JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH
Pedagogická fakulta
Katedra informatiky

IDS systém SNORT

Bakalářská práce

Vedoucí práce:
Ing. Ladislav Beránek, CSc., MBA

Autor práce:
Jakub Mauric

České Budějovice 2009

Prohlášení

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

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách.

V Českých Budějovicích, 12. dubna 2009 Jakub Mauric

Abstrakt

Tato práce se zabývá systémy detekce průniků. Rozděluje tyto systémy do kategorií a popisuje jejich funkčnost. Popisuje příklady jejich praktického nasazení. Zabývá se především IDS systémem Snort, jeho obsahem i ukázkou implementace do existujícího systému.

Tento dokument bude sloužit jako stručný ucelený návod, který popíše potřebnou teorii a možnosti praktického nasazení.

Abstract

This work deals with Intrusion Detection Systems. It divides them into categories and describes their functions. It describes examples of their using. It deals primary with IDS System Snort, content of them and with an example of their implementation into an existing system.

This document will be used like a short compact manual which describes a necessary theory of Intrusion Detection Systems and possibilities of their practical using.

Poděkování

Děkuji všem, kteří mne podporovali, zvláště pak panu Ing. Ladislavu Beránkovi, CSc., MBA za odborné vedení práce a dále místopředsedovi VODVAS.Net o.s. panu Jaroslavu Vazačovi za poskytnutí realizačního zázemí.

Obsah

Seznam obrázků

Seznam tabulek

Slovníček zkratek

CER – Crossover Error Rate

DIDS – Distributed IDS

FNR – False Negative Rate

FPR – False Positive Rate

HIDS – Host-Based IDS

HIMS – Host Intergrity Monitoring System

IDS – Intrusion Detection System

IS – Information System

NIDS – Network-Based IDS

SSH – Secure Shell

SSL – Secure Sockets Layer

Kapitola 1

Úvod

Cílem této práce je doplnění zabezpečení počítačové sítě občanského sdružení VODVAS.Net o IDS systém Snort, přiblížení problematiky IDS (Intrusion Detection System) systémů a následná demonstrace vlastní realizace tohoto projektu. Tato bakalářská práce by měla sloužit jako jakýsi stručný ucelený návod v českém jazyce pro případné další realizace bezpečnostních opatření pomocí tohoto nástroje.

Právě probíhající informační revoluce není dnes téměř pro nikoho žádným tajemstvím. Značná rychlost vývoje informačních technologií je pro mnohé až udivující. Nicméně v pozadí veškerých výkonnostních pokroků jako je např. vyšší výpočetní výkon procesorů, zvyšování kapacit záznamových médií a nárůst kvality záznamových zařízení probíhá ještě jeden méně znatelný, ale o to zásadnější proces. Tímto procesem je všeobecné spojování a komunikace nejrůznějších zařízení mezi sebou. Nejvyšším vrcholem se stala světová počítačová síť Internet, jenž každému jednotlivci nabízí nepřeberné množství vzdělání, zábavy, práce a dalších služeb, které mnohdy nabývají i finančního charakteru. Nová technologie neodvratně přinesla i nové hrozby v podobě všemožného spyware, virů, spamů atd. S přirůstajícím množstvím komunikace se náležitě zvyšuje i ohrožení jednotlivých účastníků této komunikace.

Možností ochran existuje hned několik. Jednak se jedná o ta nejjednodušší, mezi něž se řadí různá hesla a kódy. Dále o zabezpečení koncových zařízení ochrannými aplikacemi tzv. Antiviry, AntiSpyware. Dalším znamenitým prvkem je Firewall, který slouží k omezení komunikace ať už na úrovni vlastní sítě či koncové stanice. Tímto způsobem bývá zabezpečena valná část informačních systémů. Nabízí se však možnost rozšíření o další vrstvu ochrany, touto vrstvou se rozumí Systém detekce průniků (Intrusion Detection System).

Kapitola 2

Problematika IDS

2.1 Intrusion Detection

Slovem průnik (angl. Intrusion) běžně rozumíme akt vniknutí na nějaké území nebo místo, a to bez pozvání, dostatečného oprávnění či vítání. Pokud je tento pojem spojován se světem informačních technologií, má slovo průnik význam neautorizovaného vstupu do informačního systému. Tyto nelegální aktivity je třeba včas odhalit a za pomocí různých nástrojů eliminovat do takové míry, do které je to jen možné. Takto definovaná činnost se nazývá detekce průniků.

Systém detekce průniků (IDS) lze přirovnat např. k zabezpečení budovy alarmem. Stejně tak jako alarm hlídá pohyb po budově, narušení oken a vstupů, tak i IDS systém monitoruje komunikaci, komunikační porty, odchylky od nadefinovaného normálu chování, porovnává právě probíhající situaci s uloženými modely chování ve své databázi – tzv. signaturami. Na základě detekce průniku je dále možné aktivovat alarmy, automaticky kontaktovat administrátory, vyvolat bezpečnostní odpověď v podobě blokace komunikace. Je tedy zřejmé, že některé IDS systémy mohou mít přímou vazbu na firewally či antiviry a dokáží tedy odpovídajícím způsoben čelit hrozbě v reálném čase za pomoci konfigurace těchto prvků. IDS je nejlepší si představit jako vysoce specifikovaný hi-tech nástroj pro zprávu zabezpečení informačního systému, jež dokáže číst a interpretovat obsah logovacích souborů routerů, firewallů, serverů a dalších síťových zařízení. Systém detekce průniků často shromažďuje jím zaznamenaná data v databázi, do nichž je možné kdykoli nahlédnout a vytvořit z nich odpovídající statistiky, analyzovat charaktery útoků, popř. vyvodit dodatečná bezpečnostní opatření.

Pro srovnání IDS poskytuje počítačové síti to samé, co poskytuje antivirový program pro systém. Což znamená, že prohlíží obsah síťové komunikace a hledá náznaky možného útoku, stejně tak jako antivirový program prohlíží obsah příchozích souborů, v nichž pátrá po virové nákaze.

Pro ještě lepší specifikaci systém detekce průniků slouží k detekci neoprávněných průniků do systémů počítačových sítí a dalších podobně založených zdrojů potencionálně citlivých dat. Stejně jako firewall i IDS může být plně softwarový nebo kombinovat hardware i software. IDS bývá velmi často instalován přímo na síťovém zařízení (server, firewall) a tvoří tak samostatnou funkčně-zabezpečující jednotku. Tímto konceptem je zajištěna kvalita zabezpečení jak vlastního zařízení, na němž IDS sonda pracuje, tak i monitorování provozu. Z tohoto důvodu je systém schopen vypořádat se s externím i interním nebezpečím.

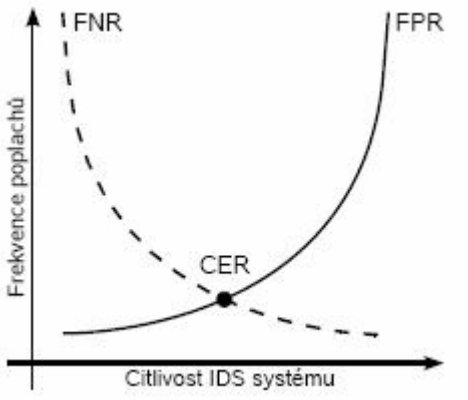
Rozeznáváme tři základní způsoby implementace systému detekce průniků. Prvním z nich je network-based IDS, ten sleduje síťové prostředí a hledá v něm objevující se signatury. Druhým je host-based IDS, jehož úkolem je obrana systému v síti, jímž je typicky stanice (počítač zapojený v síti) nebo server. Obrana je zajišťována kontrolou operací a souborového systému, kde se opět pátrá po objevujících se signaturách. Konečně posledním způsobem implementace je Distributed IDS, který je založen na systému rozmístění senzorů, jež posílají zprávy řídícímu systému. Prakticky se používají především kombinace těchto zmíněných typů. Je na první pohled zřejmé, že nejlepších výsledků co se týče odhalování nebezpečí dosahují IDS monitorující jak síťové prostředí, tak i systémy v této síti zapojené.

Aby bylo zřejmé, jakým způsobem systém detekce průniků rozeznává hrozby, je nezbytné uvést podrobnější vysvětlení. Především existují dva způsoby detekce. Prvním je technika nazvaná detekce signatur. Druhou je pak detekce anomálií. Detekce signatur pracuje obdobným způsobem jako antivirové programy používající virové signatury, které se následně snaží odhalit mezi soubory, programy či webovým obsahem vstupujícím do systému. Je tedy patrné, že detekce signatur prováděná systémem detekce průniků probíhá sledováním síťové komunikace, její analýzou a následném porovnání s databází známých ukazatelů, jež značí případné bezpečnostní riziko. Tato technika je velmi často upřednostňována a bývá používána jako primární u většiny komerčních IDS. Naopak detekce anomálií je založena na identifikování abnormální aktivity systému. Používá pravidla, jež definují koncept normální a abnormální aktivity systému (tzv. heuristiky), aby odhalil odklon od běžného chování. Následně přijme potřebné opatření, jímž je např. blokace komunikace či informování správce systému. Tento způsob je ovšem náročnější na čas, obslužný personál i finance. Vyžaduje značné úsilí při ladění tzv. na míru konkrétnímu informačnímu systému. Jelikož se jedná o jedno ze stěžejních témat, bude tomuto tématu věnovaná samostatná kapitola (2.4 Metody detekce průniků). K problematice ladění IDS systému neodmyslitelně patří i pojem efektivnost IDS systému, jež má zásadní vliv na konečnou bezpečnost a použitelnost IDS. Je právě jednou z největších překážek při implementaci systému detekce průniků do informačních systémů, a to z výše uvedených požadavků, které platí nejen pro variantu s detekcí anomálií, ale v menší míře i u detekce pomocí signatur.

2.2 Efektivnost IDS

Tuto kapitolu dobře popisuje David Šumský:

„Praktická aplikovatelnost IDS je závislá na jejich efektivnosti (přesnosti) a efektivitě (výkonnosti). Efektivností IDS rozumíme poměr mezi počtem nezachycených průniků a počtem falešných poplachů. Označíme-li tyto veličiny postupně FNR a FPR, pak jejich vzájemnou závislost vyjádříme graficky takto 2.3: Z grafu vyplývá, že s rostoucí veličinou FPR, kdy roste citli-



Obrázek 2.3: Závislost mezi FNR a FPR

vost IDS, veličina FNR na úkor rostoucího počtu falešných poplachů klesá. Je zřejmé, že existuje střední hodnota CER (Crossover Error Rate), která je průnikem veličin FPR a FNR a definuje optimální nastavení IDS. Formálněji můžeme prezentovanou závislost, kde veličina C udává nastavenou citlivost IDS a výraz X" resp. X# rostoucí resp. klesající hodnotu veličiny X, vyjádřit takto:

$$C\uparrow \Rightarrow FNR\downarrow \wedge FPR\uparrow$$
$$C = CER \Rightarrow FNR = FPR$$
$$C\downarrow \Rightarrow FNR\uparrow \wedge FPR\downarrow$$

Uvědomíme-li si, že poměr mezi počtem anomálních a normálních událostí IS je relativně nízký, poněvadž vznik normálních událostí je v reálném prostředí častější, a vznik normální resp. anomální události předchází vzniku poplachu, pak na základe zákonu podmíněné pravděpodobnosti dospějeme k překvapivému závěru:

- *Je-li IDS dostatečně spolehlivý v identifikaci anomálních událostí, má nízké FNR a FPR, pak výsledná pravděpodobnost, že generovaný záznam o anomálii skutečně identifikuje potenciální zneužití IS, je paradoxně nižší než pravděpodobnost odhalení tohoto zneužití.*

Události IS kategorizujeme jako anomální a normální. Anomální událost označme jako jev M (Misuse) a normální jako jev ¬M, jedná o negaci anomálního jevu M. IDS zkoumá události IS a na základe toho generuje záznamy o anomáliích příp. auditní záznamy normálních událostí. Tyto jevy

postupně označme jako jev A (Alarm) a jev ¬A. Efektivnost IDS pak vychází z pravděpodobnosti dvou jevů a to:

1. *P(A/M) – pravděpodobnost, že bude generován záznam o anomálii A, jestliže došlo ke vzniku anomální události M.*
2. *P(A/¬M) – pravděpodobnost, že bude generován záznam o anomálii A, jestliže došlo ke vzniku normální události ¬M.*

Pro úplnost uveďme, že jevu A/¬M odpovídá veličina FPR a komplementární veličině FNR jev ¬A/M.“

([3], str. 17, 18)

2.3 Úroveň implementace IDS

V této kapitole budou hlouběji rozebrány tři úrovně implementace IDS, o nichž bylo již zmíněno několik obecných informací.

2.3.1 Network-Based IDS (NIDS)

Jak již bylo řečeno, NIDS se zaobírá monitorováním sítě, resp. segmentu sítě, kde je implementován. Sleduje provoz ve všesměrovém komunikačním médiu pomocí promiskuitního módu síťové karty hostitelského systému. NIDS bývá instalován na strategických místech v počítačové síti tak, aby mohl co nejlépe bránit celý systém.

Jeho zásadním problémem jsou přepínané sítě a šifrovaná data. V případě přepínaných sítí musí být sondy umístěny na takových místech, kde mohou monitorovat celý segment (typicky router nebo bridge do této subsítě) a nebo důležitý prvek, jakým je např. server. Co se týče šifrovaných dat NIDS pracuje na úrovni třetí vrstvy ISO/OSI modelu (síťová vrstva), kde nelze analyzovat data kódovaná protokoly SSH, SSL/TLS, PPTP apod. Problém se částečně řeší přidáním podpory těchto protokolů přímo do NIDS.

Příklad zapojení v topologii sítě je možné shlédnout na obr 2.3.1.1. Příklad síťové topologie s NIDS.

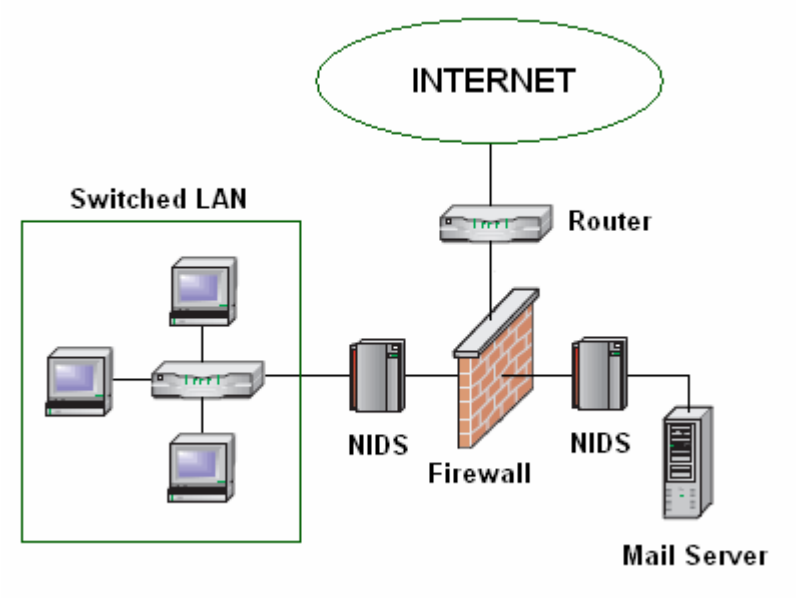
Výhody tedy jsou:

- Schopnost monitorovat provoz na síti, resp. subsíti.
- Dokáže spolehlivě odhalit útok na síťovou vrstvu.
- Analýza síťového provozu dokáže spolehlivě sledovat parametry TCP/IP hlaviček.
- Nezávislost na hostitelském systému. Sledovaný síťový provoz je vůči různým implementacím TCP/IP transparentní.

Nevýhody tedy jsou:

- Náchylnost na přetížení a zahlcení hostitelského systému.
- Nemá přístup na Aplikační vrstvu ISO/OSI modelu, z čehož pramení následující nedostatky. Většinou nerozlišuje, na které

operační systémy nebo aplikace byl veden útok. Mnohdy nedokáže rozlišit povahu útoku. Nedokáže určit subjekty, které jsou iniciátory spojení.



Obr. 2.3.1.1. Příklad síťové topologie s NIDS

2.3.2 Host-Based IDS (HIDS)

Host-Base IDS je implementován na úrovni hostitelského systému. Je tedy jeho součástí, z čehož plyne jeho síla. Rozumí tomu, jak hostitelský systém funguje a reaguje. Zároveň dokáže identifikovat útoky, které NIDS identifikovat neumí. Je schopen sledovat i šifrovanou komunikaci, jelikož pracuje na úrovni aplikační vrstvy ISO/OSI modelu.

Problematickou je jeho údržba. Jelikož je Host-Base IDS umístěn na separátních systémech, je nutné jeho aktualizace a případná další nastavení aplikovat postupně všude, kde je instalován. Skutečným problémem je však selhání při útoku na síťovou vrstvu. Pokud totiž dojde k napadení např. DoS útokem a následnému odstavení systému, je odstaven i Host-Base IDS. Z toho plyne, že při útoku na nižší vrstvy ISO/OSI modelu tato koncepce selhává. Dalším neduhem se stává integrita vlastních dat poskytovaná hostitelským systémem. Pokud skutečně dojde k situaci, kdy se útok podaří, začne IDS zpracovávat nedůvěryhodná data.

Příklad zapojení v topologii sítě je možné shlédnout na obr 2.3.2.1 Příklad síťové topologie s HIDS.

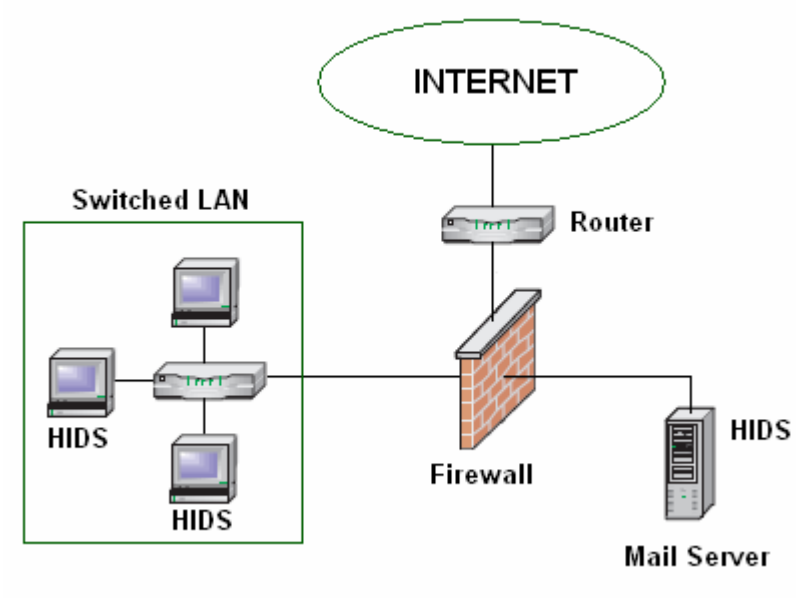
Výhody tedy jsou:

- Optimalizace pro hostitelský systém.
- Schopnost analyzovat podstatu útoku.
- Schopnost analyzovat aktivity uživatele.

- Analyzuje pouze hostitelský systém, není tedy zahlcován.

Nevýhody tedy jsou:

- Neschopnost čelit útokům na nižší vrstvy ISO/OSI modelu.
- Zpracovává data poskytovaná hostitelským systémem, je tedy závislý na jejich integritě.
- Nepřenositelnost na jiné systémy.



Obr. 2.3.2.1 Příklad síťové topologie s HIDS

2.3.3 Distributed IDS (DIDS)

Tento koncept zahrnuje oba výše popsané způsoby implementace IDS. Distributed IDS senzory mohou být síťového i systémového charakteru. Systém senzorů rozmístěných po topologii počítačové sítě zasílá výsledky svého měření centrální řídící stanici (NIDS Management Station). Toto řešení dosahuje nejlepších výsledků, zahrnuje totiž výhody obou výše zmiňovaných implementací.

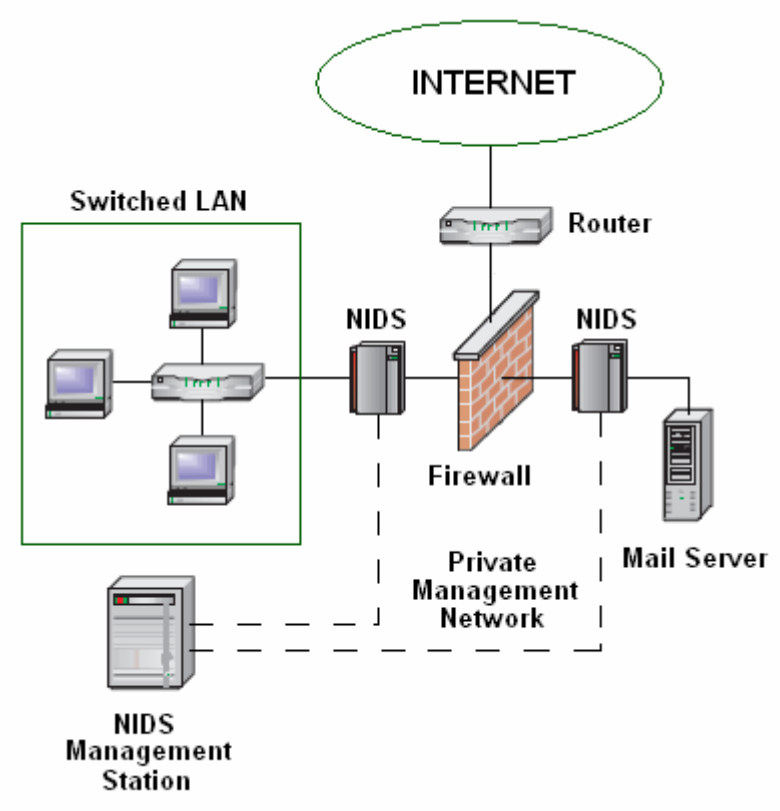
Příklad zapojení v topologii sítě je možné shlédnout na obr 2.3.3.1 Příklad síťové topologie s DIDS.

Výhody tedy jsou:

- Odstranění většiny neduhů samostatně stojících HIDS a NIDS.
- Monitorování jednotlivých systémů i celé sítě, resp. subsítě.
- Souhrnný přehled o případných hrozbách z řídící stanice.

Nevýhody tedy jsou:

- Interakce s procesy běžícími na síťové vrstvě (Firewall, VPN server apod.), které mohou vést k jejich nekorektnímu chování.



Obr. 2.3.3.1 Příklad síťové topologie s DIDS

2.4 Metody detekce průniku

Tato kapitola bude dále rozvíjet a detailněji zkoumat popsané způsoby detekce průniků. Jedná se o detekci založenou na bázi signatur (knowledge-based resp. misuse Detection system) a normálního/anomálního chování systému (behaviour-based resp. anomaly detection systems).

Detekce průniku do informačního systému (IS) je založena na myšlence, že je možné probíhající útoky, resp. jejich formát definovat a následně i strojově číst. Stejně tak je možné rozeznat chování systému.

Systémy detekce průniku by také měly být adaptabilní na změnu v okolním prostředí, též by měly být škálovatelné a rozšiřitelné pro různé domény použití.

2.4.1 Systémy založené na znalosti formátu průniků

Tuto problematiku opět skvěle vysvětluje David Šumský:

„IDS založené na znalosti formátu průniku označujeme jako signaturové systémy a specifikaci formátu průniku výrazem signatura. Přesněji se jedná o reprezentaci určitého bitového vzorku ve vhodném tvaru, jenž se může vyskytnout ve zkoumaných datech, která jsou ekvivalentní událostem IS.

IDS pak tyto data srovnává se signaturami uloženými v bázi signatur, což je množina tech signatur, které IDS dokáže rozpoznat. Dojde-li ke shodě, IDS vygeneruje záznam o anomálii o průniku do IS.

Binární vzorek signatury odpovídá formátu průniku do IS. Formát vzorku je implementačně závislý na použitém IDS, typicky se jedná o speciální jazyky umožňující formálně zapsat formát průniku. Např. vyjádření signatury LAND útoku v přirozeném jazyce vypadá takto:

> *Signatura LAND útoku je ekvivalentní TCP/IP datagramům, které mají nastaven příznak SYN a zároveň platí, že zdrojová a cílová IP adresa resp. zdrojový a cílový port se shodují.*

Signaturové systémy disponují omezenějšími prostředky detekce průniku z hlediska spolehlivosti v odhalování polymorfních útoků. Dojde-li totiž ke změně formátu průniku, je typicky nevyhnutelné nadefinovat signaturu novou, a to i v případě, že se jedná o podobný průnik. Z toho vyplývá zásadní vlastnost signaturových systémů, a to udržovat jejich bázi signatur aktualizovanou (v případně tzv. zero-day útoku to ale nepomáhá, poněvadž na útok signatura ještě neexistuje).

Předností signatur je snadnost, s jakou je můžeme vytvářet díky vyjadřovacím schopnostem jazyku k tomu určených. Výhodou je i to, že signaturami dokážeme zachytit pouze známé útoky. Vrátíme-li se ale k veličinám FPR a FNR, pak dojdeme k opačnému závěru. FPR dosahuje minimálních hodnot a hodnotu FNR není možné spolehlivé definovat, poněvadž nelze jednoduše určit, kolik průniků IDS nedokáže zachytit. Signaturovým systémům je totiž vlastní neúplnost báze signatur, kterou nikdy nepokryjeme všechny možné průniky. Nový průnik implikuje potřebu definovat novou signaturu a bázi signatur aktualizovat. Průnik se může zároveň v různých prostředích projevovat odlišně a nadefinovat univerzální signaturu tak, aby dokázala pokrýt tyto odchylky, je nemožné. Signaturové systémy jsou tak závislé na prostředí, ve kterém jsou aplikovány (např. odlišnost H-IDS pro Unix nebo MS Windows).

Mezi signaturové systémy řadíme i prototypy IDS implementující praxí neověřené postupy detekce průniku. Využívá se k tomu expertních systémů definovaných množinou pravidel ekvivalentních s množinou podezřelých událostí IS. Jím odpovídající auditní záznamy, v terminologii expertních systémů dostupná fakta, jsou přeložena do jazyka expertního systému. Tento postup je využíván z toho důvodu, že není nutné precizně oddělovat sémantiku záznamu od pravidel, která se tak stává plně jejich součástí. Expertní systémy jsou pro jednoúčelové použití snadno programovatelné, poněvadž se typicky jedná o vyhodnocení výrazu tvaru if-then-else, a využívají se zejména pro potřeby otestování prototypu.

Stavové systémy lze rovněž využít k detekci průniku do IS. Průnik do IS si můžeme představit jako sled akcí, kterým odpovídá přesně definovaný přechod z výchozího stavu do navazujícího stavu IS. Stav definujeme jako množinu proměnných dostatečně hodnotících prostředí IS z hlediska jeho bezpečnosti. Každá podezřelá událost je pak jednoznačně zachycena scénářem ve tvaru stavového diagramu, který znázorňuje stavy IS a přechody mezi nimi, resp. sled akcí nutně proveditelných pro dosažení koncového stavu „došlo k průniku do IS“.“

([3], str. 34, 35)

2.4.2 Systémy založené na znalosti chování systému

Druhý, principiálně odlišný způsob, rozpoznává na základě nadefinovaných pravidel abnormální chování IS. Zde bych taktéž využil skvělé práce pana Davida Šumského, který navazuje na předchozí citované téma:

„Průnik do IS je možné rozpoznat v jeho odchylkách chování. Vytvoříme-li model normálního chování IS, pak za podezřelou událost považujeme každou takovou událost, která do definovaného modelu nezapadá. Detekcí odchylek v chování IS vzhledem k normálnímu resp. očekávanému chování IS pak můžeme podezřelé události identifikovat a označit záznamem o anomálii.

V úvodní fázi je nutné zachytit normální stav IS, který bude předlohou definice modelu normálního chování IS. Tento model označujeme jako referenční model normálního chování systému. Referenční model je vytvářen v omezeném časovém intervalu a zachytíme jím všechny behaviorální aspekty IS, které se během tohoto intervalu v IS vyskytnou. Referenční model chování dosahuje narozdíl od signatur podstatně vyšší úplnosti, na druhou stranu není tak přesný a je u něj častější výskyt falešných poplachů – veličina FPR nabývá vyšších hodnot. Zároveň ale platí, že FNR je definovatelné. Tyto IDS systémy dokážou detekovat i nové útoky a nejsou tak svázány pouze s prostředím, ve kterém se podezřelá událost vyskytla.

Zásadní problém techniky vyplývá z překlenovacího období, kdy je nutné zachytit normální stav chování IS. Během tzv. fáze učení je IDS nepoužitelný a generuje více falešných poplachů. Během této fáze zároveň existuje riziko, že výskyt podezřelých událostí v IS zůstane neodhalen a bude přidán do referenčního modelů Potenciální průnik do IS tak může být později vyhodnocen jako běžná událost IS.

Statistické metody detekce průniku se zdají být pro účely zachycení normálního chování IS optimální. Chování entit IS je definováno množinou charakteristických proměnných, jejichž hodnoty jsou měřeny a zpracovávány v průběhu vytváření referenčního modelu normálního chování IS (např. průměrný počet neúspěšných přihlášení uživatele do IS apod.). Referenční model je dále definován těmito proměnnými. Detekce průniku pak probíhá tak, že aktuální hodnoty proměnných jsou porovnávány s hodnotami referenčního modelu (často vycházíme z několika modelů zároveň) a překročení stanovených mezních hodnot indikuje výskyt podezřelé události, který vede ke vzniku záznamu o anomálii.

Jednodušší metody založené na sledování překročení mezních hodnot charakteristických proměnných označujeme jako kvantitativní analýza. IS může mít nadefinován např. maximální počet neúspěšných přihlášení do systému na uživatele. Pokročilejšími metodami jsou statistická měření. Jedná se o IDS udržující specifickou bázi profilu IS, které statisticky definují normální chování IS a které jsou pravidelně aktualizovány, přitom platí, že profil je významově ekvivalentní referenčnímu modelu. Odchylka aktuálního chování IS od příslušného profilu signalizuje opět podezřelou událost IS. IDS založené na znalosti chování IS budeme dále označovat jako statistické systémy.“

([3], str. 35, 36)

2.4.3 Srovnání principů detekce průniku

Srovnání těchto dvou metod detekce průniků nemá jednoznačného vítěze, záleží na plánovaném prostředí a doméně použití. Dalšími požadavky jsou hodnoty FNR a FPR, schopnost procentuálního odhalení průniků do IS. Pokud se správce IS rozhoduje jaký princip detekce průniků použije, je vhodné položit si otázky, které se budou týkat jednotlivých specifických vlastností obou metod. Příkladem mohou být následující tři otázky:

- Kolik času a zdrojů jsem ochoten investovat do správného nastavení a odladění IDS?
- Jsou mé odborné schopnosti na takové úrovni, abych byl schopen bezpečně rozhodnout o normálním/abnormálním chováním IS?
- Jsem ochoten akceptovat prodlevu mezi novým způsobem průniku do IS a vytvořením odpovídající signatury?

2.5 Důležitost a použití IDS

Jestliže vám stále není jasné, proč je IDS tak důležitý a k čemu se přesně hodí, věžte, že rozuzlení vám poskytne sledující podkapitola.

V reálném životě člověk ví, co mu může uškodit, ale zároveň stále existují, popřípadě vznikají či jsou objevována další nebezpečí, o nichž se postupem času dozvídá. Stejně tak i v prostředí IS a počítačové komunikace je mnoho známých nebezpečí, nicméně je především ještě větší množství nebezpečí, která teprve vzniknou. Filosofií počítačových pirátů a hackerů není postupovat ve vyznačených kolejích a vymýšlet nové útoky na již známých pravidlech. Tito lidé se vyznačují především enormní kreativitou mezi mnohými nazývanou uměním. Smyslem IDS je především reagovat na nové a neznámé hrozby, které jiné obranné mechanismy nejsou schopny zachytit. Slouží jako velice kvalitní doplněk ochrany informačního systému, jenž nemůže nahradit firewall, antivir ani cokoli jiného. To co může udělat je, že upozorní správce IS na to, co je podezřelé a co firewall, antivir ani jiný prvek nedokázal zachytit a poskytnout správci IS čas na řešení nově vzniklého problému ještě před tím, než dojde k průlomu a škodám. Tím zároveň ukazuje na slabiny ostatních komponent zabezpečení. V případě absence takového systému je IS oslaben a dochází k vystavení se riziku vzniku situace, kdy se o proběhlém průniku správce IS dozví až tehdy, když systém zkolabuje a databáze jsou infiltrovány. Touto situací rozumíme selhání zabezpečení IS a snažíme se jí předejít.

O některých možných využitích již byla řeč v předešlém textu. Obecně lze však říci, že je vhodné jej využít všude tam, kde se nacházejí cenné zdroje, na strategických pozicích v sítích i tam, kde si jen přejeme ozkoušet spolehlivost ostatních prvků zabezpečení IS. O strategickém umístění v sítích již bylo napsáno dost. Nyní je vhodné pohovořit i o dalších konkrétních příkladech, které ne jen podle ([1], str. 20-23) existují.

2.5.1. Monitorování přístupu k databázím

Databáze jsou zřejmě nejvíce kritickým místem informačních systémů vůbec. Často obsahují to nejcennější co organizace, společnosti či vlády vlastní. Průnik do takovéto databáze by znamenal naprostou katastrofu, a proto bývají nejpřísněji chráněny. U těch nejdůležitějších dat ani neexistuje síťové spojení pomocí počítačové sítě a je k nim umožněn přístup pouze z lokálních hlídaných terminálů. V případě, že k databázím existuje síťové spojení (naprostá většina případů), má monitorování přístupů k nim nejvyšší prioritu. Je nezbytně nutné naprosto přesně vědět, kdo kdy a k čemu přistupoval a povolit přístup jen oprávněným uživatelům databáze.

Například IDS systém SNORT obsahuje komplexní sadu pravidel, jež umožňuje chránit databázi. Několik příkladů pro ilustraci:

- ORACLE drop table attempt
- ORACLE EXECUTE_SYSTEM attempt
- MYSQL root login attempt
- MYSQL show databases attempt

2.5.2. Monitorování funkce DNS

Důležité je uvědomit si, že DNS server sítě obsahuje řadu citlivých informací o dané síti jako jsou jména komponent, IP adresy apod. Vhodný zásah do serveru DNS umožní útočníkovi zmapovat tuto síť nebo přesměrovat uživatele hledající na internetu. Typickým příkladem je technika zvaná pharming. Cílem této techniky je podsunout surfujícímu uživateli falešné stránky, které nejsou na první pohled rozpoznatelné od originálu. Tato technika se zaměřuje na podvodné získávání citlivých údajů jako jsou bankovní přístupy apod. Snort opět obsahuje řadu pravidel odhalujících netypické dotazy či zásahy, aby ochránil váš jmenný prostor. Několik příkladů pro ilustraci:

- DNS Name Version Attempt
- DNS Zone Transfer Attempt

2.5.3. Ochrana e-mailového serveru

Dalším typickým terčem útoků je poštovní server. Tyto servery jsou často chráněny různými antivirovými programy. Stejně tak i IDS jako je Snort mají pravidla pro detekci e-mailových virů jako je QAZ worm nebo NAVIDAD worm. Výhodou IDS v tomto případě je opět rychlost reakce na nové hrozby. V případě útoků, které využívají prodlevy mezi jejich vypuštěním po jejich zahrnutí v bezpečnostních řešeních jednotlivých společností, je IDS opět schopen pomocí některých pravidel odhalit nebezpečí na základě jejich podezřelého obsahu či chování. Snort může být nastaven na blokaci útočných e-mailů stejně tak dobře jako jiných hrozeb, které mohou vypnout poštovní služby. Navíc nic nebrání tomu využívat oba systémy paralelně, což je samozřejmě nejlepší řešení.

Kapitola 3

Existující IDS systémy

Cílem této kapitoly je přiblížit některé z nejběžnějších existujících systémů detekce průniků. Je důležité uvědomit si, že systém Snort není zdaleka jediným řešením v této oblasti. Těchto systémů existuje celá řada a to jak volně šiřitelných, tak i komerčních. Systémem Snort je určena čtvrtá kapitola tohoto dokumentu.

3.1 BRO Intrusion Detection System

BRO je open-source NIDS založený na Unixu, který pasivně monitoruje provoz a hledá podezřelé aktivity. BRO pracuje tím způsobem, že nejdříve rozebere komunikační provoz, aby extrahoval sémantiku aplikační vrstvy a v ní dále hledá objevující se náznaky podezřelých aktivit. Tyto analýzy jsou schopny zachytit jak již známé hrozby na základě signatur, známých okolností nebo událostí, tak i očekávané hrozby na základě odepřeného přístupu či spolehlivosti spojení určité služby. Jeho bezpečnostní skripty jsou psány vlastním Bro jazykem a jsou schopny spouštět i některé akce. V případě, že si uživatel osvojí jazyk Bro, má možnost psát skripty vlastní, popř. upravovat skripty stávající. Bro obsahuje velké množství již hotových skriptů, které jsou připravené k použití a není k nim potřeba znalost tohoto jazyka. Tyto skripty jsou schopny zachytit téměř všechny známé hrozby a to s nízkým FPR.

Bro byl původně vytvořen jako platforma pro výzkum detekce průniků a analýz datového provozu. I přes to, že obsahuje již existující sadu skriptů, není určen pro někoho, kdo hledá jednoduché „balíkové řešení“. Původně byl určen jako nástroj pro potřeby Unixových odborníků, který by jim pomohl vypořádat se s technikami útočníků a bezpečnostní politikou.

Jak již bylo řečeno, Bro je open-source produkt, který funguje na běžném komerčním hardware. Tímto hardware se rozumí PC hardware, u něhož platí, čím větší je zapracovávané množství dat, tím je logicky zapotřebí výkonnějšího stroje. Díky této kombinaci otevřeného kódu a funkčnosti na PC Bro poskytuje levné řešení pro vyzkoušení alternativních technik ochrany.

Tento produkt má své využití i u společností či organizací, které již využívají některý z komerčních IDS. Tyto možnosti jsou podle [6] následující:

- Ověření výsledků komerčního IDS.
- Dosažení lepších rozhodovacích schopností.
- Kontrolovat kvality bezpečnostní politiky, které nejsou založeny na komerčním IDS.
- Experimentovat s novými metodami a připojit se k výzkumu.

Jako poslední zmínku je nutné uvést, že Bro obsahuje nástroj snort2bro, který je schopen konvertovat signatury ze IDS Snort a tím obohatit vlastní sadu pravidel a snížit hodnotu veličiny FPR.

3.2 OSSEC Host-Based Intrusion Detection System

OSSEC je open-source HIDS, který se zabývá analýzami, kontrolami integrity, monitorováním registrů Windows, detekcí nástrojů, varováním v reálném čase a aktivní odezvou. Pracuje na většině operačních systémů jako je Windows, GNU/Linux, OpenBSD, FreeBSD, MacOS a Solaris. Jedná se o rostoucí projekt využívaný některými poskytovateli internetu, univerzitami a společnostmi. Počet stažení OSSEC dosahuje měsíčně pěti tisíc a stále probíhá aktivní vývoj s dobou aktualizace každé cca tři až čtyři měsíce. Značí se též snadnou instalací – stačí se držet některé instalační příručky. Též je schopen vyvolat aktivní odezvu.

Součástí tohoto produktu je vlastní hlavní aplikace vyžadovaná pro DIDS nebo samostatně stojící instalace. Další aplikací je agent pro Windows, který vyžaduje konfiguraci hlavní instalace pro serverový mód. Grafický výstup je podobně jako u Snort řešen konzolovým výpisem, je však možné doinstalovat speciální aplikaci, jež umožní grafický výstup ve webovém rozhraní. Je tedy možné prohlížet graficky zpracovaná data ve svém oblíbeném prohlížeči.

Všeobecně je doporučováno postupovat v instalaci se standardním nastavením. Nicméně pokročilí uživatelé si samozřejmě mohou funkcionalitu odladit a přizpůsobit ji tak svým potřebám. Za zmínku stojí whitelist, ignorace určitých typů hlášení či změna doby blokace určité IP, jež je standardně stanovena na 10 minut.

Toto řešení se v průběhu času dočkalo mnoha recenzí a ocenění, jako poslední stojí za zmínku, že např. 12. března 2007 byl vybrán v „Top 5 open source security tools in the enterprise“ serverem LinuxWorld.

3.3 OSIRIS Host Integrity Monitoring System

OSIRIS je open-source systém, který periodicky monitoruje jeden nebo více hostitelských systémů, je možné jej provozovat na Windows i Unixových operačních systémech. Pracuje v režimu klient server, kde vlastní server

obsahuje databázi integrity souborů a konfigurací jednotlivých klientů. V určených časových intervalech odesílá klientům konfigurační soubor, spouští na nich sken a výsledky skenování vrací zpět serveru pro porovnání. Detekuje změny v souborovém systému, seznamu uživatelů a skupin, kernelových modulech a další. Podporuje rozličné množství funkcí jako jsou filtrace bezpečnostních hlášení, manuální spouštění skenů a konfigurace jejich atributů. Tyto změny může odesílat administrátorům pomocí e-mailu. Komunikace mezi klientem a serverem je vždy šifrovaná pomocí OpenSSL.

Je vhodné udržovat centrální server co nejvíce chráněný a blokovat k němu přístup až na několik výjimek administrátorských vstupů. Standardně je po instalaci nastaven přístup pouze z adresy 172.0.0.1. Připojení je možné realizovat lokálně či vzdáleně z definované IP adresy. V případě napadení klienta a smazání nebo vyřazení jeho OSIRIS klienty je zjištění napáchaných škod triviální. Stačí pouze zprovoznit tohoto klienta a spustit na něm příslušná skenování. Po jejich dokončení budou výsledky porovnány s databází a případné změny budou odhaleny.

Díky šifrovanému spojení a architektuře klient-server je tento monitorovací systém vhodný pro rozlehlá prostředí s velkým množstvím počítačů, jež mohou být vzdáleny i tisíce kilometrů.

	Platforma	**Licence**	**Zaměření**
BRO	unix	open-source	NIDS
OSSEC	multiplatformní	open-source	HIDS
OSIRIS	multiplatformní	free software	HIMS

Tab. 3.3.0.1 Klíčové parametry uvedených systémů detekce průniku

Kapitola 4

Snort Intrusion Detection System

Snort je program určený pro síťovou detekci průniků. Díky jeho vývoji pod open source licencí je možné jej provozovat bezplatně, jsou k němu v hojné míře vytvářena pravidla a není zde ani nouze o uvolňování nových verzí. Je také dostupný v mnoha verzích pro různé operační systémy.

Původně byl koncipován pouze jako tzv. paket sniffer, dnes již pracuje i jako tzv. paket logger a NIDS. Snort pracuje na bázi porovnávání síťového provozu se svou databází pravidel, tzn. pracuje na bázi signatur. Kombinuje výhody detekce pomocí signatur, anomálií i protokolů. Je možné jej používat s řadou zásuvných modulů (plug-in), které např. umožňují výstup do databází SQL či umožňují unifikovaný výstup.

4.1 Systémové požadavky

Zde je nejdříve nutné uvědomit si několik věcí. Jednou je, že Snort generuje značné množství dat a proto je nutné počítat s dostatečně velkým pevným diskem. Další neméně důležitou potřebou je bezpečné monitorování systému ze vzdáleného přístupu. Vlastní řídící server i sondy mohou být a často jsou v síti fyzicky značně vzdáleny, také se mnohdy nacházejí na nepřístupných místech. Snort z tohoto důvodu používá Secure Shell (SSH) a Apache se Secure Socket Layer (SSL) pro Linux, Terminal Services (s právy uživatelů k přístupu na jednotlivé počítače) a Internet Information Server (IIS) pro Windows.

4.1.1 Hardware

Tento systém, jak již bylo zmíněno, vyžaduje značný úložný datový prostor. Zvláště pak, pokud Snort pracuje v režimu NIDS. Obecně by se dalo říci, že na každou sondu je třeba vyčlenit cca 10GB diskového prostoru. V případě, že jsou data skladována na centrálním serveru, musí tento obsahovat alespoň takový pevný disk, jako je součet všech sond, které na něm svá generovaná data shromažďují. Vlastní sondy pak vystačí pouze s prostorem pro

případné malé logovací soubory, které se cyklicky přepisují. Ty lze nastavit na libovolnou velikost např. 128 MB.

Doporučováno je druhé síťové rozhraní. Jedno je použito pro typické účely jako jsou síťové služby, SSH a další komunikaci. Druhé, pracující v promiskuitním módu, se využívá pro účely připojení softwarové sondy. Promiskuitním módem síťové karty se rozumí mód, ve kterém karta nekontroluje cílovou MAC adresu a zpracovává veškeré pakety, které se k ní dostanou. Obě rozhraní musí disponovat potřebnou propustností s ohledem na jejich vytížení.

Jiné požadavky se již neuvádějí. Je běžné, že jakákoli aplikace na silnějším hardware pracuje rychleji než na hardware pomalejším. Proto je vhodné vzít v úvahu i toto a připravit stroje dimenzované na očekávané vytížení. V případě nedostatku výpočetního výkonu dochází k zahazování některých paketů, což vede k nepřesnému provozu a nesprávné odezvě v reálném čase.

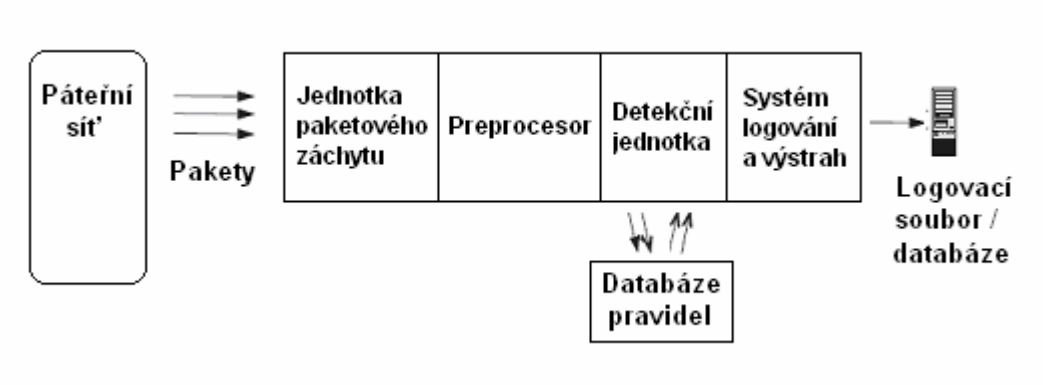
4.1.2 Software

Snort je možné provozovat na jakémkoli moderním operačním systému jako je Windows, Linux, FreeBSD, OpenBSD, NetBSD, MacOS X, MkLinux, Sparc Solaris nebo HP-UX.

Doplňující a podpůrný software, dle ([1], str. 23):

- MySQL, Postgres, Oracle
- Smbclient při použití WinPopup zpráv
- Apache nebo jiný webový server
- PHP nebo Perl, pokud jsou potřeba zásuvné moduly, které je vyžadují
- SSH nebo Terminal Server pro vzdálený přístup

4.2 Architektura IDS Snort



Obr. 4.2.0.1 Architektura IDS Snort

(přeloženo z [1], str. 34)

Preprocesor, detekční jednotka a systém logování a výstrah jsou zásuvné moduly. Tyto moduly se tváří jako součásti vlastního jádra IDS Snort, nicméně jsou z důvodu potřebné snadné modifikace odnímatelné.

Jednotka záchytu paketů (Packet Sniffer) umožňuje aplikaci nebo hardwarovému zařízení naslouchat síťovému provozu. Toto se ovšem netýká telefonních sítí, jež slouží pro přenos hlasu. Obvykle to bývá provoz IP, ale dokáže si poradit i s IPX a AppleTalk. Je schopen poradit si s množstvím protokolů včetně TCP, UDP, ICMP, roubovacích protokolů a IPSec. Množství těchto jednotek interpretuje pakety do pro lidi čitelné formy nebo umožňuje unifikovaný výstup pro další zpracování.

Preprocesor se používá pro hledání určitých nebezpečí a přípravu paketů pro detekční jednotku. Preprocesor poskytuje pakety rozličným zásuvným modulům a pokud zachytí možné nebezpečí, podstoupí je detekční jednotce. Preprocesory se také používají pro paketovou defregmentaci. Pokud jsou přenášena velká data, jsou pakety většinou rozděleny, v Ethernetové síti je hodnota MTU (Maximum Transfer Unit) 1500 bajtů. Před kontrolou je nezbytné tyto pakety nejdříve zkompletovat.

Detekční jednotka je hlavní částí IDS Snort. Přijímá data z preprocesoru i jeho zásuvných modulů a porovnává je s databází pravidel. Tato pravidla jsou dělena do kategorií např. trojské koně, přístup k různým aplikacím, přetečení zásobníku. Jakmile je zjištěna shoda s jednou nebo více signaturami, jsou tato data předána systému logování a výstrah.

Systém logování a výstrah. Tato jednotka vyvolá výstrahu v závislosti na datech, která obdrží z detekční jednotky . Dále je může zapsat do logovacích souborů, jež jsou standardně uloženy v /var/log/snort. Nebo je může v závislosti na instalovaných zásuvných modulech odeslat na jiný počítač, uložit do SQL databáze, zobrazit pomocí webového rozhraní či odeslat e-mailem. Tím dochází k informování správců systému v reálném čase.

4.3 Úskalí a interní bezpečnost Snort

Snort má jako jakýkoli reálný systém či nástroj určité nedostatky, které je nutné během jeho nasazení brát v úvahu.

4.3.1 Úskalí

Existují tři hlavní úskalí, kterými tento jinak silný nástroj trpí. Všechny tyto neduhy lze vhodným způsobem buď eliminovat nebo značně potlačit. Jsou jimi tyto:

- Všechny pakety nejsou zpracovány
- Počet nezachycených průniků
- Počet falešných poplachů

Příčinou prvního jmenovaného problému může být nedostatečný výkon hardware, na kterém Snort pracuje. Může se jednat jak o pomalé síťové rozhraní, popř. pomalé v promiskuitním módu nebo o celkovou pomalost

výpočetního systému. V takovém případě je třeba vzít v úvahu použití rychlejší NIC, procesoru nebo navýšení operační paměti.

Pokud Snort neodhalí žádné nebezpečí a přesto jiné nástroje signalizují podezřelé aktivity, je načase se zamyslet nad možnými příčinami. Ani tato situace není nijak ojedinělá. V takovém případě je vhodné zkontrolovat aktuálnost používaných balíků pravidel, hledat chyby v případných vlastních pravidlech i znovu se zamyslet nad vlastní koncepcí rozvržení sond v síti.

Počet falešných poplachů je neméně nebezpečný. Ve velkém množství výstrah lze jen těžko sledovat, popř. prověřovat všechna hlášení. Je pak velmi snadné opomenout skutečné nebezpečí. Řešením je postupné přizpůsobování pravidel síti a na ní pracujícím službám. Tento proces zabere mnoho času a vyžádá si velké množství prostředků. To je bohužel neduhem všech existujících systémů detekce průniků.

4.3.2 Interní bezpečnost Snort

Snort, stejně tak jako jiné systémy, je sám zranitelný. Pokud je udržován zabezpečený, jsou jím generovaná data důvěryhodnější. Pokud by došlo k průniku a Snort by byl infiltrován, stal by se nepoužitelným do té doby, než by byly všechny jeho komponenty přeinstalovány a stará data vymazána.

Je tedy potřebné udržovat všechny jeho komponenty, zásuvné moduly i podpůrné aplikace aktualizované. Není na škodu jeho hlavní systém chránit firewallem.

Aktualizace IDS Snort může být problematická. Může se změnit syntaxe i interface. To se stalo při vypuštění verze 2.0, která zjednodušila používání pravidel a zrychlila proces detekce až 18x. Nicméně je nutné dodat, že se takto velké změny nedějí příliš často.

4.4 Pravidla

Jak již bylo řečeno, základem celého systému jsou jeho pravidla, jde zároveň o jeho největší sílu. Jazyk pravidel je triviální a velmi intuitivní, lze si jej během krátké doby snadno osvojit.

Každé pravidlo se skládá z hlavičky (head) a volby (options). Hlavička pravidla popisuje jakou akci má Snort vykonat, protokol ke kterému se váže, zdrojovou adresu a port, cílovou adresu a port. Ve volbách je pak možné připojit popis a kontrolovat množství dalších atributů, které Snort může zpracovávat v rozsáhlé knihovně zásuvných modulů.

Hlavičku je možné rozebrat na následujícím příkladu:

```
log tcp $EXTERNAL_NET any -> $HOME_NET 21
```

- `log` - tento atribut říká, že má být generovaný záznam o paketu (nikoli výstraha) uložen do logovacího souboru. Existují i další hodnoty jako alert, který generuje a zaznamená výstrahu o paketu. Pass paket ignoruje.

- `tcp` – určuje protokol, který je v pravidle brán v potaz. Další možnosti jsou UDP, IP, ICMP.
- `$EXTERNAL_NET` je proměnná zastupující IP adresu vnější sítě
- `any` – značí, že zdrojový port může být jakýkoli. Any je možné použít i v případě adres, pokud chceme brát v potaz všechny sítě.
- „`->`" je atributem určujícím směr odkud a kam má paket putovat, aby vyhovoval pravidlu. Je možné použít oboustrannou hodnotu „<>", pak na směru nezáleží.
- `$HOME_NET` je proměnná zastupující IP adresu vnitřní sítě
- `21` – číslo na tomto místě značí číslo cílového portu. Je také možné použít reprezentaci s dvojtečkou. Tato reprezentace slouží k definování rozsahů portů v pravidle. 1:80 určuje rozsah portů 1-80 včetně, :80 jsou všechny porty do 80 včetně, 80: pak porty 80 a více.

Ve volbách pravidla jsou všechny atributy vepsány do společné závorky, oddělují se středníkem a mezi názvem atributu a jeho hodnotou je vepsána dvojtečka.

Další příklad se zabývá právě volbami pravidla:
`(msg: "FTP přístup"; rev: 1;)`

- `msg: „FTP přístup"` – zpráva zobrazovaná výstrahou a logem paketu. V tomto případě je to řetězec „FTP přístup".
- `rev: 1` – informuje o revizi pravidla. Zde se jedná o první revizi.

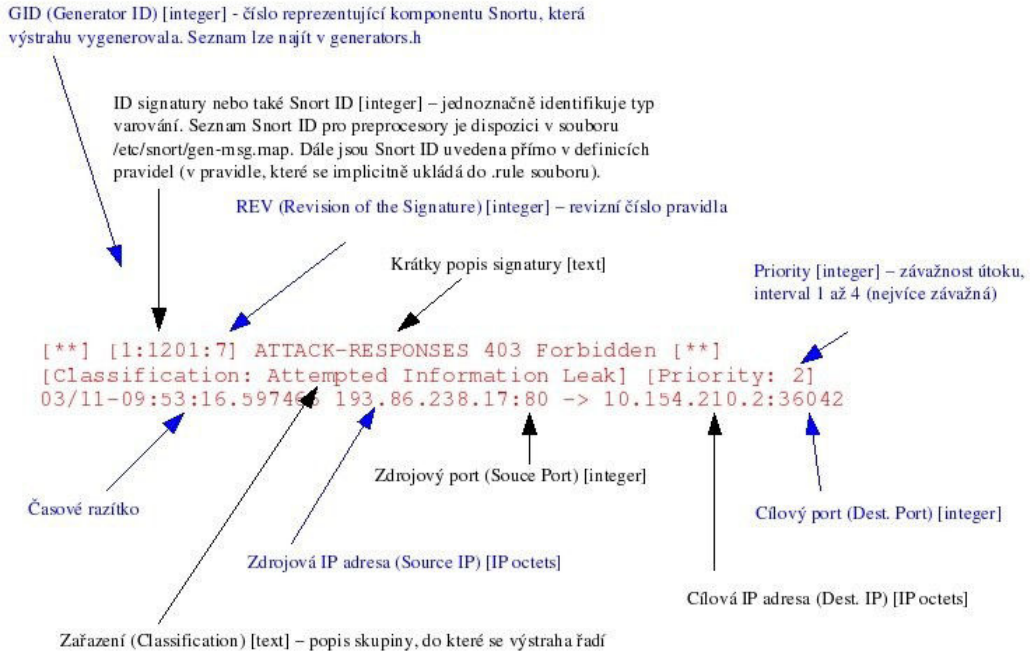
Výčet některých zbylých atributů pravidel, kompletní obsáhlý seznam je možné dohledat v dokumentaci.

- `id: číslo` – testuje id v hlavičce paketu na hodnotu daného čísla
- `content: "|binární řetězec|"` – pátrá v paketech po výskytu uvedeného binárního řetězce.
- `logto: soubor` – paket bude zaznamenán do definovaného souboru
- `nocase` – malá a velká písmena nebudou rozlišována
- `priority: číslo` – číslo reprezentuje vážnost útoku
- `classtype: jméno` - označuje klasifikaci události
- `seq: číslo` – testuje TCP číselné sekvence na určitou hodnotu

Celé ukázkové pravidlo tedy vypadá následovně:

```
log tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg: "FTP
přístup"; rev: 1;)
```

Na závěr kapitoly je vhodné uvést příklad výstrahy, tak jak ji generuje IDS systém Snort. Vhodnou ukázku použil pan Radomír Orkáč ve své semestrální práci na straně 10:



Obr. 4.4.0.1 Příklad výstrahy generované IDS Snort

([20], str. 10)

4.5 Existující analytické nástroje

Účinné vyhodnocování probíhajících aktivit uvnitř informačního systému by bylo nemyslitelné bez patřičného analytického nástroje. Vzhledem k velkému množství záznamů generovaných systémem Snort v průběhu času je potřebné vést podrobné statistiky, grafy a seznamy varování. Žádoucí je i možnost odkazování se na potřebné popsání každého jednotlivého incidentu. Tyto potřeby zaštiťují aplikace s grafickým uživatelským rozhraním. Jejich vlastnosti jsou velmi rozmanité. Od prostého prohlížení a filtrování záznamů až po komplexní tvorbu statistik a grafů na základě rozličných parametrů. Tyto prvky zároveň mají rozdílnou architekturu a i jejich určení, podpora operačních systémů a potřebné programové vybavení se diametrálně liší. V následujících několika podkapitolách budou představeny tři nejhlavnější analytické nástroje.

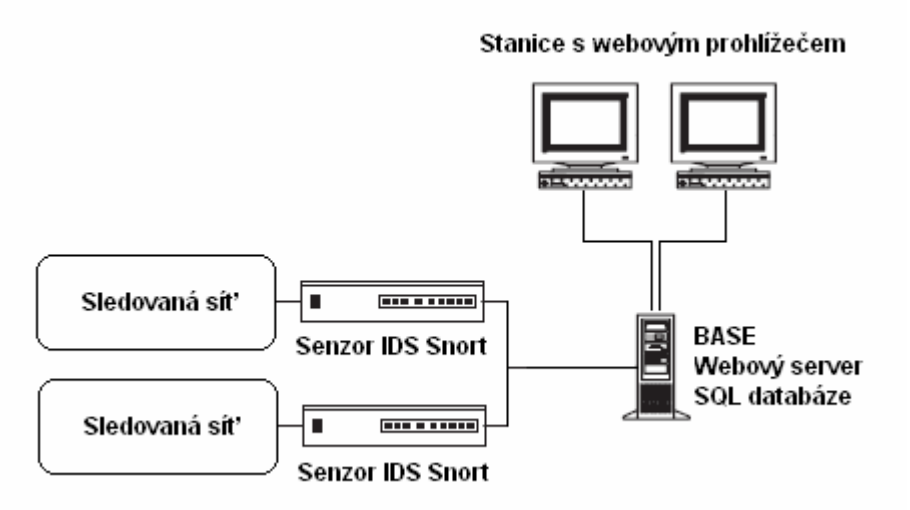
4.5.1 BASE

BASE (Basic Analysis and Security Engine) je založen na projektu ACID (Analysis Console for Intrusion Databases), který začal stagnovat. BASE

odstranil jeho nedostatky a stal se tak jeho vývojovým nástupcem. Tento nástroj analyzuje data zaznamenána v SQL databázi, je založen na jazyce PHP a stále se vyvíjí. Jeho vývoj zaštiťuje skupina kvalifikovaných dobrovolníků, která je schopná reagovat na případné dotazy, výhrady či upozornění.

Architekturu zobrazuje obrázek 4.5.1.1 Architektura BASE. Jak je vidět, BASE analyzuje data získaná prostřednictvím IDS Snort. Tato data mohu být uložena ve všech SQL databázích podporovaných PHP. Oficiálně se jedná o PostgreSQL, MySQL a Microsoft SQL Server. Databázový server i BASE s webovým serverem mohou být jak na stejném počítači, tak i na počítačích různých. Podmínkou ovšem zůstává soudržnost BASE a webového serveru na jednom stroji. Celý systém je prezentován prostřednictvím webových prohlížečů.

Pro zajištění integrity dat a bezpečnosti interní komunikace je celý systém založen na šifrované komunikací prostřednictvím SSL (Secure Sockets Layer) a autentizaci (ověření totožnosti). Přístup je možné rozdělit na uživatele a administrátory, tím jim také lze upravovat pravomoci.



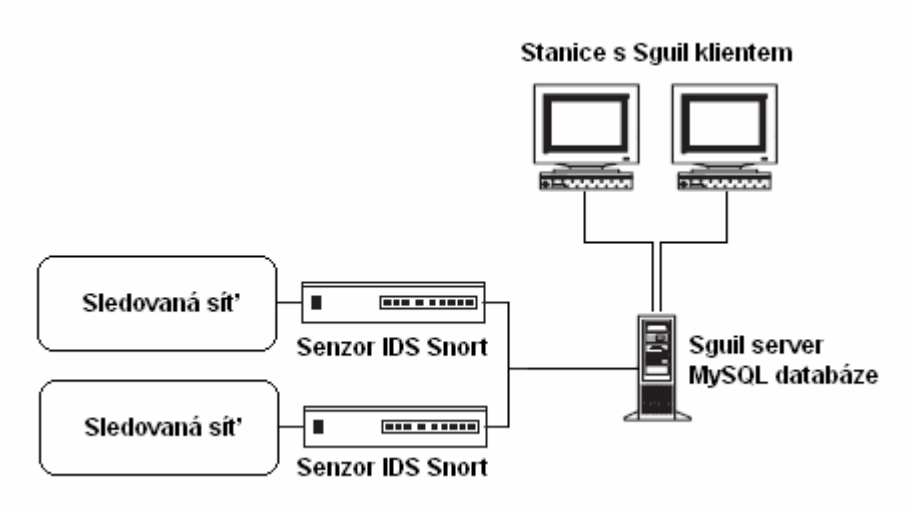
Obr. 4.5.1.1 Architektura BASE

4.5.2 SGUIL

Hlavním mottem projektu Sguil je to, že je tvořen analyzátory síťové bezpečnosti pro analyzátory síťové bezpečnosti. Měl by tedy být maximálně přizpůsoben praktickému použití. Je určen ke zkoumání událostí zaznamenaných pomocí IDS Snort v reálném čase. Jeho klient je psán v tcl/tk a je možné jej provozovat na všech operačních systémech podporujících tcl/tk jako je Windows, Linux, BSD, MacOS, Solaris. Jeho problémem ovšem zůstává problematická instalace. Také je nutno dodat, že poslední verze tcl/tk Sguil nepodporuje.

Architekturu je možné prohlédnout na obrázku 4.5.2.1 Architektura SGUIL. Jak celý princip funguje je patrné z níže uvedeného obrázku. Senzory IDS Snort uloží své výsledky do databáze MySQL, k nimž pak přistupuje serverová aplikace Sguil, k níž se připojují jednotliví klienti za použití jména a

hesla. Teoreticky je možné vše provozovat na jednom počítači, to se nicméně nedoporučuje z výkonnostních důvodů.



Obr. 4.5.2.1 Architektura SGUIL

4.5.3 IDScenter

IDScenter je aplikace určená pro IDS Snort pracující na platformě Windows. Tento nástroj umožňuje centralizovanou správu pravidel, výstrah, podává hlášení a umožňuje další funkce v závislosti na nainstalovaných zásuvných modulech, např. blokování paketů.

IDScenter nabízí vesměs veškeré prvky, kterými disponují jeho UNIXoví kolegové. Disponuje webovým výstupem, editorem pravidel, uživatelskou konfigurací systému Snort, testováním konfigurace, prohlížečem záznamů ze souboru, XML souboru či MySQL databáze.

Ovládání je jako u většiny programů pracujících na platformě Windows velmi intuitivní. Pro svou práci potřebuje pouze IDS Snort 2.x, a WinPCAP 2.3. U těchto programů se vždy doporučuje používat nejnovější verze.

	Platforma	Licence	Databáze	Potřebné aplikace	Počet uživatelských stanic
BASE	multiplatformní	open-source	SQL	IDS Snort, SQL, Apache	mnoho
SGUIL	multiplatformní	open-source	MySQL	IDS Snort, MySQL, tcl/tk	mnoho
IDScenter	windows	open-source	MySQL	IDS Snort, WinPCAP 2.3	jedna

Tab. 4.5.0.1 Porovnání uvedených analytických nástrojů

Kapitola 5

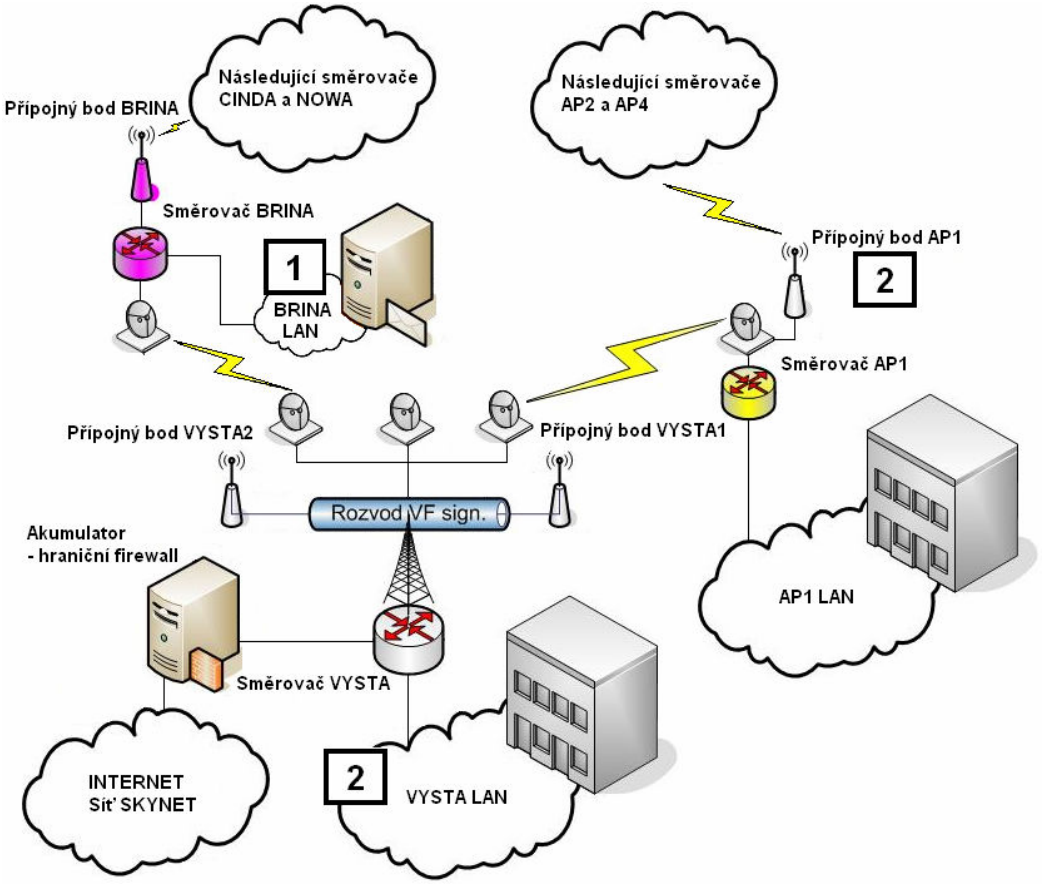
Implementace systému Snort

VODVAS.Net je, jak již bylo zmíněno, občanské sdružení, které je postupně rozšiřováno za pomoci odborných znalostí jeho členů nebo pravidelných příspěvků. Jelikož cílem sdružení je především sdílení internetového připojení a případná interní datová komunikace jeho členů, je třeba zmínit, že neobsahuje příliš mnoho systémů s citlivými daty. I přes to se zde nachází poštovní server, který má v bezpečnosti nejvyšší prioritu. Samozřejmě nelze opomíjet bezpečí členů občanského sdružení, kterých v době psaní této práce je zhruba 150. Člen tohoto sdružení má mimo jiné za povinnost dbát na dodržování zákonů, autorských práv a bezpečnost svých aktivit tak, aby neohrožoval ostatní uživatele systému. Proto bude přikročeno nejen k nepřetržitému hlídání poštovního serveru, ale budou provedena i některá namátková měření na hranicích některých subsítí. Ale o tom více až v následujících podkapitolách.

Jelikož nemám z bezpečnostních důvodů oprávnění zveřejňovat veškeré detaily včetně administrátorských účtů, hesel, popř. některých IP adres, budou některá schémata blokově zjednodušena a přihlašovací účty i hesla budou změněny na fiktivní hodnoty. Také některé IP adresy budou změněny z důvodu ochrany soukromí daných členů. Tyto zásahy ovšem nebudou mít žádné dopady na srozumitelnost postupů či přesnost popsaných řešení.

Jednotlivé bloky architektury VODVAS.Net jsou tvořeny přísně hvězdicovou topologií sítě typu Ethernet, celek je pak spojen do stromové struktury. Toto si lze prohlédnout na následujícím obrázku 5.0.0.1 Architektura VODVAS.Net o.s., kde jsou zobrazeny části sítě, ve kterých bude bakalářská práce realizována.

Ještě bych rád podotkl, že jednotlivá měření nemají za cíl sledování členů občanského sdružení VODVAS.Net. Z tohoto důvodu nebudou zveřejněny IP adresy, které by přímo odkazovaly na konkrétní osoby. Tyto adresy budou zaměněny za všeobecně používané adresy 192.168.X.X, nicméně adresy vně sítě občanského sdružení zůstanou nezměněny.



Obr. 5.0.0.1 Architektura VODVAS.Net o.s.

Jak je z obrázku patrné, prvním zařízením, kterým komunikace přicházející z internetu prochází, je hraniční firewall, dále následuje směrování skupinou směrovačů až k cíli. Také je patrná skutečnost, že ke každému ze zde uvedených směrovačů je připojena lokální metalická síť a bezdrátový přípojný bod (popř. body), na něž je připojen následující směrovač a okolní členové. Upozorňuji, že realita je složitější. Zde uvedené přístupové body zastupují skupinu vysílačů/přijímačů pracujících na frekvencích 2,4 GHz i 5 GHz, k tomuto zjednodušení bylo přistoupeno za účelem zjednodušení schématu. V horní části je možné povšimnout si dvou obláčků, které značí pokračování obdobné architektury ke koncovým směrovačům Cinda, Nowa, AP2 a AP4.

Pro účely této práce jsou zcela jistě nejdůležitější značky míst konání této práce, ty jsou označeny číslem ve čtverci. Číslo jedna značí umístění stálé sondy v subsíti společně s poštovním serverem. Číslo dvě ukazuje na místa plánovaného použití mobilní sondy. Adresa každého veřejného zařízení či člena je přístupná všem uvnitř sítě VODVAS.Net. Z tohoto důvodu, jak již bylo zmíněno, budou v ukázkách interní adresy změněny na privátní rozsah ve tvaru 192.168.X.X.

5.1 Očekávání a metodologie

Aktivní členové VODVAS.Net o.s. v čele s radou sdružení očekávají potvrzení bezpečnosti poštovního serveru. Zde se vzhledem k jeho zabezpečení a malému vytížení neočekávají bezpečnostní incidenty. Plán umístit zde stálou sondu má především význam do budoucna pro zabezpečení některých plánovaných služeb.

Dále sledováním běžných aktivit v některých lokalitách odhalit a zhodnotit možná rizika, která zde mohou vznikat. Za problematické se očekávají následující služby:

- peer-to-peer sítě, především populární torrent
- externí datová úložiště
- komunikačních protokoly v reálném čase (instant messenger), především ICQ
- různé webové stránky pochybného obsahu

Mobilní sonda i bezpečnostní server jsou v zásadě stejná zařízení, jediným rozdílem je hardware, na kterém pracují. Vzhledem k situaci, kdy se očekává nárůst síťového vybavení okolo poštovního serveru, bylo schváleno vytvoření dostatečně výkonného počítače, jenž by vyhovoval i po případném upgradu systému, ač pro stávající situaci není potřebný. Pro mobilní řešení se počítá s notebookem s dostatečně silným hardware. Tento je ovšem uvolněn pouze pro potřeby bakalářské práce. Další sondy zatím nebyly radou sdružení povoleny. Rada čeká na výsledky práce a odzkoušení stability systému. V případě úspěchu bylo předběžně uvažováno o instalaci sondy Snort na router VYSTA, který je tvořen velmi výkonným serverem se 4 jádrovým procesorem. Tato sonda by ukládala výsledky svého měření na již připravovaný bezpečnostní server. Vzhledem ke klíčovosti směrovače VYSTA je nejdříve nutné provést řadu testů, než bude možné k této akci přistoupit. Tato sonda by již monitorovala komunikaci pouze těch členů, kteří by si výslovně přáli zaštítění systémem Snort. V individuálních případech by též bylo možné na přání zhotovit soukromé sondy provázané s hlavním serverem a vytvoření analytických uživatelských přístupů.

Konfigurace bezpečnostního serveru:

- CPU: Intel Pentium 4 2,6 GHz
- 256 MB DDR2 667 MHz
- 80GB HDD
- 100 Mbps síťová karta

- poznámka: připravuje se případné navýšení operační paměti, upgrade síťového adaptéru na 1Gbps, systém je postaven na kvalitní serverové základní desce

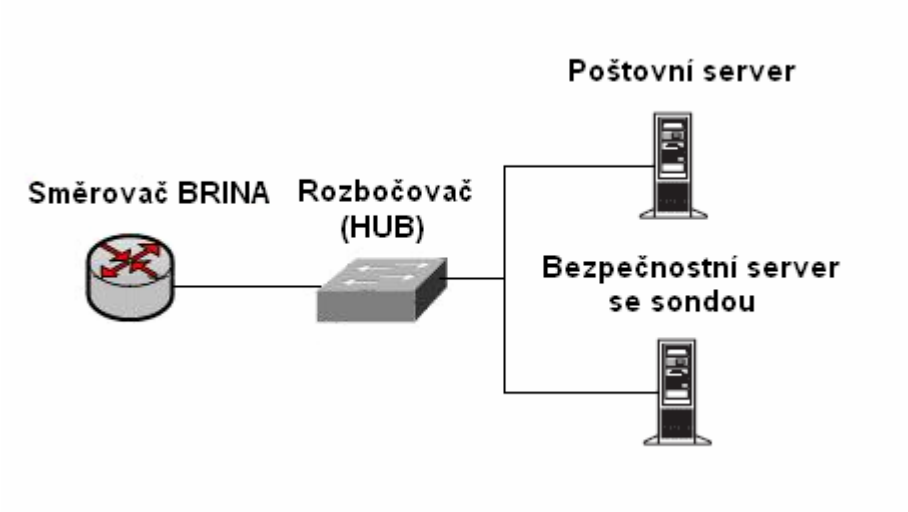
Konfigurace mobilní sondy:

- CPU: AMD Sempron 3400+
- 1,32 GB DDR2 667 MHz

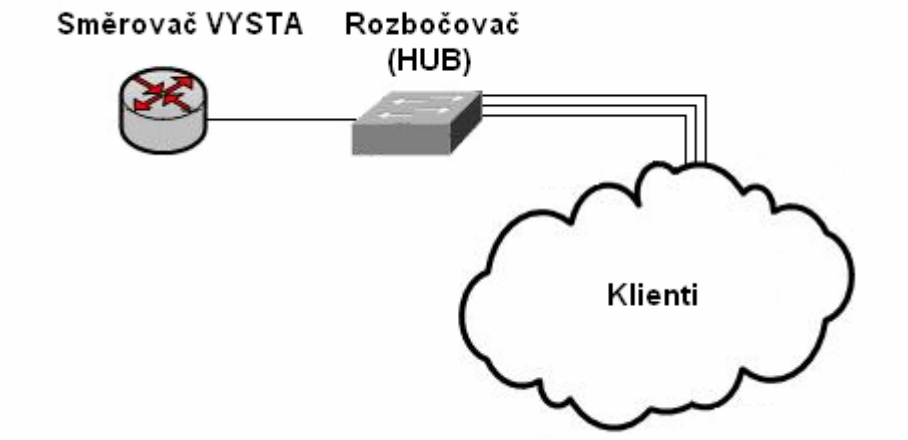
- 80GB HDD
- 100 Mbps síťová karta

Oba tyto přístroje budou pracovat na platformě Linux (konkrétně Ubuntu 8.04 Server Edition) v kombinaci Snort + Barnyard + BASE + MySQL + PHP + Apache.

Sondy budou připojeny k aktivním prvkům, které jim budou zrcadlit popř. rozbočovat síťovou komunikaci. Mezi tyto prvky se řadí managementovatelné přepínače (switch) s funkcí zrcadlení a dostatečně rychlé rozbočovače (HUB). Jak bude toto zapojení vypadat, je možné shlédnout na obrázku 5.1.0.1 Zapojení pevné sondy. Zapojení mobilní sondy bude vždy improvizovaným řešením připraveným tak, aby pro potřeby této práce stačilo. Práce této sondy bude dokončena po plánovaném měření a přijetím opatření či stanoviska. Ukázku takového řešení lze nalézt na obrázku 5.1.0.2 Zapojení mobilní sondy.



Obr. 5.1.0.1 Zapojení pevné sondy



Obr. 5.1.0.2 Zapojení mobilní sondy

5.2 Instalace systému

Instalace celého systému bude rozdělena do jednotlivých podkapitol, které budou zaměřeny na jednotlivé softwarové nástroje. Postup bude obsahovat i několik testů, které prověří dosavadní funkčnost v daném bodě přípravy.

5.2.1 Operační systém

Za operační systém byla zvolena stabilní verze Ubuntu 8.04 Server Edition. Byl instalován a konfigurován pod DHCP, čímž se zjednodušila instalace. Nastavení statických hodnot bude provedeno až v závěru celého instalačního procesu. Operační systém lze bezplatně stáhnout z internetu např. zde:

`http://www.ubuntu.com/getubuntu/download`

Po zavedení instalačního programu z CD je uživatel nejdříve dotázán na jazyk instalace. Při instalaci jsem zvolil jazyk anglický, všeobecně je totiž brán za standard.

- `English`

Následuje volba započetí instalace.

- `Install Ubuntu Server`

Dalším krokem je volba jazyka pro instalační proces. Tento jazyk bude zvolen jako standardní jazyk operačního systému. Doporučuji jazyk anglický.

- `English`

Instalační program se dotáže na lokalitu počítače. Vzhledem ke skutečnosti, při níž byl zatím upřednostňován anglický jazyk, instalátor nabídne Spojené státy. Tuto volbu však není vhodné použít s odkazem na budoucí nastavení tzv. zrcadel (mirrors). Proto je třeba vybrat jinou zemi a zvolit Českou republiku.

- `other`
- `Czech Republic`

Je možné pokusit se o vyhledání popisku klávesnice, nicméně tato funkce je při serverovém využití bezpředmětná.

- `No`

Nyní je třeba zvolit typ klávesnice, kterou bude operační systém používat. Jednoznačně jsem zvolil klávesnici anglickou – Spojené státy.

- `USA`

Uživatel bude upozorněn, že existuje více klávesnic Spojených států a dostane se do jejich rozšířené nabídky. Osobně jsem zvolil standardní klávesnici USA.

- `USA`

Instalátor zjistí konfiguraci počítače, na který instaluje, připraví se na něj a následuje dotaz na pojmenování hostitelského systému. Je možné vyplnit cokoli, např. pouze ubuntu. Ve skutečnosti jsem zvolil jméno v rámci tradic VODVAS.Net o.s.

- `Ubuntu`

Nyní přichází na řadu jedna z nejklíčovějších částí instalace linuxového operačního systému. A sice rozdělení disků na oddíly a nastavení umístění jednotlivých složek po oddílech a vlastností těchto oddílů. Zvolil jsem následující rozvržení:

Manuální management oddílů pro dosažení maximálního možného přizpůsobení aktuálním potřebám.

- `Manual`

Volba disku celého existujícího pevného disku a jeho potvrzení.

- `<název pevného disku>`
- `Yes`

Tvorba oddílů musí splňovat pravidla, která stanovují tvůrci operačního systému i pravidla bezpečnosti.

Oddíl, z nějž bude zaváděn operační systém, musí být chráněn proti pozdějšímu zápisu a není u něj tedy nutné zaznamenávat poslední přístup k souborům a složkám. Neexistuje ani potřeba žurnálovacího souborového systému a vystačí si s kapacitou 300 MB. Je nezbytné u něj zavést bootovací značku.

- `<výběr volného místa>`
- `Create a new partition`
- `300 MB`
- `Primary`
- `Beginning`
- `Use as: Ext2 file systém`
  `Mount point: /boot`
  `Mount options: notime, ro`
  `Label: boot`

Bootable flag: on

- Done setting up the partition

Naproti tomu odkládací soubor paměti tzv. swap vyžaduje cca dvojnásobek operační paměti, maximálně však cca 1500 MB. Souborový systém je nutno nastavit na swap area.

- <výběr volného místa>
- Create a new partition
- 510 MB
- Primary
- Beginning
- Use as: swap area
- Done setting up the partition

Konečně do oddílu root se uloží vše ostatní, proto je potřeba žurnálovacího souborového systému i záznamu přístupu k souborům a složkám.

- <výběr volného místa>
- Create a new partition
- <zbytek místa> GB
- Primary
- Beginning
- Use as: Ext3 file systém

  Mount point: /

  Mount options: realtime

  Label: root
- Done setting up the partition

Nyní nastal čas dokončit tvorbu oddílů.

- Finish partitioning and write changes to disc
- Yes

Instalační program nainstaluje systém dle všech dříve uvedených parametrů.

Dalším krokem je volba jména, celého jména , hesla uživatelského účtu a ověření tohoto hesla. Opět z bezpečnostních důvodů uvádím fiktivní hodnoty.

- user

- `user`
- `passwd`
- `passwd`

Jelikož nepoužíváme proxy server, byla následující volba ponechána prázdná.

-

Instalátor prozkoumá tzv. zrcadla (mirrors). Odkud budou stahovány aktualizační a jiné balíčky.

Nyní je čas vybrat volitelné aplikace. Osobně velmi doporučuji vybrat LAMP server, který je jednou z velkých výhod distribuce Ubuntu. Nainstaluje následující: Apache2, MySQL, PHP5. Na škodu není ani instalace SSH serveru umožňujícímu vzdálený přístup např. skrze aplikaci Putty. Oba tyto prvky jsem zvolil.

- `LAMP server`
- `OpenSSL server`

Instalující uživatel bude vyzván k volbě hesla pro uživatele root MySQL. Opět lze volit zcela libovolně i zde uvádím zástupné hodnoty.

- `passwd`
- `passwd`

5.2.2 Příprava a aktualizace operačního systému

V této fázi instalace je nanejvýš vhodné systém nejdříve aktualizovat. Toho lze snadno dosáhnou příkazy update a upgrade. Ještě před tím je ovšem nutné znovu připojit bootovací oddíl a umožnit mu možnost zápisu. Toto bude platit do restartu počítače.

- `sudo mount –n –o remount , rw /dev/sda1 /boot`
- `sudo apt-get update`
- `sudo apt- get upgrade`

Nyní dozrál čas na instalaci všech balíčků potřebných pro instalaci celého systému. Tyto balíčky jsou uvedeny v tabulce 5.2.2.1 Potřebné balíčky. Poslední dva zvýrazněné nejsou nezbytné, jedná se však o výraznou pomoc při práci v příkazovém řádku. Souborový manager Midnight Commander (mc) obsahuje řadu funkcí, mimo jiné i vlastní textový editor. Ovšem na škodu není ani vynikající textový editor vim, kterému někteří správci dávají přednost. Vše je možné realizovat příkazem:

- sudo apt- get install <jméno balíčku>

build-essential	libpcre3-dev	ssl-cert
php5-gd	autoconf	libssl-dev
libmysqlclient15-dev	automake1.9	libtool
libcap0.8-dev	openssl	php5-cli
mysql-client	php-pear	libphp-adodb
libnet1	php-mail	**mc**
libret1-dev	php-mail-mime	**vim**

Tab. 5.2.2.1 Potřebné balíčky

5.2.3 Instalace a konfigurace systému Snort

Tento krok bude popisovat instalaci systému Snort, přidání jeho uživatele, nastavení konfiguračního souboru a v neposlední řadě i osazení systému aktuálními bezpečnostními pravidly.

Pro linuxové distribuce Ubuntu a Debian existuje možnost instalace systému Snort z balíčku. Ovšem domnívám se, že vzhledem k tendenci autorů některých balíčků zjednodušovat aplikace a vypouštět některé části, popř. funkce, není toto příliš vhodné řešení a je z bezpečnostního i kvalitativního hlediska lepší použít originální instalační soubor. Ten je možné bezplatně stáhnout a nainstalovat. To vše umožní následující série příkazů:

- `cd/usr/local/src`
- `sudo wget http://www.snort.org/dl/snort-2.8.3.2.tar.gz`
- `sudo tar xvzf snort-2.8.3.2.tar.gz`
- `cd snort-2.8.3.2`
- `sudo ./configure --with-mysql --enable-dynamicplugin`
- `sudo make`
- `sudo make install`

Následuje vytvoření cílových složek, přidání skupiny s uživatelem i následné přidělení vlastnictví. Zde opět přistupuji ke kroku zatajení skutečného jména a skupiny a uvádím zástupné hodnoty.

- `sudo mkdir /etc/snort`
- `sudo mkdir /var/log/snort`
- `sudo groupadd snortu`
- `useradd -g snortg snortu`

- `sudo chown snortu:snortg /var/log/snort`

V této chvíli je nezbytné nejdříve zkopírovat veškeré soubory uchovávající nastavení do předpřipravených složek. Upozorňuji, že po konci řádky navazuje následující řádek okamžitě a bez mezery.

- `sudo cd /usr/local/src/snort-2.8.3.2/etc/*.conf* /etc/snort`
- `sudo cd /usr/local/src/snort-2.8.3.2/etc/*.map /etc/snort`

Po dokončení vlastní instalace IDS Snort je nezbytné osadit jej nejnovější dostupnou sadou pravidel. Politika pravidel organizace Snort rozděluje přístup k oficiálním sadám pravidel na tři vrstvy. První vrstvou jsou uživatelé či organizace, které si za update v reálném čase platí. Pro tuto vrstvu jsou aktualizace pravidel ihned k dispozici. Druhou vrstvou jsou registrovaní uživatelé, kterým je stahování aktualizací umožněno bez poplatku s třiceti denním zpožděním. Poslední jsou neregistrovaní uživatelé, kteří si mohou stahovat hotové aktualizované balíky pravidel až tehdy, když je uvolněna nová oficiální sada.

V této práci jsem použil onu prostřední variantu, kterou nelze jinak než doporučit. Registrovat se lze na oficiálních stránkách organizace Snort, uživateli je obratem poslán e-mail, který obsahuje nezbytné přihlašovací údaje. Bohužel se mi nepovedlo stahovat přímo z příkazového řádku, proto jsem pro tento případ použil klasické stahování pod Windows a následné umístění na soukromý FTP server. Z tohoto serveru jsem již velmi snadno stáhl aktualizace do obou zařízení. Po rozbalení souboru je nutné jednotlivé složky zkopírovat do adresáře /etc/snort.

- cd /home
- sudo mkdir download
- cd download
- sudo wget <celá URL FTP serveru>/snortrules-snapshot-CURRENT.tar
- sudo tar xvzf snortrules-snapshot-CURRENT.tar.gz
- sudo cp ./so_rules /etc/snort
- sudo cp ./rules /etc/snort
- sudo cp ./doc /etc/snort
- sudo cp ./etc /etc/snort

V okamžiku, kdy je Snort nainstalován a jeho pravidla jsou aktuální, nastal ten správný čas na jeho konfiguraci a otestování funkčnosti. První kroky vedou k editaci konfiguračního souboru. To lze provést například příkazem:

- `vim /etc/snort/snort.conf`

Zde je nutné přepsat následující řádky na uvedené hodnoty. Dodávám, že hodnota HOME_NET je vždy nastavena na aktuální měřenou síť či subsíť – z tohoto důvodu se u jednotlivých sond liší.

```
var RULE_PATH /etc/snort/rules
var HOME_NET 192.168.1.0/24
var EXTERNAL_NET !$HOME_NET
```

Pro otestování funkčnosti jsem zavedl jednoduché pravidlo v lokálních pravidlech.

- `sudo /etc/snort/rules/local.rules`

Toto pravidlo jednoduše pojmenované „Test“ označí jakýkoli paket za podezřelý:

```
Alert tcp any any -> any any (msg:"Test"; 
sid:1000002;)
```

Nyní je konečně možné spustit aplikaci Snort. Jelikož se v mých sondách objevovalo jen jedno síťové rozhraní, nebylo nutné konfigurovat při spuštění parametrem “– i <rozhraní>“ rozhraní, jež má být určeno jako monitorovací.

- `/usr/local/bin/snort -Dq -u snortu -g snort -c /etc/snort/snort.conf`

Po spuštění tohoto příkazu by mělo být v souboru /var/log/syslog jeden z posledních řádků upozornění:

```
Snort[1731]: Snort initialization completed
successfully (pid=1731)
```

Posledním krokem je ověření funkčnosti dosavadního postupu. V souboru /var/log/messages by měly při síťové komunikaci přibývat varovná hlášení o každém prošlém paketu.

5.2.4 Konfigurace MySQL serveru a logování do databáze

Klíčovou vlastností IDS Snort je logování do databáze, proto je následujícím logickým krokem nastavení MySQL databáze a posléze i výstupu nainstalovaného senzoru do této databáze.

Nejprve je nutné přihlásit se jako root MySQL databáze, vytvořit databázi, vytvořit uživatele a jeho heslo, přiřadit tomuto uživateli práva k databázi. Zde nezbývá než upozornit na syntaxi zadávání hesla. To se skládá z

pomlčky, písmene „p“ a vlastního hesla. Celý tento řetězec se píše bez mezer. I v tomto případě neuvádím reálně používané hodnoty.

Přihlášení a vytvoření databáze:

- `mysql –u root –ppasswd`
- `mysql>Create database snortdb;`

Vytvoření uživatele, jeho hesla, definování jeho privilegií:

- `GRANT ALL PRIVILEGES ON snortdb.* TO dbuser@localhost IDENTIFIED BY 'passwd2' WITH GRANT OPTION;`
- `mysql>quit`

Dále je potřeba v databázi vytvořit příslušné schéma, v němž bude možné uchovávat data.

- `cd /usr/local/src/snort-2.8.3.2/schemas`
- `mysql –u –ppasswd < create_mysql snort`

Není na škodu databázi zkontrolovat.

- `mysql –u –ppasswd`
- `mysql>use snortdb;`
- `mysql>show tables;`

Po té, co je databáze připravena ukládat data a existuje i k přístupu připravený uživatel, přistoupil jsem ke konfiguraci systému Snort. Tato operace se provede prostou editací konfiguračního souboru.

- `vim /etc/snort/snort.conf`

Je třeba vyhledat řádku pro ukládání do MySQL databáze, následně ji okomentovat a editovat. Tato řádka by po úpravách měla vypadat takto:

```
output database: log, mysql, user=dbuser
password=passwd2 dbname=snortdb host=localhost
```

Výše uvedené změny se projeví až po restartu IDS Snort. Na škodu určitě není ani restart celého systému, je ovšem nutné opět spustit aplikaci Snort již dříve uvedeným příkazem.

Systém by nyní měl ukládat všechny procházející pakety do MySQL databáze. Správnost dosavadní konfigurace je možné ověřit níže uvedeným příkazem. Tento příkaz vypisuje aktuální množství záznamů v databázi:

- `Mysql –uroot –ppasswd –D snortdb –e „SELECT COUNT(*) FROM EVENT“`

5.2.5 Nastavení SSL

Korektní práce systému detekce průniků by byla nemyslitelná bez zabezpečené komunikace jeho jednotlivých prvků. Vzhledem k povaze řešení popisovaného v této práci jde především o šifrování komunikace mezi webovým prohlížečem na vzdálené stanici a vlastní instalovanou sondou. K řešení této problematiky je připravena technologie SSL. Naneštěstí instalovaná verze Ubuntu Server obsahuje známou chybu v podobě chybějících souborů potřebných pro generování bezpečnostních certifikátů. Tento bug lze snadno odstranit uvedeným postupem.

- `cd /home/download`
- `sudo wget http://librarian.launchpad.net/7477840/apache2-ssl.tar.gz`
- `sudo tar xvzf apache2-ssl.tar.gz`
- `sudo mkdir /etc/apache2/ssl`
- `sudo cp ./apache2-ssl-certificate /usr/bin`
- `sudo cp ./ssleay.cnf /usr/share/apache2/`
- `cd /usr/bin/`
- `sudo ./apache2-ssl-certificate`

Po té, co je certifikát úspěšně vytvořen, lze přejít k instalaci modulu.

- `Sudo a2enmod ssl`
- `Sudo /etc/init.d/apache2 force-reload`

Následuje vytvoření virtuálního hostitele pojmenovaného ssl.

- `Sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/ssl`

Tento hostitel je kopií původního standardního hostitele, aby bylo dosaženo správné funkčnosti, je nutné oba hostitele editovat. Prvním z nich je nově vytvořený hostitel ssl, jenž musí být nastaven pro komunikaci skrze port 443. Podmínkou též zůstává povolení SSL a nastavení cesty k certifikátu. Níže pak uvádím ukázku, jak bude vypadat začátek souboru po úpravě.

- `sudo nano –w /etc/apache2/sites-available/ssl`

```
NameVirtualHost *:443
<virtualhost *:443>
ServerAdmin webmaster@localhost

SSLEngine On
SSLCertificateFile /etc/apache2/ssl/Apache.pem

DocumentRoot /var/www/
```

- `sudo nano -w /etc/apache2/sites-available/default`

```
NameVirtualHost *:443
<virtuálnost *:443>
```

Celý postup nastavení SSL se završí povolením virtuálního hostitele:

- `sudo a2ensite ssl`
- `sudo /etc/init.d/apache2 reload`
- `sudo /etc/init.d/apache2 restart`

5.2.6 Instalace a konfigurace BASE

Jak již bylo popsáno, BASE je jednou z předních aplikací umožňujících grafický výstup naměřených aktivit. Mimo značné efektivity je jednou z jeho předností i triviální instalace. V níže uvedené instalaci jsem opět použil pohodlné stažení a umístění na osobní FTP server.

- `cd /var/www/`
- `sudo rm index.html`
- `sudo wget <celá URL FTP serveru>/base-1.4.1.tar.gz`
- `sudo tar xvzf base-1.4.1.tar.gz`
- `sudo mv base-php4 base`
- `chmod 777 base` – jen po dobu konfigurace

Vlastní nastavení BASE se provádí skrze webový prohlížeč. Nejprve je tedy potřeba zadat : <IP adresa sondy>/base. Opět uvádím zástupnou hodnotu IP adresy.

- `192.168.2.20/base`

Po načtení úvodu, je prvním krokem pouhé pokračování v konfiguraci.

- `Continue`

Následuje nastavení jazyka.

- `Czech`

Ve třetím kroku je potřeba nastavit cestu k souboru adodb.

- `/usr/share/php/adodb`

Čtvrtý list obsahuje vyplnění přihlašovacích údajů do MySQL databáze.

- `Database Name: snortdb`
  `Database Host: localhost`
  `Database Port: –` prázdné pro standardní hodnotu
  `Database User Name: dbuser`
  `Database Password: passwd2`

V pátém kroku je vhodné nastavit administrátora systému BASE. I zde neuvádím reálné jméno a heslo. Podotýkám, že je možné provozovat celou řadu administrátorských i uživatelských účtů.

- `Login: base`
  `Celé jméno: base`
  `Heslo: base`

A konečně za šesté stačí kliknout na tlačítko „create baseag“. Dojde k rozšíření databázových tabulek tak, aby byly schopny pojmout i funkcionalitu aplikace BASE.

- `baseag`

Nyní by již mělo být vše nastaveno, BASE by měl zobrazovat obrovské množství záznamů generovaných testovacím pravidlem. K dokončení správné základní funkcionality vedou tedy následující kroky:

- zakomentování testovacího pravidla
- vymazání všech dosavadních hlášení
- restartování celého serveru
- aplikace příkazu chmod 775 na složku base v adresáři /var/www

Jelikož analytický nástroj BASE umožňuje i tvorbu nejrůznějších grafů, je vhodné systém dokonfigurovat a doinstalovat i potřebné balíčky.

- `sudo rm /etc/alternatives/php`
- `sudo ln -s /usr/bin/php5 /etc/alternatives/php5`
- `sudo pear config-set preferred_state alpha`
- `sudo vim /etc/php5/cli/php.ini` – odkomentovat řádku „extension=gd.so“
- `sudo pear install Image_Color`
- `sudo pear install Image_Canvas`
- `sudo pear install Image_Graph`
- `sudo /etc/init.d/apache-ssl restart`

5.2.7 Instalace a konfigurace Barnyard

Barnyard je programem, jenž na sebe přebírá břímě ukládání výstupu IDS Snort do MySQL databáze. Snort tak může ukládat svůj výstup pouze do unifikovaných binárních souborů a veškerou zátěž spojenou s konverzí ponechává aplikaci Barnyard, může se tak více soustředit na analýzu síťové komunikace.

- cd /usr/local/src
- sudo wget http://www.snort.org/dl/barnyard/barnyard-0.2.0.tar.gz
- sudo tar xvzf barnyard-0.2.0.tar.gz
- cd barnyard-0.2.0
- sudo ./configure --enable-mysql
- sudo make
- sudo make install
- sudo cp /usr/local/src/barnyard-0.2.0/etc/barnyard.conf /etc/snort

Po úspěšné instalaci zbývá modifikace konfiguračního souboru IDS Snort.

- sudo vim /etc/snort/snort.conf

Je potřeba zakomentovat řádek umožňující programu Snort ukládat do databáze.

```
#output database: log, mysql, user=dbuser
password=passwd2 dbname=snortdb host=localhost
```

A odkomentovat dva řádky, jež specifikují ukládání do binárních souborů. Číslo 128 znamená, že soubor se bude přepisovat po 128 MB.

```
Output alert_unified: filename snort.alert,
limit 128

Output log_unified: filename snort.log, limit
128
```

Též je nutné modifikovat konfigurační soubor programu Barnyard.

- `sudo vim /etc/snort/barnyard.conf`

V tomto souboru je nutné nastavit jméno hostitele, síťové rozhraní, na němž bude naslouchán a výstup, kam bude Barnyard ukládat. Soubor po editaci obsahuje tedy tyto řádky:

```
Config hostname: ubuntu

Config interface: eth0

Output log_acid_db: mysql, database snortdb,
server localhost, user snortu, password
passwd2, detail full
```

V této chvíli je možné vytvořit waldo soubor, jenž umožní aplikaci Barnyard práci v módu záchytných bodů.

- `cd /etc/snort`
- `sudo vi bylog.waldo`

Obsah souboru bylog.waldo. Třetí řádek je nutno nastavit na poslední časovou známku aplikace Snort. Uvádím jednu z mnoha vygenerovaných.

```
/var/log/snort

snort.log

1237235524

0
```

V této fázi je možné přistoupit ke spuštění programu barnyard. Po spuštění by již mělo dojít k plné operativnosti systému.

- `/usr/local/bin/barnyard -c /etc/snort/barnyard.conf -g /etc/snort/gen-msg.map -s /etc/snort/sid-msg.map -d /var/log/snort -f snort.log -w /etc/snort/bylog.waldo &`

5.2.8 Automatizace spouštění

I když celý systém řádně pracuje, není ještě vše dotaženo do úplného konce. Zbývá příprava skriptu, jenž bude všechny instalované prvky spouštět se správnými parametry hned po startu systému.

- `sudo vim /etc/init.d/snort-barn`

Obsah souboru snort-barn:

```
#!/bin/bash
/usr/local/bin/snort -Dq -u snortu -g snort -c
/etc/snort/snort.conf
/usr/local/bin/barnyard -c
/etc/snort/barnyard.conf -g /etc/snort/gen-msg.map
-s /etc/snort/sid-msg.map -d /var/log/snort -f
snort.log -w /etc/snort/bylog.waldo &
```

- `sudo chmod +x /etc/init.d/snort-barn`
- `sudo update-rc.d snort-barn defaults 95`
- `sudo reboot`

5.2.9 Nastavení statických hodnot síťového rozhraní

Pro nastavení pevné IP adresy a dalších k ní potřebných hodnot je nutné editovat soubory /etc/network/interfaces a /etc/resolf.conf. Upozorňuji, že i zde neuvádím všechny hodnoty reálné.

- sudo vim /etc/network/interfaces

Soubor by měl obsahovat tyto řádky:

```
iface eth0 inet static

address 192.168.2.22
netmask 255.255.255.0
network 192.168.2.0
broadcast 192.168.2.255
gateway 192.168.2.1
```

Zbývá nastavit alespoň jeden DNS server.

- `sudo vim /etc/resolf.conf`

Soubor by měl obsahovat tyto řádky:

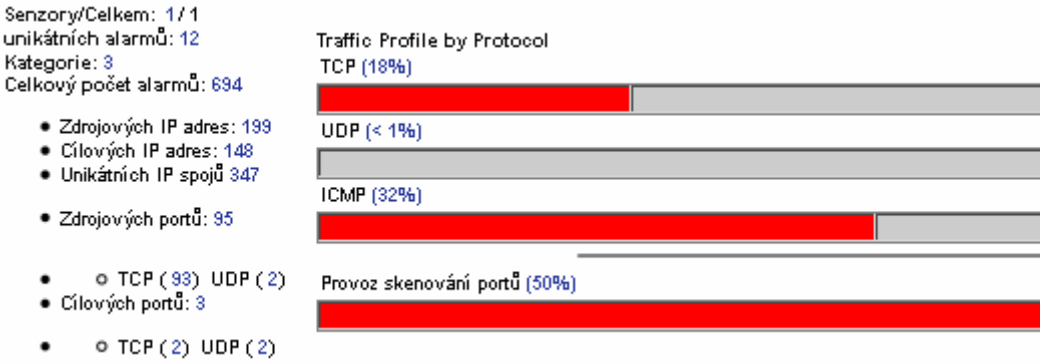
```
nameserver 193.165.254.1
```

Restartováním počítače se celý proces dovrší a bude možné přejít k praktickému nasazení.

5.3 Postřehy z nasazení

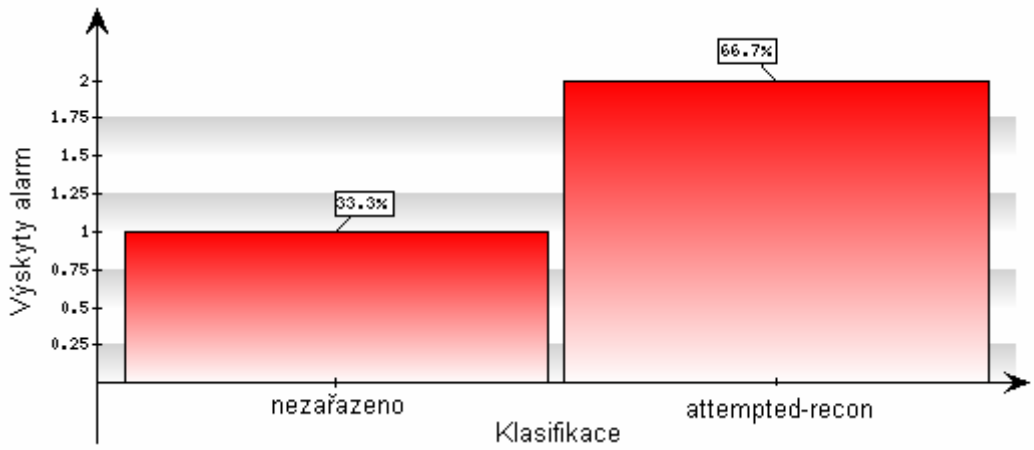
Celý uvedený systém lze ovládat velmi intuitivně. Existuje řada možností jak s rozšiřující se zásobou výstrah nakládat. Jednotlivé funkce jsou navrženy tak, aby co možná nejvíce usnadnily orientaci ve velkém množství záznamů.

Jednou z takových možností je jednoduché třídění uložených výstrah. Základní informace uvádí již střední část úvodního rozcestníku, jejíž výřez je vyobrazen na obrázku 5.3.0.1. Informační výňatek z úvodního rozcestníku. Třídit lze dle rozličných kritérií, např. podle protokolu, sondy, druhu výstrahy, unikátních alarmů, portu či lze vyhledávat dle parametrů. Též je možné zobrazit posledních pět nebo patnáct výstrah.



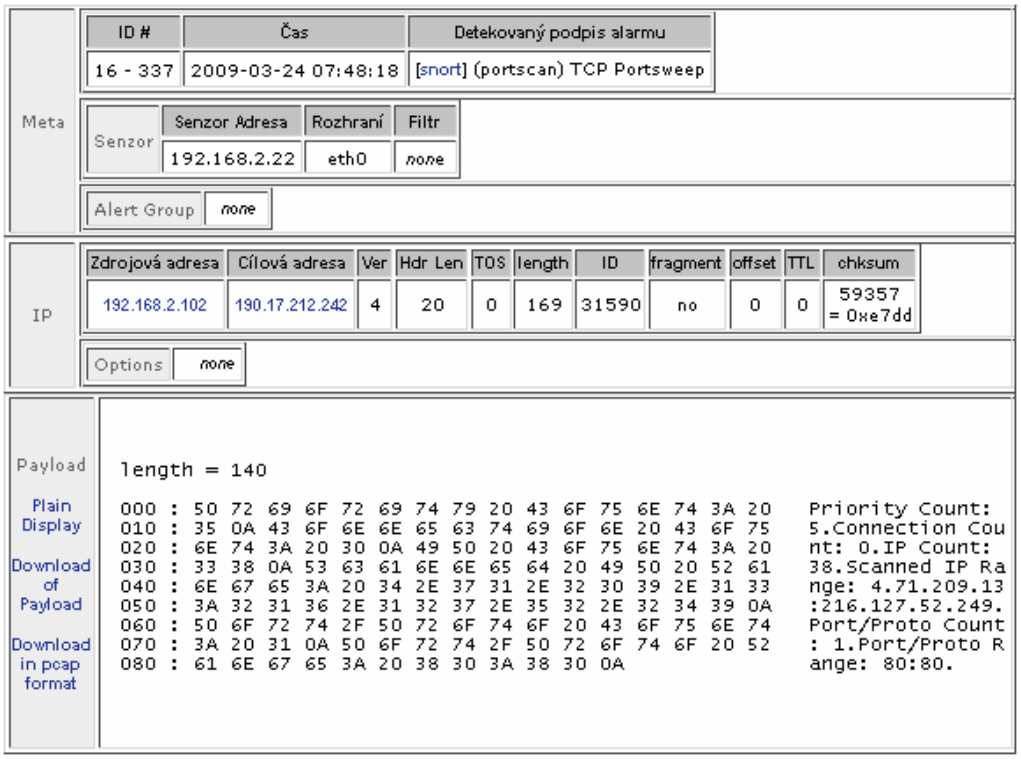
Obr. 5.3.0.1 Informační výňatek z úvodního rozcestníku

Pokročilejší funkcí je tvorba vypovídajících grafů dle některého z šestnácti možných pravidel. Samostatnou grafickou operací je pouhé vykreslení množství alarmů vyskytujících se ve zvoleném časovém období. Celkově lze říci, že tento grafický prvek značně zvyšuje stupeň přehlednosti v uložených záznamech. Nicméně nezbývá než doplnit, že je při tvorbě opravdu vypovídajících grafů nutné značně omezovat vstupní množinu dat. Jinak dochází ke slévání a nečitelnosti nápisů, obrovskému množství vedle sebe sousedících dílů grafu či jiným podobným problémům obdobného ražení. Vše záleží na objemu dat. Ukázku grafu ilustruje obrázek 5.3.0.2 Klasifikace podpisů proti počtu alarmů.



Obr. 5.3.0.2 Klasifikace podpisů proti počtu alarmů

Neméně důležitou funkcí, jenž aplikace BASE umožňuje, je detailní průzkum každého existujícího záznamu. Příklad toho co pohled umožňuje, je možné shlédnout na obrázku 5.3.0.3 Detail záznamu.



Obr. 5.3.0.3 Detail záznamu

Aktualizace pravidel je možné aplikovat vždy tehdy, kdy jsou uvolněné pro danou skupinu uživatelů. V případě této práce se jedná o 30 denní zpoždění za platícími členy, ti mohou aktualizace stahovat a aplikovat v reálném čase např. pomocí aplikace Oinkmaster. Já nahradím nynější složky složkami aktuálními manuálně skrze vzdálený přístup.

V případě potřeby je možné vyčistit databázové tabulky a tím smazat všechny záznamy. Po této operaci je ovšem nutný restart celého systému.

Bohužel musím podotknout, že poněkud zklamala aplikace Barnyard, jež se stala zdrojem dvou nepříjemných chyb a nakonec jsem byl nucen upřednostnit spolehlivost před efektivitou v podobě vypnutí této aplikace. K tomuto kroku jsem dospěl na základě skutečně existující nepotřeby pracovat s aplikací Barnyard na přenosové rychlosti 100 Mb/s, ve které se mé měření odehrávalo. Reálně tato potřeba vzniká až u 1 Gb/s. Co se týče oněch zmíněných chyb, jednalo se především o chybějící popisy jednotlivých výstrah a co bylo ještě závažnější, o chybné určení času dané události. Tyto nepřesnosti se lišily v řádech desítek minut až několika hodin. Po reinstalaci aplikace popř. i celého systému problémy vždy zmizely, ovšem vzápětí se po nějakém čase opět objevily bez zjevných příčin. Zvláštnost tohoto případu je o to větší, že tytéž problémy se vyskytovaly na obou instalovaných strojích. Příčina tohoto problému mi není zřejmá, jelikož jsem nenašel vhodné řešení a ani internetové zdroje mnoho přínosných informací neuvádějí.

Kapitola 6

Závěr

IDS systém Snort nás překvapil svou úměrnou náročností. Očekávali jsme enormní množství varovných hlášení, především z členské komunikace, což se naštěstí nestalo. Samozřejmě při vyšším zatížení se dá očekávat daleko vyšší počet výstrah. Nicméně i v daleko rozsáhlejších datech očekávám dobrou orientaci díky analytickému nástroji BASE. Přehledný management několika set záznamů nečinil obtíže.

Negativně překvapila aplikace Barnyard, jenž se stala zdrojem neočekávaných chyb, jež vyústily až k rozhodnutí vyřadit tento program z provozu a soustředit se na bezchybně pracující kombinaci Snort + BASE + MySQL + PHP + Apache. Nicméně před možným umístěním sondy na směrovač VYSTA bude muset být tento problém odstraněn.

Všeobecně lze říci, že bezpečnostní systém byl radou sdružení schválen jako použitelný a bude mu umožněna expanze v podobě umístění další experimentální sondy (obdobná sondě mobilní) sledující provoz na směrovači VYSTA u těch členů, kteří si budou výslovně přát zaštítění systémem Snort. Pokud se osvědčí, bude vyměněna za sondu stálou, která bude ukládat veškeré výstrahy na již zhotovený bezpečnostní server. V individuálních případech by též bylo možné na přání zhotovit soukromé sondy provázané s hlavním serverem a vytvoření analytických uživatelských přístupů.

Níže uvedená podkapitola již pojednává o konkrétních výsledcích měření a přijatých protiopatření.

Rád bych také podotkl, že po dokončení této bakalářské práce byly všechny záznamy týkající se monitorování členské komunikace zničeny. V žádném případě nebylo cílem sledování konkrétní komunikace, nýbrž utvoření rámcové představy o možných rizicích ohrožujících členy občanského sdružení VODVAS.Net.

6.1 Poznatky VODVAS.Net o.s.

6.1.1 Bezpečnostní server

Bezpečnostní server dosud zaznamenal pět varování. To přisuzuji malému vytížení poštovního serveru s jedinou přilehlou stanicí a přísně nastavenému firewallu. I přes tento zjevně malý počet záznamů rada sdružení tento projekt nepovažuje za zbytečný výdaj prostředků ze dvou důvodů:

- jedná se o kýžený experiment, jenž přináší potřebné bližší seznámení s problematikou a ukazuje možné varianty budoucího využití
- již vytvořený bezpečnostní server se má stát pilířem budoucího monitorovacího systému

Varování zaznamenaná na této sondě jsou uložena v tabulce 6.1.1.1 Záznamy - bezpečnostní server.

Popis	Klasifikace
(http_inspect) DOUBLE DECODING ATTACK	nezařazeno
WEB-CGI redirect access	attempted-recon
WEB-CGI calendar access	attempted-recon
(http_inspect) BARE BYTE UNICODE ENCODING	nezařazeno

Tab. 6.1.1.1 Záznamy - bezpečnostní server

První a poslední varování uvedená v této tabulce byla namířena proti stanici nacházející se v monitorované síti, druhé dvě pak proti poštovnímu serveru. Po prostudování oficiální dokumentace na stránkách organizace Snort byly vyvozeny dva závěry, které se týkají dvou zúčastněných stran. Tyto závěry vycházejí z oficiálních doporučení uvedených u dokumentace každé signatury. Většina těchto doporučení se váže na kontrolu aktualizace operačního systému, záplat jednotlivých aplikací a kontrolu důležitých dat.

Jednou z těchto stran je člen občanského sdružení, jenž byl o incidentu informován a byl mu navrhnut odpovídající postup. Bylo poukázáno na trvající povinnost uchovávat členský systém aktualizovaný a chráněný standardními prostředky. Tyto bezpečnostní prvky nemusí být nutně komerční, proto byly navrženy některé neplacené produkty, jež byly se souhlasem výše zmíněného člena instalovány a aktualizovány. Sada zahrnovala novější verzi antivirového programu, anti-spyware a personální firewall. Operační systém nebyl označen za rizikový, jelikož byl pravidelně automaticky aktualizován. Další operace již osobně provedeny nebyly, a to vzhledem ke skutečnosti, že nikdo z administrátorů sítě občanského sdružení nemá právo zasahovat do soukromí jednotlivých členů. Celá akce byla zakončena příslibem pravidelných aktualizací, týdenním spouštěním kontrol instalovaných bezpečnostních aplikací a též bylo doporučeno prohlédnutí důležitých dat uložených na zmíněném PC.

Druhou zmíněnou stranou byl poštovní server, jenž je plně v pravomoci administrátorů sítě. Tento systém je často aktualizován a nejevil žádné známky průniku. Konzistence databází, administrátorských práv, uživatelů, nastavení aplikací i vlastních dat se jeví nezměněna. Oba výše zmíněné záznamy byly vhledem k popsaným skutečnostem a charakteristice údajného útoku vyhodnoceny jako plané poplachy.

6.1.2 Mobilní sonda

Mobilní sonda bezpečně a levně vyplnila prostor pro jiné možné existující sondy. Stala se tak vhodným dočasným doplňkem celého systému, jenž úspěšně sloužil k splnění těchto cílů:

- zjištění některých možných nebezpečí, jež skýtá používání sítě internet běžným uživatelem, v našem případě členem občanského sdružení
- upřesnění počtu generovaných výstrah s ohledem na počet účastníků komunikace

Ukázalo se, že během provozu nevzniká velké množství různých varování, ale naopak velmi značné množství stále stejných varování. Mobilní sonda za dobu svého působení na daných pozicích zaznamenala necelé dva tisíce obdobných varování, jež se týkají používaných služeb. Výčet těchto varování uvádí tabulka 6.1.1.2 Záznamy - bezpečnostní server. Tuto skutečnost přisuzuji především tomu, že běžný uživatel nepoužívá internet k obrovskému množství jeho využití, nýbrž ke svým oblíbeným činnostem, na něž je zvyklý a jež uspokojují jeho potřeby. Tyto činnosti má mnoho lidí společné, jsou to např. prohlížení elektronické pošty, komunikace po v našich končinách běžných protokolech (ICQ, Skype), stahování různých materiálů, četba zpravodajství, sledování streamovaného videa.

Popis	Klasifikace
ICMP Destination Unreachable Communication Administratively Prohibited	misc-activity
(http_inspect) IIS UNICODE CODEPOINT ENCODING	*nezařazeno*
ICMP Destination Unreachable Communication with Destination Host is Administratively Prohibited	misc-activity
(portscan) TCP Portsweep	*nezařazeno*
(portscan) Open Port	*nezařazeno*
(snort decoder) Bad Traffic Same Src/Dst IP	*nezařazeno*
(http_inspect) OVERSIZE CHUNK ENCODING	*nezařazeno*
(http_inspect) BARE BYTE UNICODE ENCODING	*nezařazeno*
(http_inspect) OVERSIZE REQUEST-URI DIRECTORY	*nezařazeno*
ICMP Destination Unreachable Communication with Destination Network is Administratively Prohibited	misc-activity
(http_inspect) DOUBLE DECODING ATTACK	*nezařazeno*
SQL probe response overflow attempt	attempted-user
ICMP Source Quench	bad-unknown

Tab. 6.1.1.2 Záznamy – mobilní sonda

Dle očekávání se objevilo mnoho hlášení spojených s v dnešní době populární peer-to-peer sítí torrent. Tyto aktivity generovaly dvě nejčastější výstrahy „ICMP Destination Unreachable Communication Administratively Prohibited“ a „(portscan) Open Port“, které tvořili necelých 76 % všech zachycených varování. Překvapením se stal protokol ICQ, jehož aktivita přímo zachycena nebyla, a to přes existenci několika pravidel na něj zaměřených. Došlo i k několika poměrně závažným výstrahám, jako např. OVERSIZE REQUEST-URI DIRECTORY. Cílem byli opět členové VODVAS.Net, kteří jsou známi svými nepříliš bezpečnými aktivitami. Vzhledem k našemu linuxovému zaměření nebylo ani varování „MS-SQL probe response overflow attempt“ označeno za vysoké riziko, jelikož případný útok je směrován na komponenty Microsoft Windows Data Access.

Data z mobilní sondy se ukázala jako velmi kontroverzní. Bylo potvrzeno několik očekávaných aktivit, jež ovšem není možné vzhledem k charakteristice svobodné metropolitní sítě VODVAS.Net, omezit. Došlo by tím k omezení svobody členů občanského sdružení a tím k porušení stanov tohoto občanského sdružení. Zásah by byl možný až tehdy, pokud by prokazatelně došlo k porušení zákonů České republiky nebo k výzamně neohleduplnému chování člena.

Jako opatření byla přijata opětovná výzva k zabezpečení vnitřních systémů a struktur jednotlivých členů a k ohleduplnému chování během používání informačních technologií.

Literatura

[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 přispěvovatelů. ISBN 1-931836-74-4.

[2] UR REHMAN, Rafeeq,. *Intrusion Detection Systems with Snort*. Mary Sudul; Jill Harry. [s.l.] : [s.n.], c2003. 275 s. ISBN 0-13-140733-3.

[3] ŠUMSKÝ, David. Systémy detekce průniku. [s.l.], 2006. 115 s. Masarykova univerzita - Fakulta informatiky. Vedoucí diplomové práce Dr. Václav Matyáš ml. Dostupný z WWW: <http://is.muni.cz/th/51653/fi_m/diplomka.pdf>.

[4] INNELLA, Paul, et al. The Evolution of Intrusion Detection Systems. *Security Focus* [online]. 2001-11-16 [cit. 2008-12-29]. Dostupný z WWW: <http://www.securityfocus.com/infocus/1514>.

[5] BITTO, Ondřej. Rhybaření střídá pharming. *Lupa : server o českém internetu* [online]. 31. 3. 2005 [cit. 2008-12-29]. Dostupný z WWW: <http://www.lupa.cz/clanky/rhybareni-strida-pharming/>.

[6] Lawrence Berkeley National Laboratory. *Bro Intrusion Detection System* [online]. c2003-2008 [cit. 2008-12-29]. Dostupný z WWW: <http://www.bro-ids.org/Overview.html>.

[7] Third Brigade, Inc.. *OSSEC* [online]. c2008 [cit. 2008-12-29]. Dostupný z WWW: <http://www.ossec.net/>.

[8] VISSCHER, Bamm, VIKLUND, Andreas. *Sguil: The Analyst Console for Network Security Monitoring* [online]. c2007 [cit. 2008-12-29]. Dostupný z WWW: <http://sguil.sourceforge.net/index.html>.

[9] *Basic Analysis and Security Engine (BASE) project* [online]. c2000-2008 [cit. 2008-12-29]. Dostupný z WWW: <http://base.secureideas.net/contact.php>.

[10] WOTRING , Brian , POTTER , Bruce . OSIRIS [online]. c2006 [cit. 2009-03-02]. Dostupný z WWW: <http://osiris.shmoo.com/>.

[11] BARR, Joe. An open source security triple play. Linux.com [online]. 7.8.2006 [cit. 2009-03-03]. Dostupný z WWW: <http://www.linux.com/articles/56118>.

[12] VISSCHER, Bamm, VIKLUND, Andreas. Sguil: The Analyst Console for Network Security Monitoring [online]. c2007 [cit. 2009-03-08]. Dostupný z WWW: <http://sguil.sourceforge.net/>.

[13] Engage Security [online]. c2003-2007 [cit. 2008-03-15]. Dostupný z WWW: <http://www.engagesecurity.com/>.

[14] Detect Network Intrusions with Snort/BASE : Essential Open Source Network Administration Tools [online]. [cit. 2009-03-23]. Dostupný z WWW: <http://homepage.mac.com/duling/halfdozen/Snort-Howto.html#d0e43>.

[15] SSL Install Method. Ubuntu documentation [online]. 2009-8-3 [cit. 2009-03-25]. Dostupný z WWW: <https://help.ubuntu.com/community/forum/server/apache2/SSL>.

[16] Lingam's Home On Web [online]. c2009 [cit. 2009-03-25]. Dostupný z WWW: <http://www.lingams.net/?p=30>.

[17] APRIAS, Roman. Systémy detekce průniku v Linuxu [online]. [cit. 2009-04-01]. Dostupný z WWW: <http://www.cs.vsb.cz/grygarek/SPS/projekty0405/IDS/ids.html>.

[18] Snort.org [online]. c2009 [cit. 2009-02-27]. Dostupný z WWW: <http://www.snort.org/pub-bin/sigs.cgi?sid=119:7>.

[19] SCARFONE, Karen, MELL, Peter. Guide To Intrusion Detection and Prevention Systems (IDPS). [s.l.] : [s.n.], 2007. 127 s. Dostupný z WWW: <http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf>.

[20] ORKÁČ, Radomír. IDS Snort. [s.l.], 2006. 22 s. Seminární práce. Dostupný z WWW: <http://www.cs.vsb.cz/grygarek/SPS/projekty0506/Snort.pdf>.

[21] ROESCH, Martin. Writing Snort Rules : How To write Snort rules and keep your sanity [online]. c1999 [cit. 2009-03-18]. Dostupný z WWW: <http://www.ussrback.com/docs/papers/IDS/snort_rules.htm>.

[22] ENDORF, Carl, SCHULZ, Eugene, MELLANDER, Jim. Detekce a prevence počítačového útoku. [s.l.] : GRADA , [2005]. 356 s.

[23] FIRMAN, Andy. 11 step guide to build a Debian based Intrusion Detection Sensor (IDS) with Snort 2.4.5 or Snort 2.6. *Snort.org* [online]. 2001-05-23 [cit. 2009-03-17]. Dostupný z WWW: <http://www.snort.org/docs/setup_guides/deb-snort-howto.pdf>.