Iptables pravidlo zahození a zapiše
reject	přidá do Iptables pravidlo zahození a zapiše a vyšle TCP reset
sdrop	přidá do iptables pravidlo o zahození paketu a neprovede záznam

Tab. 2: Volby nastavení akce v definicích pravidel ^[22]

7.4.16 Logický pohled na pravidlo Snortu

- akce - protokol - zdrojová IP adresa – zdrojový port – směr toku dat -> cílová IP adresa - cílový port (nastavení).

7.4.17 Definice adres

- Jakákoliv adresa je vyjádřena slovem „any“, rozsahy adres zapišeme ve tvaru CIDR. Např.192.168.20.0/21. Výjimky se uvádí negací „!“ vykřičníkem !192.168.20.0/21 (všechno mimo daný rozsah).

7.4.18 Definice portů

- Rozsahy zdrojových i cílových portů lze definovat operátorem „:“ dvojtečka a pro neomezený rozsah příkazem „any“. Pravidlo „alert tcp 93.56.0.0/16 any -> 85.32.16.0/29 :22“ nám říká, že bude vytvořen záznam „alert“, pokud přichází TCP požadavek z rozsahu 93.56.0.0/16 na rozsah 85.32.16.0/29 na portu menším nebo rovno 22. Podobným způsobem se určí větší nebo rovno portu (22:) nebo rozsah portů (10:22).

7.4.19 Směr toku dat

- Operátor `->`, `<-`, `<->` určuje sledovaný směr komunikace. Záznam „log udp any 53 <-> any 53“ nařizuje zápis do logu, pokud bude odkudkoliv kamkoliv probíhat komunikace na UDP portu 53.

7.4.20 Nastavení rolí

- Hlavní částí IDS je detekční mechanismus, který rozhoduje o osudu testovaného paketu. Mechanismus je uskupením porovnávacích definic a parametrů. Jednotlivé volby pravidel jsou od sebe odděleny středníkem“;“ a jejich argumenty dvojtečkou „:“.

Volba pravidla	Argument	Význam
msg	text zprávy	Zpráva
reference	id systému	informace o umístění signatury
sid	id pravidla snortu	identifikace pravidla
rev	číslo	revize
classtype	jméno	klasifikace události
priority	číslo	vážnost události
logto	jméno souboru	kam se uloží log
ttl	číslo	životnost paketu - test podle hlavičky
id	číslo	test ID z hlavičky
dsize	číslo	>< a velikost paketu
content	řetězec	hledá paket podle vzoru
nocase	-	velká/malá písmena
offset	číslo	nastavení počáteční vyhledávací pozice
depth	číslo	hloubka prohledávání paketu
flags	příznaky	test podle příznaků v TCP hlavičce
seq	číslo	test TCP sekvence
itype	číslo	test ICMP

Tab. 3: Jmenný seznam atributů z definic a pravidel pro nastavení rolí ^[22]

7.4.21 Příklad vytvoření pravidla

- Zachycení komunikace průzkumného paketu s řetězcem "ISSPNGRQ", na portu ICMP 8 vyvolá alert a zapíše zprávu "ICMP ISS Pinger“:
- `alert icmp any any -> any any (msg:"ICMP ISS Pinger"; itype:8; content:"ISSPNGRQ"; depth:32; reference:arachnids,158; classtype:attempted-recon; sid:465; rev:4;).`

- Zachycení komunikace na TCP portu 22 s obsahem „user root“.
- alert tcp any any -> any any 22 (content:"user root");).

7.4.22 Automatická aktualizace Oinkmaster

Každý bezpečnostní prvek musí umět reagovat na dynamiku vývoje hrozeb. V systému IDS Snort můžeme aktualizovat ručně nebo automaticky pomocí aplikace „Oinkmaster“. Ruční obsluha je časově náročná a dříve nebo později na aktualizace zapomeneme. Proto raději rovnou počítejme s bezobslužným systémem.

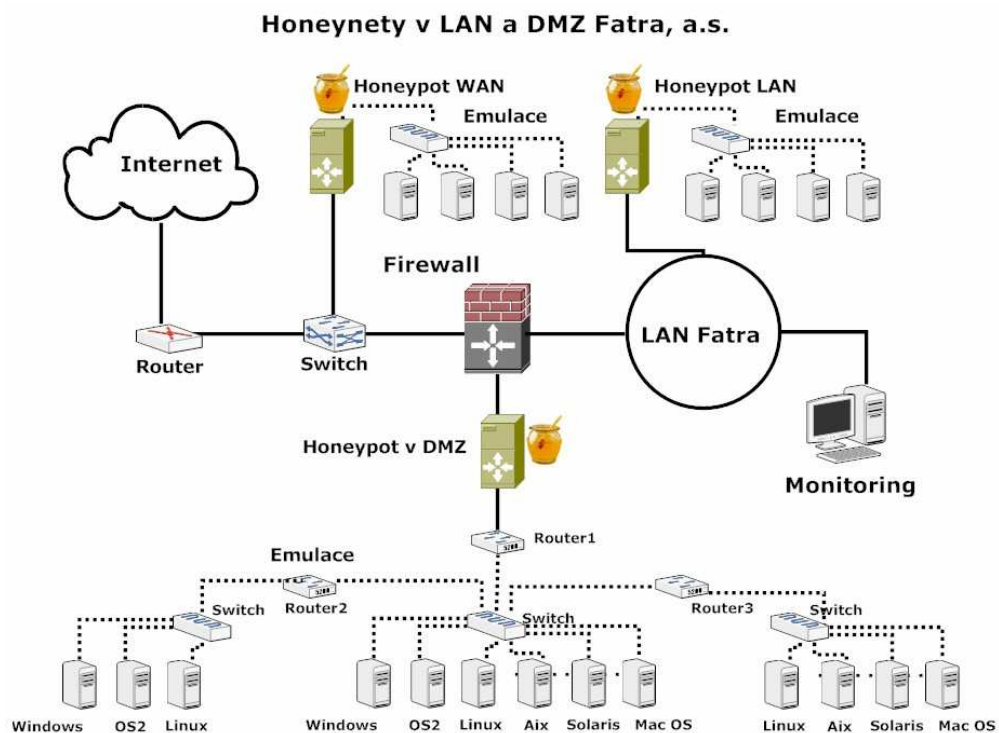
Oinkmaster je skript pro usnadnění kontroly a správy pravidel Snort. Je šířen pod BSD licencí a funguje na většině platforem, ve kterých lze spouštět Perl skripty, např. Linux, *BSD, Windows, Mac OS X, Solaris, atd.

Nejdříve je nutno provést registraci na domovských stránkách www.snort.org. Získáme tím přístup k souborům a nárok na obnovu ve 30 denním intervalu. Pokud chceme získávat aktuální signatury ihned po uvolnění, musíme si zaplatit podporu. Správu aktualizací nám zajistí jednoduchý Perlův skript Oinkmaster. Po jeho rozbalení na disk provedeme nastavení konfiguračního souboru „oinkmaster.conf“. Nejdůležitějším krokem je vložení následujícího řetězce : `#url = http://www.snort.org/pub-bin/oinkmaster.cgi/ náš oinkcode/snortrules-snapsh ot-x.x.tar.gz`. Oinkcode získáme po přihlášení na <https://www.snort.org/account/oinkcode> pod správným účtem. ^[10]

8 IMPLEMENTACE HONEYPOTU HONEYD

Každý Honeyd server bude obsahovat jedno síťové rozhraní, za kterým se bude rozprostírat jeho virtuální síťová infrastruktura s emulací služeb serverů a jiných aktivních prvků. Honeypoty budou umístěny ve vnější i vnitřní části sítě.

Instalaci Honeyd lze provést více způsoby. Máme možnost stáhnout zdrojové kódy²² a provést vlastní kompilaci, stáhnout předkompilovanou verzi²³ nebo použít balíčkovací systém APT. Tyto postupy jsou uvedeny v následujících kapitolách.



Obr. 12: Struktura instalovaných nástrah honeypotů v podnikové síti Fatra, a.s. [20]

8.1 Konfigurace systému

Operačním systémem pro Honeyd je opět Linux Ubuntu Server 10.04. Jelikož se stále pohybujeme ve virtualizovaném prostředí, můžeme instalaci provést jednoduše pomocí

²² Zdrojové kódy: [<http://www.honeyd.org/release.php>].

²³ Zdrojové kódy: [http://www.citi.umich.edu/u/provos/honeyd/honeyd_kit-1.0c-a.tgz].

importu appliance. Po importu je nutno provést rutinní kroky přejmenování, změnu internetové a hardwarové adresy.

- Přejmenování Ubuntu serveru provedeme ve dvou souborech: `/etc/hosts`, `hostname`.
- Editujeme konfigurační soubor `/etc/network/interfaces` a nastavíme novou IP adresu a rozhraní.
- Editujeme konfigurační soubor `/etc/hosts`, `/etc/hostname` a nastavíme novou IP adresu a jméno serveru.

8.1.1 Kompilace Honeyd 1.5c

- První dva způsoby instalace vyžadují doplnění podpůrných knihoven do systému `libevent` [<http://www.monkey.org/~provos/libevent/>], `libdnet` [<http://libdnet.sourceforge.net/>] a `libpcap` [<http://www.tcpdump.org/>], `zlib1g-dev` [<http://packages.debian.org/sid/zlib1g-dev>].
- Stáhneme zdrojové kódy: `wget http://www.honeyd.org/uploads/honeyd-1.5c.tar.gz`, rozbalíme soubory `tar -xvf honeyd-1.5c.tar.gz`, vstoupíme do adresáře `cd honeyd-1.5c`, `./configure`, `make`, `make install`.
- Spuštění Honeyd: `/usr/local/bin/honeyd -d -f /usr/local/share/honeyd/honeyd.conf -p /usr/local/share/honeyd/nmap.prints -x /usr/local/share/honeyd/xprobe2.conf -a /usr/local/share/honeyd/nmap.assoc --disable-webserver '192.168.24.20-192.168.24.28'`.
- Nastavení `honeyd.conf` viz. kap. 8.1.4.

8.1.2 Předkompilovaná verze - Honeyd Kit

- Doplníme knihovny: `libevent` [<http://www.monkey.org/~provos/libevent/>], `libdnet` [<http://libdnet.sourceforge.net/>] a `libpcap` [<http://www.tcpdump.org/>], `zlib1g-dev` [<http://packages.debian.org/sid/zlib1g-dev>].
- Stáhneme kompilát: `wget http://www.citi.umich.edu/u/provos/honeyd/honeyd_kit-1.0c-a.tgz`. Rozbalíme soubory `tar -xvf honeyd_kit-1.0c-a.tgz` do `/etc/honeyd`.
- V této verzi jsou dva spouštěcí skripty, které je nutno upravit podle našich potřeb: Skript `start-arpd.sh` určuje síťový rozsah, na kterém Honeyd naslouchá `./arpd`

192.168.1.0/24“. Ve scriptu „start-honeyd.sh“ definujeme zdroj instrukcí a chování systému: „./honeyd -f honeyd.conf -p nmap.prints -x xprobe2.conf -a nmap.assoc -0 pf.os -l /var/log/honeyd 192.168.1.100-192.168.1.253“.

- -f -Určení konfiguračního souboru.
- -p -Fingerprints - Honeyd se snaží věrně emulovat zdroje, proto musí na útočníka přesvědčivě reagovat vhodnými pakety. Fingerprints jsou osobnosti operačních systémů, které lze nalézt v souborech nmap.prints a xprobe2.conf.
- -x -Soubor s otisky prstů stylu „xprobe“. Tento soubor určuje, jak Honeyd reaguje na ICMP otisky prstů.
- -a -Soubor, který sdružuje nmap a xprobe styly otisky prstů.
- -0 -Dynamická šablona databáze otisků.
- -l -Určení místa uložení logů.
- Pracovní rozsah, na kterém naslouchá nastavíme podle potřeby.
- Nastavení *honeyd.conf* viz. kap. 8.1.4.
- Samotné spuštění Honeyd zajistíme spuštěním dvou skriptů „start-arpd.sh a start-honeyd.sh“.

8.1.3 APT balíček Honeyd

- Snazší variantou je instalace pomocí balíčkovacího systému APT. Spustíme příkaz „apt-get install honeyd“. Spolu s balíčkem honeyd se automaticky nabídne instalace aplikací a souvisejících knihoven: defoma, farpd, fontconfig, honeyd-common, libcairo2, libdatrie1, libdirectfb-1.2-0, libdumbnet1, libevent-1.4-2, libfontenc1, libpango1.0-0, libpango1.0-common, libpixmap-1-0, libreadline5, librrd4, libsysfs2, libthai-data, libthai0, libts-0.0-0, libxcb-render-util0, libxcb-render0, libxfont1, libxft2, libxrender1, rrdtool, tsconf, ttf-dejavu, ttf-dejavu-extra, x-ttcidfont-conf, x11-common, xfonts-encodings, xfonts-utils, ttf-japanese-gothic, ttf-japanese-mincho, ttf-thryomanes, ttf-baekmuk, ttf-arphic-gbsn00lp, ttf-arphic-bsmi00lp, ttf-arphic-gkai00mp, ttf-arphic-bkai00mp, librrds-perl.

- Rozsahy simulovaných sítí, serverů, služeb a aktivních prvků definujeme v konfiguračním souboru „/etc/honeypot/honeyd.conf“: Začneme volbou rozsahů sítí, podsítí, aktivních prvků a služeb. Určíme jak se má chovat, na jaké síti má naslouchat a na co má reagovat.

8.1.4 Nastavení emulace síťové struktury a služeb v souboru „honeyd.conf“

- route entry 10.13.1.137 //brána honeypotu - adresa našeho serveru
- route 10.13.1.137 link 10.2.0.0/24 //připojíme 1.rozhraní na virtuální LAN rozsah
- route 10.13.1.137 add net 10.30.0.0/16 10.30.0.1 latency 8ms bandwidth 10Mbps
//1.rozhraní routeru, rozsah a zpoždění paketů
- route 10.13.1.137 link 10.30.0.0/24 //připojíme rozhraní Honeypotu na IP rozsah
- route 10.30.0.1 add net 10.30.1.0/24 10.30.1.1 latency 7ms loss 0.5
//2.rozhraní routeru, rozsah a zpoždění paketů
- route 10.30.1.1 link 10.30.1.0/24 //připojíme 2.rozhraní na virtuální LAN rozsah
- # Example of a simple host win_xp and its binding // jednoduchý příklad hostitele a vazeb
- create win_xp //název instance emulovaného systému
- set win_xp personality "Microsoft Windows XP Professional SP1"
//jaký operační systém budeme simulovat
- set win_xp uptime 1728650 //pomyslná doba běhu stanice
- set win_xp maxfds 35 //počet popisovačů souboru
- # For a complex IIS server // komplexní IIS server²⁴

²⁴ IIS server (Internet Information Server) - web server od Microsoft podporující služby HTTP, HTTPS, FTP, FTSP, SMTP and NNTP současná verze 7.5. Zdroj: [http://en.wikipedia.org/wiki/Internet_Information_Services].

- add win_xp tcp port 80 "sh /usr/share/honeyd/scripts/win32/web.sh"
//aktivace portů, na kterých má virtuální PC naslouchat a spouštění příslušných skriptů²⁵.
- add win_xp tcp port 22 "/usr/share/honeyd/scripts/test.sh \$ipsrc \$dport"
- add win_xp tcp port 23 proxy \$ipsrc:23
- add win_xp udp port 53 proxy 141.211.92.141:53
- set win_xp default tcp action reset *//typ odpovědí na pakety viz. kapitola 6.4.2*
- create default *//definice odpovědí na pakety*
- set default default tcp action block
- set default default udp action block
- set default default icmp action block
- create router *// vytvoření aktivního prvku*
- set router personality "Cisco 1601R router running IOS 12.1(5)"
- set router default tcp action reset
- add router tcp port 22 "/usr/share/honeyd/scripts/test.sh"
//aktivace portů, na kterých má virtuální router naslouchat a spouštění příslušných skriptů²⁶.
- add router tcp port 23 "/usr/share/honeyd/scripts/router-telnet.pl"
- bind 10.30.0.1 router *//adresa síťového rozhraní routeru*
- bind 10.30.1.1 router *//adresa síťového rozhraní routeru*

²⁵ Rozšířené skripty pro další služby Honeyd lze stáhnout z [<http://www.honeyd.org/contrib.php>].

²⁶ Rozšířené skripty pro další služby Honeyd. Zdrojové kódy: [<http://www.honeyd.org/contrib.php>].

- bind 10.30.1.12 win_xp //adresa síťového rozhraní 1.PC
- bind 10.30.1.11 win_xp //adresa síťového rozhraní 2.PC
- set 10.30.1.11 personality "Microsoft Windows NT 4.0 SP3"
//jaký operační systém budeme simulovat

8.1.5 Spuštění a test démona Honeyd (verze „APT balíček“)

- První spuštění vyžaduje změnu parametru na „RUN=yes“ a „NETWORK=10.30.0.0/16“ v konfiguračním souboru „/etc/default/honeyd“.
- Přidáme záznam „ /etc/init.d/honeyd“ do souboru „/etc/rc2.d/rc.local“ a provedeme restart serveru.
- Ověříme, jestli běží démon: Do příkazového řádku zadáme „/etc/init.d/honeyd status“, odpověď by měla být [OK].
- Po startu ověříme funkci honeypotu Honeyd, dostupnost emulací virtuálních sítí, routerů a PC stanic. Zvolíme si libovolné testovací PC v lokální síti a přidáme do jeho routovací tabulky záznam „route -p add 10.30.0.0 mask 255.255.0.0 10.13.1.142“.

8.2 Utility pro Honeyd

Veškerá zaznamenaná komunikace s honeypotem je uložena v logovacím adresáři „/var/log/honeypot“. Aktuální soubor „honeyd.log“ obsahuje záznamy v řádcích, kde každý řádek odpovídá jednomu spojení. Popisuje datum, čas, zdrojový port, typ protokolu, cílovou adresu a cílový port. Vzhledem k velkému množství je tato forma záznamu pro běžné použití nevhodná. Vhodnější je zobrazení statistik pomocí utilit, analytických nástrojů, které zpracují logovací soubory a podají čitelnější přehledy o incidentech.

8.2.1 Honeydstats

Honeyd generuje do svých logovacích souborů velké množství dat a po krátké době je velmi obtížné údaje kategorizovat. Honeydstats umožňuje rozdělení dat do čtyř kategorií.

- 1.) podle verzí operačních systémů
- 2.) podle cílového portu

3.) podle zeměpisné lokalizace původu útočníka

4.) podle spam reportu

8.2.2 Honeydsum

Honeydsum je nástroj napsaný v jazyce Perl, který generuje přehledy z logovacích záznamů honeypotu Honeyd. Výstupy lze získat v textové podobě, kde si můžeme zvolit parametry a filtry nebo v grafické podobě html stránky s odkazy na předvolené sestavy. Uspořádání dat v časové ose poskytuje přehledy nejčastějších výskytů incidentů, podle IP adres, portů, protokolů nebo počtu připojení za časový úsek. V aplikaci Honeydsum lze také korelovat výstupy z více zdrojů (honeypotů) současně.

Konfigurace „honeydsum.conf“:

Nejprve musíme provést úpravy v konfiguračních souborech „apache2“ a „honeydsum“ a poté přidat zásuvné grafické moduly Perl. Skripty Honeydsum jsou založeny na jazyku Perl, je tedy potřeba provést instalaci samotného jazyka a modulů pomocí CPAN²⁷.

- Nastavíme konfigurační soubor „honeydsum.conf“, který se nachází v adresáři „/usr/share/honeyd/scripts/honeydsum-v0.3“:
„honeyd_conf=/etc/honeyd/honeyd.conf“, „honeypot_list=10.30.1.11,10.30,
„net_list=10.130.0.0/19“, „dest_port=21,22,23,80,443“, „proto_list=tcp,udp,icmp“.

Perl - instalace:

- Z příkazové řádky spustíme „apt-get install perl“ a potvrdíme „yes“.
- Modul Perl zpracování grafů: „libgd-graph-perl“.
- Modul Perl zpracování 3D grafů: „libgd-graph3d-perl“.
- Spustíme „cpan“, potvrdíme automatickou konfiguraci „yes“ a doplníme požadované moduly:

²⁷ CPAN (Comprehensive Perl Archive Network) je archivem modulů jazyka Perl, usnadňuje instalace a práci se softwarem, obsahuje kolem 10000 modulů. Po celém světě jsou stovky mirrorů s tímto archivem, jen

- `cpan[1]> install YAML`
- `cpan[1]> install Net::Netmask`
- `make install GD::Graph::pie`
- `GD::Graph::bars`
- `GD::Graph::bars3d`

Přepínače Honeydsum:

- `-c` určení konfiguračního souboru - `honeydsum.conf`
- `-h` zobrazení helpu
- `-V` verze honeydsum
- `-w` export dat jako webová stránka
- Úpravy Perl souboru `./usr/share/honeyd/scripts/honeydsum-v0.3/honeydsum.pl` spočívají v záměně typu exportu ve skriptu Perl u pěti požadovaných obrazových souborů s příponou „png“ za „gif“.
- Z příkazového řádku spustíme skript `./honeydsum.pl -c honeydsum.conf -w /var/log/honeypot/honeyd.log`. Systém vypíše chybové hlášky: „Can't locate object method "gif" via package "GD::Image“.
- Provedeme editaci `./usr/share/honeyd/scripts/honeydsum-v0.3/honeydsum.pl`, vyhledáme řádky s příkazem konverze obrázku do „png“ a nahradíme „jpg“. Příklad: `printf(FIG_FILE "%s", $graph->gd->png);` nahradíme `printf(FIG_FILE "%s", $graph->gd->gif);`.
- V konfiguračním souboru Apache2 nastavíme alias. Přidáme záznam pro snadnější cestu k souborům přes www prohlížeč:
- Alias `/honeydsum/ "/usr/share/honeyd/scripts/honeydsum-v0.3/"`

- <Directory "/usr/share/honeyd/scripts/honeydsum-v0.3/">
- Options Indexes MultiViews FollowSymLinks
- AllowOverride None
- Order deny,allow
- Allow from all
- </Directory>
- Po restartu zadáme do prohlížeče www adresu „http://10.13.1.142/honeydsum“.

Export záznamů Honeyd do textového souboru:

- Souhrn výsledků ze záznamu převedeme do textového souboru pomocí Perl skriptu: „./honeydsum.pl -c honeydsum.conf honeyd.log >out.txt“.
- Pokud chceme vytvořit přehled jen určitého období z určitého souboru, spustíme s parametry : Příklad –„./honeydsum.pl -c honeydsum.conf honeyd.log.2011-05-15-11:00 honeyd.log.2003-11-11-00:00“.

```

### Honeyd's Configuration ###
-----
Connection Counter
-----
Total:      364
TCP:       359
UDP:        5
ICMP:       0
-----

Honeyd: 10.3.1.11
-----
Source IP
Resource Connections
10.130.10.2
21/udp      1
22/udp      1
23/udp      1
80/udp      1
443/udp     1
21/tcp     12

-----
IPs
Resource Connections
1
5      85
-----

Honeyd: 10.3.1.12
-----
Source IP
Resource Connections
10.130.10.2
21/tcp      5
22/tcp     37
23/tcp     12
80/tcp     24
443/tcp     7
-----

Top 10 Source Hosts
Rank Source IP
Connections
1 10.130.10.2
364

Top 10 Accessed Resources
Rank Resource
Connections
1 22/tcp
181
2 23/tcp
82
3 80/tcp
65
4 21/tcp
17
5 443/tcp
14
6 80/udp
1
7 21/udp
7
1

-----
Resources Connections
8 23/udp
1
9 443/udp
1
10 22/udp
1

Connections per Hour
Hour Connections
00:00 0
01:00 0
02:00 0
03:00 0
04:00 0
05:00 0
06:00 0
07:00 0
08:00 3
09:00 0
10:00 0
11:00 0
12:00 0
13:00 0
14:00 0
15:00 3
16:00 130
17:00 0
18:00 0
19:00 228
20:00 0
21:00 0
22:00 0
23:00 0

```

Obr. 13: Ukázka textového výstupu z aplikace Honeydsum ^[21]

Export záznamů Honeyd do html:

Honeydsum nabízí neméně zajímavý a lépe čitelný výstup v grafické podobě. Jedná se o html výstup a jeho zdroj se nachází v adresáři „/usr/share/honeyd/scripts/honeydsum-v0.3“.

- Vstoupíme do adresáře „/usr/share/honeyd/scripts/honeydsum-v0.3“, kde se nachází skripty.
- Spuštění skriptu s výstupem do html souboru definujeme parametrem „w“: „./honeydsum.pl -c honeydsum.conf -w /var/log/honeypot/honeyd.log“.

8.2.3 Testování Honeyd v provozu

V předchozí kapitole jsou uvedeny podrobné postupy instalace a konfigurace Honeyd.

- Nejdříve prověříme dostupnost emulovaných služeb. Spustíme testovací ping například na virtuální PC 10.30.1.11, 10.30.1.12, kde můžeme také sledovat zvýšenou latenci odezvy.
- Prověříme emulaci Cisco IOS²⁸ routeru pomocí příkazu „telnet 10.30.0.1“.

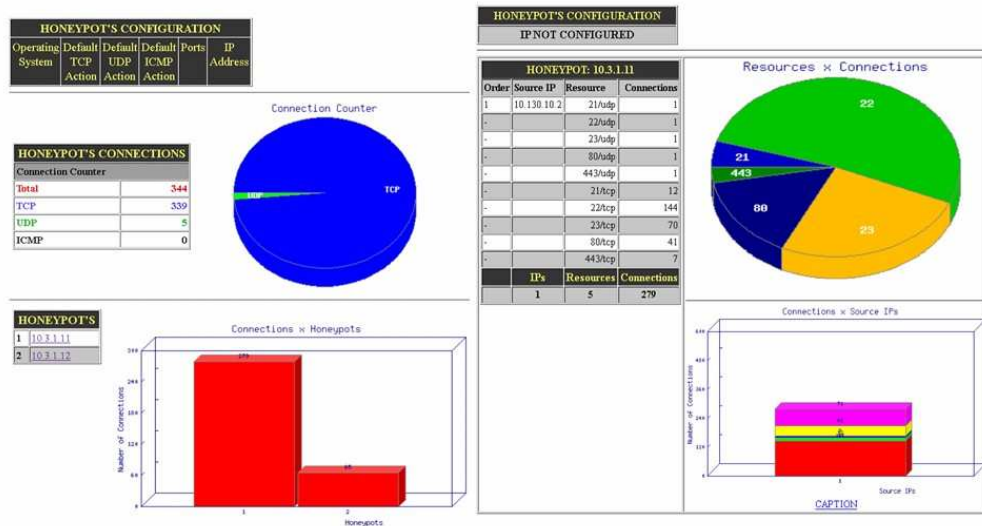
K prověření správné funkce nástrah je nejvhodnější použít stejné skenovací nástroje, které používají internetoví útočníci. Spustíme automatické skenování virtuálních strojů nástrojem např. Nmap nebo Xprobe.

Nástroj Nmap umožňuje zjišťování nabízených služeb, portů, topologii sítí a hlavně typ operačních systémů (fingerprinting). Velmi výkonný nástroj pro odhalování nedostatků sítě je Zenmap, což je graficky a účelově upravený Nmap. Je dobře ovladatelný a příkazová řádka je nahrazen předvolenými metodami kontroly.

Příklad skenovacích testů:

- Intenzivní skenování: nmap -T4 -A -v 10.30.1.12
- Intenzivní skenování s protokolem UDP: nmap -sS -sU -T4 -A -v 10.30.1.12
- Intenzivní skenování všech TCP portů: nmap -p 1-65535 -T4 -A -v 10.30.1.12

- Ping skenování: nmap -sn 10.30.1.12
- Trasování topologie: nmap -sn --traceroute 10.30.1.12



Obr. 14: Ukázka grafického výstupu aplikace Honeydsum do html kódu [21]

8.2.4 Automatické reporty

Databáze záznamů je uchována v logu „/var/log/honeypot/honeyd.log“. Aktuální přehled získáme až po spuštění skriptu „./honeydsum.pl -c honeydsum.conf -w /var/log/honeypot/honeyd.log“ ze správného adresáře. Rutinní akci naprogramujeme do démona cron²⁹ a zajistíme pravidelnou aktualizaci stavu honeypotu.

²⁸ Cisco IOS (Internetwork Operating System) - operační systém používaný v aktivních prvcích Cisco Systems. Zdroj: [http://en.wikipedia.org/wiki/Cisco_IOS].

²⁹ Cron - softwarový démon, který v operačních systémech automatizovaně spouští v určitý čas nějaký příkaz resp. proces (skript, program apod.). Jedná se vlastně o specializovaný systémový proces, který v operačním systému slouží jakožto plánovač úloh, jenž umožňuje opakované spuštění periodicky se opakujících procesů (např. noční běhy dávkových úloh při hromadném zpracování dat) apod. Zdroj: [http://cs.wikipedia.org/wiki/Cron].

9 DISKUZE

Jedním z cílů tohoto pojednání bylo upozornit čtenáře na užitečnost nasazení detektorů, které budou působit jako prevence proti neoprávněnému průniku do lokálních sítí. Monitoring a analýza datových toků patří často k podceňovaným činnostem administrátorů. Mezi důvody oné nepopularity patří vysoké pořizovací náklady profesionálních systémů, značná časová náročnost implementace nebo neznalost problematiky.

Pro zachování bezpečnostních standardů jsou rozhodně vážné důvody proč se zajímat a „vidět hlouběji“, více zkoumat obsah příchozích dat a snažit se poznat protivníka. V pojednání jsem se pokusil seskupit dostupné informace o problematice detekce a prevence průniku a na základě této studie pak probíhala implementace projektu. Práce obsahuje konkrétní postupy a podrobné kroky realizace, které byly předneseny do praktické části projektu integrace systémů IDS/IPS a Honeypot do podnikové sítě Fatra, a.s.

9.1.1 Zhodnocení praktické části

V praktické části byly naplněny očekávané požadavky zpracování studie a integrace detekčních systémů do firemní sítě Fatra, a.s. IDS Snort a honeypot Honeyd správně reaguje na podezřelé aktivity a generuje upozornění do databáze, která je dále vizuálně zpracována. Výstrahy v podobě informací o aktivitách malware, jsou využívány k doplnění návazných opatření jako utěsnění pravidel na firewallu, aplikace systémových záplat nebo odstranění virové infekce.

Odladění systémů probíhalo na virtuální platformě VMware ESXi na zkušebních linuxových serverech s aplikacemi IDS Snort a honeypotů Honeyd. Funkční a otestované systémy byly poté zařazeny do kritických částí produkční sítě Fatra, a.s. (WAN, DMZ, LAN, VLAN).

Výsledkem projektu jsou čtyři NIDS systémy Snort ve verzích 2.8 a 2.9., a dva honeypot Honeyd. Servery pracují samostatně a jsou vzdáleně spravovány prostřednictvím terminálu na TCP portu 222. Dostupnost informací pro správce je zajištěna prostřednictvím html prohlížeče na jednotlivých IP adresách.

9.1.2 Řešené problémy a nedostatky

Musíme vzít v úvahu, že žádný systém není dokonalý a každé bezpečnostní opatření nebo ochranný prvek sebou přináší dočasné provozní omezení. Systémy IDS/IPS se zpočátku po implementaci potýkají s problémem mnoha falešných poplachů a je nutno dohlížet a kontrolovat korektnost jejich rozhodnutí, případně je „doučít“. Příčinou je využívání principů složitých intuitivních metod. IDS/IPS operují na nevyšších síťových vrstvách a jejich rozlišovací mechanismy spočívají v porovnávání databází a vyhodnocování proměnných struktur, což má občas za následek chybovost. Více pozornosti je nutno věnovat tzv. „odladění systému“, nastavení filtrů a restrikcí. Špatně odladěný systém generuje příliš mnoho údajů a přináší opačný efekt. Podstatné údaje jsou ignorovány v záplavě poplašných zpráv.

Postup implementace probíhal většinou podle bodů zadání bez větších problémů. Malý zádrhel nastal při zpracování většího množství dat v aplikaci Snort Report 2.9, kdy nepracovala dostatečně rychle databáze MySQL. Důvodem byl špatně nastavený filtr, který způsoboval generování velkého množství falešných poplachů. Při dosažení kritické kapacity se databáze stala prakticky nedostupnou. Řešení tohoto problému spočívalo v podrobnější analýze incidentů, vyladění pravidel do užších mantinelů. Zvýšení výkonu databáze lze dosáhnout volbou robustnějšího databázového systému, což ale není předmětem této práce.

Nepovedlo se mi dokončit systém automatického generování a export bezpečnostních pravidel pro firewall neboť tato problematika je značně obsáhlá a stojí za samostatnou studii.

ZÁVĚR

Již dlouhá desetiletí se snaží výrobci bezpečnostního softwaru soupeřit s kyberkriminalitou, ale přes veškeré úsilí se jim nedaří dosáhnout uspokojivých výsledků. Důvodem je dynamika tohoto procesu.

Cílem projektu byla včasná detekce, identifikace a zachycení neznámého rizika, snaha obohatit softwarové bezpečnostní složky a přispět k minimalizaci incidentů plynoucích z využívání internetových komunikací. Prezentovaný systém dokáže analyzovat datové pakety a rozpoznat povahu komunikace na vyšších úrovních než je tomu u běžných bezpečnostních prvků jako firewall nebo antivirus.

Výše zmiňované dvě bezpečnostní řešení disponují rozdílnými přístupy ke sběru informací o škodlivých kódech a zaručují tak širší spektrum zachycených údajů.

Cíl projektu byl naplněn implementací detekčních sond a pastí do produkční sítě Fatra, a.s.

ZÁVĚR V ANGLIČTINĚ

All producers of security software for many years trying to fight cyber crime, but they still fail to achieve satisfactory results. The reason is the dynamics of this process.

The Purpose of the project was quick detection, identification and interception of unknown risk, enriching the software security forces and helps minimize the risks of the use of Internet communications. Presented system can analyze the data packets and identify the nature of communication at higher levels than that of conventional safety features such as firewall or antivirus.

The above presented two security solutions are different methods of collecting information of malware. This guarantees a wider range of data captured.

Target of the project implementation was filled with traps and detection probes into the production network Fatra, a.s.

SEZNAM POUŽITÉ LITERATURY

- [1] BEALE, Jay, FOSTER, James C. Snort 2.0 Intrusion Detection. Brian Caswell; Catherine B. Nolan; Technical Advisor: Jeffrey Posluns. [s.l.] : [s.n.], c2003. 559 s., 1 CD. Značné množství spisovatelů. ISBN 1-931836-74-4.
- [2] CASWELL, Brian; BEALE, Jay; BAKER, Andrew. Snort® : IDS and IPS Toolkit. [s.l.] : Syngress, 2007. 750 s. ISBN 1-59749-099-7, 978-1-59749-099-3.
- [3] DUNNIGAN, James. Bojiště zítřka : tváří v tvář globální hrozbě kybernetického terorismu. [s.l.] : Baronet Publishers, 2004. 356 s. ISBN 8072146424, 9788072146420.
- [4] Harrisová S., harper A., Eagle Ch., Ness. Hacking - manuál hackera: podrobnější popis. Praha : Grada Publishing, 2008. 399 s. ISBN 9788024713465.
- [5] JIROVSKÝ, Václav. Kybernetická kriminalita. Praha : Grada, 2007. 288 s. ISBN 978-80-247-1561-2.
- [6] Know Your Enemy : Learning about Security Threats. 2nd Edition. [s.l.] : Addison-Wesley Professional, 2004. 800 s. ISBN 10:0-321-16646-9.
- [7] PROVOS, Niels, HOLZ, Thorsten. Virtual Honeypots : From Botnet Tracking to Intrusion Detection. [s.l.] : Addison Wesley Professional, 2007. 480 s. ISBN 978-0-321-33632-3.
- [8] SCHULTZ, EUGENE; MELLANDER, JIM; ENDOR, CARL. Hacking – detekce a prevence počítačového útoku. 1. vydání . Praha : Grada Publishing, 2005. 365 s. ISBN 8024710358.
- [9] SPITZNER , Lance . Honeypots : Tracking Hackers. USA : Addison-Wesley , 2002. 480 s. ISBN 0321108957.

SEZNAM INTERNETOVÝCH ZDROJŮ

- [10] HARPER, Patrick. [Http://oinkmaster.sourceforge.net](http://oinkmaster.sourceforge.net) [online]. 2008 [cit. 2011-04-29]. Installing and configuring OinkMaster. Dostupné z WWW: http://www.internetsecurityguru.com/documents/Installing_and_configuring_Oink_Master.pdf.
- [11] [Http://www.symmetrixtech.com](http://www.symmetrixtech.com) [online]. 12.10.2010 [cit. 2011-04-29]. Snort 2.8.6 and Snort Report 1.3.1 on Ubuntu 10.04 LTS Installation Guide. Dostupné z WWW: <http://www.symmetrixtech.com/articles/004-snortinstallguide286.pdf>.
- [12] KRČMÁŘ, Petr. www.root.cz [online]. 23.2.2011 [cit. 2011-04-29]. Internetové útoky se dnes dělají pro peníze. Dostupné z WWW: <http://www.root.cz/clanky/internetove-utoky-se-dnes-delaji-pro-penize/>.
- [13] LASEK, Petr. Detekce a ochrana před hackerským útoky. Svět sítí [online]. 2003, č.1, [cit. 2010-11-17]. Dostupný z WWW: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=258&clanekID=259>.
- [14] ORKÁČ , Radomír. IDS Snort. [s.l.], 2006. 22 s. Seminární práce. Dostupný z WWW: <http://www.cs.vsb.cz/grygarek/SPS/projekty0506/Snort.pdf>.
- [15] PROVOS, Niels. A Virtual Honeypot Framework. In [online]. University of Michigan : [s.n.], 2003 [cit. 2011-04-29]. Dostupné z www: <http://www.citi.umich.edu/techreports/reports/citi-tr-03-1.pdf>.
- [16] SnortTeam SourceFire. Snort.org [online]. 2010 [cit. 2011-04-29]. Snort® Users Manual. Dostupné z WWW: http://www.snort.org/assets/166/snort_manual.pdf.
- [17] WIKIPEDIA: Intrusion prevention system. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 16 červenec 2003, last modified on 4 říjen 2010 [cit. 2010-10-12]. Dostupné z WWW: http://en.wikipedia.org/wiki/Intrusion_prevention_system.
- [18] Denial of Service [online]. 2011 [cit. 2011-05-03]. WIKIPEDIA. Dostupné z WWW: <http://cs.wikipedia.org/wiki/DDOS>.
- [19] Man in the middle [online]. 2011 [cit. 2011-05-03]. WIKIPEDIA. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Man-in-the-middle>.

OSTATNÍ ZDROJE

- [20] Vlastní zpracování v prostředí Smart Draw ver.2008.02, Copyright © SmartDraw.com.
- [21] Vlastní zpracování snímku obrazovky v aplikaci Microsoft Paint, Copyright © 2009 Microsoft Corporation.
- [22] Vlastní zpracování v aplikaci Microsoft Excel 2003, Copyright © 1985-2003 Microsoft Corporation.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ACID	(Analysis Console for Intrusion Databases) Analytický nástroj pro Snort
API	(Application Programming Interface) Rozhraní pro programování aplikací
BASE	(Basic Anylysis and Security Engine) analytický nástroj pro Snort
CPAN	(Comprehensive Perl Archive Network) Softwarové zdroje jazyka Perl
DMZ	Demilitarizovaná zóna
DAQ	(Data Acquisition API) Proces odběru vzorků signálů
DHCP	(Dynamic Host Configuration Protocol) Automatické přidělování IP adres
DNS	(Domain Name Server) Server na překlad doménových jmen
DoS, DDos	(Denial of Service, Distributed Denial of Service) odmítnutí služby, distribuované odmítnutí služby
GPL	(General Public License) Všeobecná veřejná licence
GNU	(GNU's Not Unix!) GNU není Linux!
GNOME	(GNU Network Object Model Environment) prostředí pracovní plochy
HIDS	(Host Based Intrusin Detection System) Klientský detekční systém
HIPS	(Host Based Intrusin Prevention System) Klientský prevenční systém
HUB	Síťový rozbočovač
IDS	(Intrusion Detection System) Systém detekce průniku
IPS	(Intrusion Prevention System) Systém prevence průniku
LAN	(Local Area Network) Lokální síť
NIPS	(Network Intrusion Prevention System) Síťový systém prevence průniku
NIDS	(Network Intrusion Detection System) Síťový systém detekce průniku
NMAP	Analyzátor otisku operačních systémů
OS	Operační systém
OPSEC	(Open Platform for Security) Sdružení dodavatelů bezpečnostních řešení

SPAN	(Switched Port Analyzer) Zrcadlený port na aktivním prvku sítě
SSH	(Secure Shell) Šifrovaný komunikační protokol
VLAN	(Virtual Local Area Network) Virtuální lokální síť
VPN	(Virtual Private Network) Virtuální privátní síť
WAN	(Wide area network) Počítačová síť na rozlehlém geografickém území
Xprobe	Síťový skenovací software, analyzátor otisku operačních systémů

SEZNAM OBRÁZKŮ

Obr. 1: Příklad umístění více zařízení NIDS v lokální síti	17
Obr. 2: Příklad zapojení Network TAP před firewall	18
Obr. 3: Volba zapojení NIPS v lokální síti	19
Obr. 4: Architektura IDS Snort	32
Obr. 5: Base - nástroj pro analýzu databáze Snort verze 2.8	33
Obr. 6: SnortReport - nástroj pro analýzu databáze Snort verze 2.9	34
Obr. 7: Architektonický přehled systému Honeyd	36
Obr. 8: Rozmístění detekčních sond v DMZ a LAN	41
Obr. 9: Ukázka grafického reportu z aplikace BASE	45
Obr. 10: Detekční sonda umístěná před firewallem	46
Obr. 11: Ukázka grafického výstupu z aplikace SnortReport	51
Obr. 12: Struktura instalovaných nástrah honeypotů v podnikové síti Fatra, a.s.	55
Obr. 13: Ukázka textového výstupu z aplikace Honeydsum	63
Obr. 14: Ukázka grafického výstupu aplikace Honeydsum do html kódu	65

SEZNAM TABULEK

Tab. 1: Tabulka chování jednoduchých portů v konfiguraci Honeyd.....	37
Tab. 2: Volby nastavení akce v definicích pravidel.....	52
Tab. 3: Jmenný seznam atributů z definic a pravidel pro nastavení rolí.....	53

SEZNAM PŘÍLOH

PŘÍLOHA P I: Script „barnyard2.conf“