INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Tato práce vznikla za podpory projektu TeamIt, jenž je spolufinancován Evropským sociálním fondem a státním rozpočtem České republiky prostřednictvím grantu ESF CZ.1.07/2.3.00/09.0067.

Budování konkurenceschopných výzkumných týmů pro IT

Výzkumná skupina ANT at FIT:

Systémy pro detekci nebezpečného provozu

Technická zpráva

31.srpna 2010

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VUT V BRNĚ
Božetěchova 2, 612 66 Brno
tel.: +420 541 141 144, +420 541 212 219, fax: +420 541 141 170, 270
http://www.fit.vutbr.cz, http://teamit.fit.vutbr.cz

1 Úvod

S rozvojem počítačových sítí stále více narůstá počet útoků na počítačové systémy. Každým rokem přibývají nové bezpečnostní hrozby. Podle zprávy [23] společnosti Symantec, která popisuje bezpečnostní hrozby na Internetu, bylo jen za poslední rok identifikováno více než 240 milionů různých nových nebezpečných programů, což je 100 % nárůst proti loňskému roku. Nejčastěji detekované hrozby jsou podle zprávy viry, trojské koně a síťoví červi. Kromě nebezpečných programů narůstá i množství nevyžádané pošty, která je většinou posílána z napadených počítačů.

Postupně vznikají nejen nástroje pro detekci a ochranu sítě, ale jsou vytvářeny i nástroje pro generování útoku a vytváření škodlivého kódu. S využitím těchto nástrojů je možné prolomit ochranu počítače a nelegálně získat informace, aniž by k tomu útočníci potřebovali hluboké znalosti z oblasti bezpečnosti počítačových sítí. Mezi tyto nástroje patří například program s názvem Zeus (Zbot), který automatizuje proces vytváření škodlivého kódu podle požadavků uživatele a který je možné koupit za pouhých 700 dolarů. Pro získání cizích osobních údajů nebo pro zmocnění se cizího počítače tak není potřeba zkušeného odborníka. Stačí si pouze zakoupit vhodný program.

Metody útočníků se vyvíjejí z jednoduchých do velice důmyslných forem útoků namířených zejména na velké světové korporace, banky nebo vládní subjekty. Postupem času tak vznikla řada různých typů útoků, které je možné rozdělit do několika následujících kategorií:

Zjišťování otevřených portů – (port sken) nejedná se přímo o útok, ale často tato aktivita útoku předchází. Pomocí speciálně vygenerovaného provozu je zjišťováno, jaké porty (služby) jsou otevřené a jak jsou zabezpečené. Existuje celá řada způsobů, jak skenování udělat. Detekci je možné provést jen na základě zjištění odchylky od normálního síťového provozu. Například jsou sledovány síťové toky a je detekován velký počet pokusů o otevření spojení na uzavřených portech.

DoS útoky – slouží k zahlcení počítače velkým množstvím dat tak, aby počítač nebo nějaká služba přestala fungovat pro legitimní uživatele [20]. Počítač má při zahlcení nedostatek zdrojů, a tak dochází k omezení nebo dokonce k odstavení poskytovaných služeb. Útok může být veden z jednoho nebo více počítačů. Pokud je veden z více počítačů, mluví se o tzv. distribuovaném DoS útoku (DDoS). Detekce se nejčastěji provádí sledováním síťových toků a hledání různých odchylek od normálního provozu.

Malware - jedná se o programy, které ke svému šíření využívají známé bezpečnostní chyby nebo důvěřivost uživatelů. Pokud si uživatel nainstaluje program infikovaný malwarem na svůj počítač, může útočník získat důvěrné informace z počítače nebo dokonce využít počítač k napadení jiných počítačů. Malware se nejčastěji šíří ve formě emailu nebo napadením jiného počítače s bezpečnostní chybou, kterou dokáže využít. Pro detekci a obranu se nejčastěji používají antivirové programy nebo systémy, které hledají v síťovém provozu malware pomocí signatur definovaných v podobě řetězců nebo regulárních výrazů.

Penetrační útoky – cílem je převzít kontrolu nad systémem, získat určitou úroveň oprávnění nebo důležitá data. Při útoku je většinou využita chyba programu, která umožňuje útočníkovi nainstalovat do počítače virus nebo malware. Útoky se soustřeďují zejména na získání uživatelských nebo administrátorských oprávnění.

Existuje řada útoků, které jsou dobře známy a v literatuře dobře dokumentovány. Pro takové útoky je možné najít vhodnou signaturu v podobě řetězce nebo regulárního výrazu a hledat je v síťovém provozu. Dobře dokumentované jsou například útoky generované prostřednictvím tzv. exploitů, což jsou programy využívající chyby v aplikacích. Naproti tomu neustále vzniká řada nových útoků, které využívají nové nebo dosud neznámé chyby v zabezpečení a aplikacích. K těmto útokům není možné vytvořit signaturu v podobě řetězce nebo regulárního výrazu. Pro detekci neznámých útoků se proto používají metody založené na detekci anomálií, které mají obecně nižší úspěšnost správné detekce.

V posledních letech je kladen velký důraz na monitorování a bezpečnost počítačových sítí tak, aby byl včas detekován útok a zajištěna účinná ochrana sítě. Byla vytvořena řada nových typů zařízení. Některá dokáží útok pouze detekovat, jiná umožňují současně aplikovat i účinnou obranu. Asi nejrozšířenějším prvkem ochrany proti útokům je paketový filtr, který umožňuje podle zadaných pravidel blokovat část

paketů síťového provozu. Pravidla jsou ale spojena pouze s informacemi z hlaviček paketů, což neumožňuje detekovat a blokovat některé typy útoku. Proto se kromě paketových filtrů používají i systémy, které pro detekci nebezpečného provozu provádí detailní analýzu paketů (Deep Packet Inspection).

2 Paketový filtr

Úkolem paketového filtru (firewallu) je provádět filtraci síťového provozu podle zadané množiny pravidel. Jestli se má paket zahodit nebo propustit se rozhoduje na základě informací v hlavičce paketu. Porovnávají se zejména zdrojové a cílové IP adresy, zdrojové a cílové porty a typy použitého protokolu (ICMP, TCP, UDP). Základní funkcí firewallu je ochránit lokální síť, počítač nebo aplikaci na základě specifikovaných pravidel před nebezpečným provozem nebo nežádoucími službami.

Běžně používané firewally často umožňují filtrovat i na základě stavu komunikace. To znamená, že se pro síťové toky komunikující prostřednictvím protokolu TCP uchovává informace, kdy tok začal a skončil (hlídá se posloupnost SYN, ACK a FIN paketů). V případě stavového firewallu se tak na provoz tekoucí přes síť nedíváme jenom jako na posloupnost paketů, ale chápeme jej jako obousměrnou výměnu paketů v rámci jedné relace mezi dvojicí nebo i více uzly sítě. Ke každé relaci je uchovávána informace o stavu komunikace, která slouží ke kontrole správné posloupnosti příchozích paketů. Pokud příchozí paket neodpovídá korektní posloupnosti komunikace, je automaticky zahozen. K zahození paketu tak například může dojít, pokud přijatý paket neodpovídá žádnému existujícímu TCP spojení.

Firewally se dnes používají zejména v podobě softwarové implementace, která je běžnou součástí operačních systémů, ale existuje i řada hardwarových řešení. S příchodem stále dokonalejších a propracovanějších útoků se ukázalo, že pouze firewally nedokáží zajistit bezpečnost počítačových sítí. Existuje řada útoků, které ze své podstaty nemohou být odhaleny pomocí paketového filtru, protože nejsou spojeny s informacemi v hlavičkách paketů. Navíc se často jedná o útoky na standardní služby a porty, které nelze prostřednictvím pravidel paketového filtru zakázat. Pro detekci sofistikovaných útoků je proto nutné použít detailní analýzu paketů (DPI), kterou využívají *systémy pro detekci nebezpečného provozu*, také označované jako NIDS (Network Intrusion Detection System).

3 Systémy pro detekci nebezpečného provozu

Cílem síťových systémů pro detekci nebezpečného provozu (NIDS) je včasná detekce bezpečnostních incidentů na síti. Systémy provádí detailní analýzu paketů a hledají podezřelé aktivity, jako je skenování portů, šíření virů a další. Pokud systém najde v síťovém provozu bezpečnostní hrozbu, informuje o tom správce nebo uživatele, případně automaticky nebezpečný provoz zablokuje. Může se stát, že je omylem detekován útok i pro legitimní provoz. V takovém případě mluvíme o *falešném poplachu*, který zbytečně zaměstnává správce a vede k nedůvěře ke schopnostem daného systému. NIDS systémy mohou být umístěny na páteřních linkách nebo ke konkrétnímu serveru, případně k přepínači nebo směrovači v lokální síti.

V současné době existují dva základní přístupy realizace NIDS, které se liší v charakteru použitých detekčních metod. Jeden přístup je založený na hledání signatur, kterými jsou popsány útoky a jiné bezpečnostní hrozby. Druhý přístup se snaží pomocí různých metod detekovat anomálie a odchylky od normálního síťového provozu.

Přístup založený na hledání signatur je použit v řadě komerčních zařízení. Využívá rozsáhlou databázi signatur, které charakterizují profil známých bezpečnostních hrozeb, jako jsou viry, exploity nebo DoS útoky. Při analýze síťového provozu jsou pak jednotlivé pakety nebo celé datové toky porovnávány s databází signatur. Pokud je v některém toku nalezena signatura, je detekováno podezřelé chování a vykoná se odpovídající akce. Nejčastěji je daný tok zablokován nebo je pro něj vyhrazena jen malá část přenosového pásma.

Signatury jsou zaměřené na hlavičky a obsah paketů. U hlaviček je definována sada podmínek, které musí být současně splněny. V případě obsahu je signatura specifikována pomocí řetězců nebo regulárních výrazů, které se hledají v datech paketu nebo v celém síťovém toku. Například pro operační systém UNIX popisuje signatura `cat "+ +" > /.rhosts` příkaz, který výrazným způsobem snižuje bezpečnost systému. Pokud jsou pro detekci použity jednoduché řetězce, které dostatečným způsobem nevymezují

danou bezpečnostní hrozbu, dochází ke vzniku falešných poplachů. Z těchto důvodů se často definují signatury pomocí více řetězců. Například pro detekci známého útoků na webové servery se používá signatura složená z řetězců `cgi-bin`, `aglimp se` a `IFS`.

Kromě signatur obsahujících pouze řetězce existují i signatury, které jsou zaměřené jen na nebezpečné nebo nedovolené kombinace v hlavičkách paketů. Příkladem je signatura zaměřená na detekci útoku Winnuke, který způsobuje modrou obrazovku operačního systému Windows. Při tomto útoku je v paketu nastaven cílový port protokolu TCP na číslo 137 (NetBIOS) a v hlavičce paketu je nastaven příznak OOB (Out Of Band). Další známá hlavičková signatura zjišťuje u TCP paketů, jestli nejsou současně nastavené příznaky SYN a FIN, což by znamenalo, že se někdo snaží současně zahájit a ukončit komunikaci.

Z dosud publikovaných metod obrany, byly nejvíce rozšířeny a nejvíce komerčně úspěšné IDS systémy, které využívají k detekci útoků databázi signatur. V současné době tyto systémy generují celosvětově obrat v řádech stovek miliónů dolarů a předpokládá se, že v roce 2012 překročí tento obrat 2 miliardy dolarů. Mezi komerčně úspěšné IDS systémy patří zařízení od firem AXENT, Cisco, CyberSafe, ISS nebo Shadow. Velmi populární jsou i volně šiřitelné systémy Snort a Bro.

Zatímco systémy využívající signatury byly široce nasazeny, systémy založené na detekci anomálií dosud nezískaly na popularitě a zůstávají pouze jako téma řešené výzkumnou komunitou. Proti systémům založených na signaturách má detekce anomálií velkou výhodu v tom, že umožňuje detekovat neznámé bezpečnostní incidenty ještě před jejich rozšířením. Zatímco systémy založené na signaturách jsou *reaktivní* a umožňují reagovat pouze na známé útoky, které před jejich identifikací již napadly a zničily řadu systémů, systémy založené na detekci anomálií jsou *proaktivní* a fungují autonomně. Umožňují zajistit bezpečnost bez jakéhokoliv dalšího ručního zásahu.

NIDS založené na detekci anomálií monitorují síťový provoz a porovnávají jej proti stanovenému profilu normálního provozu. Takový profil se snaží zachytit co je pro danou síť "normální". Zkoumá charakter síťového provozu, procentuální zastoupení a využití jednotlivých protokolů, korektní kombinace čísel portů a zařízení a spoustu dalších charakteristických vlastností. Pokud se objeví v síťovém provozu anomálie, která se významným způsobem liší od normálního profilu, dostává správce nebo uživatel od systému hlášení. Je velmi problematické rozhodnout, co by mělo být považováno ještě za normální a co už vykazuje příznaky anomálie. Za anomálii je například možné považovat, pokud se běžný uživatel místo běžných dvou přihlášení za den najednou přihlásí a odhlásí k počítači dvacetkrát. Dalším příkladem je, když počítač uživatele je používán ve 2 hodiny v noci, kdy nemá nikdo k počítači přístup.

Nevýhodou systémů založených na detekci anomálií je, že často způsobují vysoký počet *falešných poplachů*. Tento problém se projevuje zejména u systémů, které mají velmi restriktivní pravidla pro detekci anomálií. Například je velmi obtížné rozlišit masivní zájem o jednu webovou stránku od distribuovaného DoS (DDoS) útoku. V takovém případě může být systémem detekována anomálie a vyvolán falešný poplach. Změnu profilu normálního provozu a vyvolání falešného poplachu mohou způsobit i změny konfigurace sítě a některé výpadky. Nemusí být také splněn předpoklad, že útok se vyznačuje anomáliemi v síťovém provozu. Inteligentní útočník může použít techniku, která bude způsobovat jen minimální změny v síťovém provozu.

Dalším problémem systémů založených na detekci anomálií je reprezentace normálního síťového provozu. Běžný síťový provoz obsahuje velké množství skenování portů, DoS útoků, síťových virů a červů. S výjimkou simulovaných dat prakticky nelze získat síťový provoz, který neobsahuje žádný útok a je vhodný k trénování detektoru. Pokud získáme data z reálné sítě a není nebezpečný provoz odstraněn, stává se pro systém součástí normálního stavu a detektor anomálií jej nedokáže odhalit.

IDS systémy založené na hledání signatur

Systémy pro detekci podezřelého provozu pomocí signatur monitorují síťový provoz a porovnávají jej s databází signatur známých bezpečnostních incidentů. Signatury jsou definovány pomocí pravidel popisující informace z hlaviček paketů a pravidel popisující obsah paketů. Pravidla pro hlavičky paketů jsou definovány pomocí podmínek aplikovaných na následujících pět položek:

- Zdrojová a cílová IP adresa
- Zdrojový a cílový TCP nebo UDP port
- Protokol

Pravidla popisující prohledávání obsahu paketů nebo síťových toků se skládají z řetězců nebo regulárních výrazů. Zatímco pravidla vztažená k hlavičkám paketů vyžadují klasifikaci vybraných položek z hlaviček paketů pomocí asociativní paměti (TCAM) nebo pomocí speciálních algoritmů [5, 15, 16, 21], hledání řetězců a regulárních výrazů vyžaduje analýzu každého bajtu datové části paketu. Pro hledání řetězců se proto používají algoritmy Aho-Corasickové [1] a Commentz-Walterové [3], které k dosažení vysoké rychlosti vyhledávání využívají předem vypočítanou datovou strukturu. Nejčastěji je v NIDS používán algoritmus Aho-Corasickové.

Vern Paxson v [7] ukázal, že regulární výrazy jsou mnohem efektivnější a flexibilnější pro specifikaci signatur útoků než jenom řetězce. Flexibilita je dána velkou vyjadřovací schopností dosaženou pomocí tříd znaků, sjednocení, volitelnými položkami a dalšími vyjadřovacími prostředky, které regulární výrazy podporují.

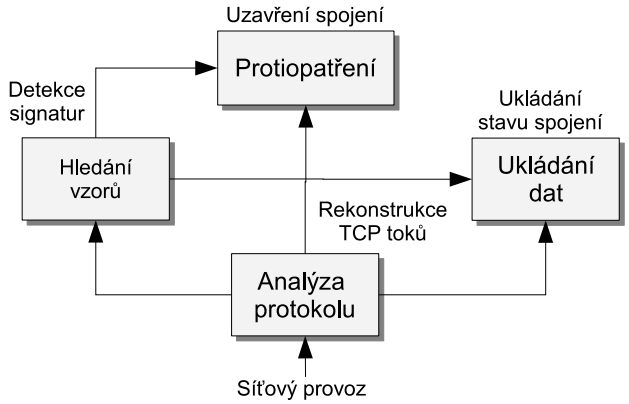
Regulární výrazy jsou použity ve volně dostupných systémech Snort [13] a Bro [7], ale i v komerčních zařízeních, jako je TippingPoint X505 [24] od společnosti 3Com nebo v bezpečnostních síťových zařízeních od společnosti Cisco Systems. Pro hledání regulárních výrazů se nejčastěji používají algoritmy a hardwarové architektury, které jsou založeny na deterministických nebo nedeterministických konečných automatech [9]. Přehled dosud známých hardwarových architektur pro hledání řetězců a regulárních výrazů je možné najít v kapitole **??**.

NIDS systémy založené na hledání signatur musí kromě hledání řetězců nebo regulárních výrazů vykonávat i další operace. Je potřeba analýzu hlaviček paketů [11] a provádět rekonstrukci síťových toků [4, 14], neboť síťová a transportní vrstva je fragmentovaná do jednotlivých paketů a signatury útoků se mohou nacházet na rozhraní dvou po sobě jdoucích paketů. Navíc pakety v rámci jednoho toku mohou přicházet mimo pořadí nebo mohou být duplikovány. Systémy proto musí obsahovat paměť, ve které je uložen stav všech aktivních spojení. Ve vysokorychlostních sítích je možné mít až milióny současně aktivních spojení, což vyžaduje velkou kapacitu paměti a ovlivňuje cenu výsledného zařízení.

Většina NIDS se soustřeďuje pouze na generování hlášení o možném útoku. Některá komerčně dostupná zařízení jsou ale kromě varování schopna vykonat i protiopatření, jako je přerušení nebo omezení podezřelého spojení, změna směrovacích tabulek nebo změna konfigurace paketového filtru. Uvedené rozšíření umožňují NIDS prakticky okamžitě eliminovat bezpečnostní hrozbu bez jakéhokoliv zásahu ze strany člověka.

Na síti se objevuje řada ssh nebo ssl spojení, které využívají šifrovanou komunikaci. Šifrovaný provoz ale nejde jednoduše analyzovat na výskyt řetězců nebo regulárních výrazů. Proto dokonalejší NIDS systémy [19] jsou schopny udržovat privátní klíče hlídaných serverů a umožňují zabezpečenou komunikaci dešifrovat.

Systémy založené na hledání signatur jsou většinou tvořeny několika vzájemně propojenými moduly, které řeší dílčí problémy spojené s detekcí útoků. Moduly řeší problematiku rekonstrukce TCP spojení a rychlé hledání signatur, ale i ukládání informací k síťovým spojením nebo generování různých protiopatření. Vzájemné propojení těchto modulů, je zachyceno na obrázku 1.



Obrázek 1: Architektura NIDS systému založeného na hledání signatur.

Architektura na obrázku 1 odpovídá většině volně dostupných ale i komerčních NIDS. Na obrázku je vidět, že nejprve jsou z jednotlivých paketů rekonstruovány TCP spojení, ve kterých jsou pak hledány signatury různých útoků. V případě nalezení signatury je dané spojení zablokováno a je aktualizována informace o síťovém toku.

IDS systémy založené na hledání anomálií

Systémy založené na hledání anomálií nebyly dosud úspěšně nasazeny, ale do budoucna se předpokládá, že budou postupně rozšiřovat v současné době široce používané NIDS systémy založené na signaturách. Základní výhodou detekce anomálií je automatická možnost nalezení nových dosud neznámých útoků, které už z principu není možné detekovat pomocí systémů využívajících signatury. Detekce anomálií se skládá ze dvou kroků. První krok se nazývá trénování, při kterém se vytváří profil normálního provozu. Druhým krokem je samotná detekce anomálií, kdy jsou v síťovém provozu hledány různé odchylky od normálního provozu. Byla vytvořena řada detekčních algoritmů, které je možné klasifikovat jako statistické metody, metody založené na dolování dat a metody založené na strojovém učení.

Statistické metody

Velké množství statistických metod předpokládá, že anomálie se projeví výrazným nárůstem charakteristik [2] jako je počet přenesených bajtů, počet paketů, častý výskyt určité množiny IP adres a portů. S využitím těchto metod je možné úspěšně detekovat velké změny provozu a detekovat útoky na zahlcení přenosového pásma.

Další statistické metody upozorňují na skutečnost, že útočník může skrýt útok tak, že udrží narušení charakteristiky síťového provozu pod hranicí, kterou využívají k rozhodování detekční mechanizmy. Například útočník může redukovat rychlost skenování portu tak, aby nebyl významně narušen objem přenášeného provozu. Proto se velký počet algoritmů zaměřuje i na malé změny chování síťového provozu. Autoři [18] využili entropii k sumarizaci různých vlastností síťového provozu a ukázali, že analýza vlastností síťového provozu pomocí entropie může vést k relativně přesné a citlivé detekci velkého množství anomálií. Další statistické metody [10, 12] využívají vlastnosti získané korelací mezi různými položkami z hlaviček paketů.

Metody založené na strojovém učení

U metod strojového učení se algoritmy automaticky učí ze vstupu a zpětné vazby s cílem postupně zlepšovat vlastnosti detekce. Na rozdíl od statistických metod, které se snaží detekovat odchylky ve vlastnostech síťového provozu, se metody založené na strojovém učení snaží detekovat anomálie pomocí určitého mechanizmu a pak v závislosti na zpětné vazbě, dané například falešnými poplachy, vylepšovat detekční mechanizmus.

Pro detekci se většinou využívá Bayesovských sítí, které umožňují modelovat pravděpodobnost vazeb mezi různými událostí a predikovat budoucí závislosti. Valdes [25] aplikoval Bayesovské sítě pro detekci anomálií na souvislém bloku síťového provozu a ukázal, že výsledný systém je schopen detekovat DDoS útoky, které by jinak nebylo možné detekovat. Bayesovské sítě byly také použity pro agregaci a potlačení falešných poplachů. Kruegel [17] ukázal, že je možné sbírat informace z více různých senzorů a agregovat informace tak, aby bylo generováno pouze jediné hlášení o incidentu.

Metody založené na dolování dat

Dolování dat pracuje s rozsáhlou množinou dat. Snaží se detekovat určité vzory nebo výrazné odchylky, které jsou jinak jen velmi obtížně detekovatelné. Proto se tyto techniky používají nejen k detekci anomálií, ale i k vytváření profilu normálního provozu.

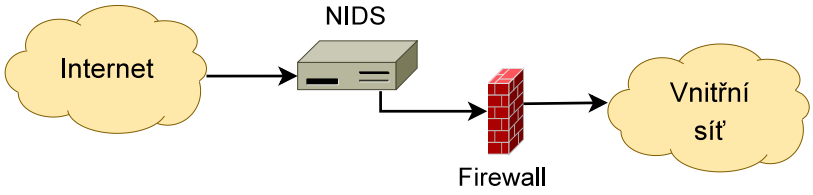
K dolování dat byly použity algoritmy používané ve fuzzy logice. Dickerson [6] použil relativně jednoduchou techniku dolování dat, která umožňuje na základě zpracování síťových dat vytvořit fuzzy pravidla pro detekci útoků. S využitím této techniky ale není možné vytvořit profil normálního síťového provozu. Výsledkem jsou pouze pravidla, které je možné použít pro detekci útoků.

Pro detekci anomálií byly také využity techniky pro hledání shluků, které umožňují hledat vzory ve vícerozměrném prostoru. Detekční schopnosti systémů založených na hledání shluků silně závisejí na velikosti množiny trénovacích dat. Mezi nejznámější algoritmy pro detekci anomálií, které pracují na principu vyhledávaní shluků, patří MINDS [8] a [22].

4 Zapojení IDS systému v síti

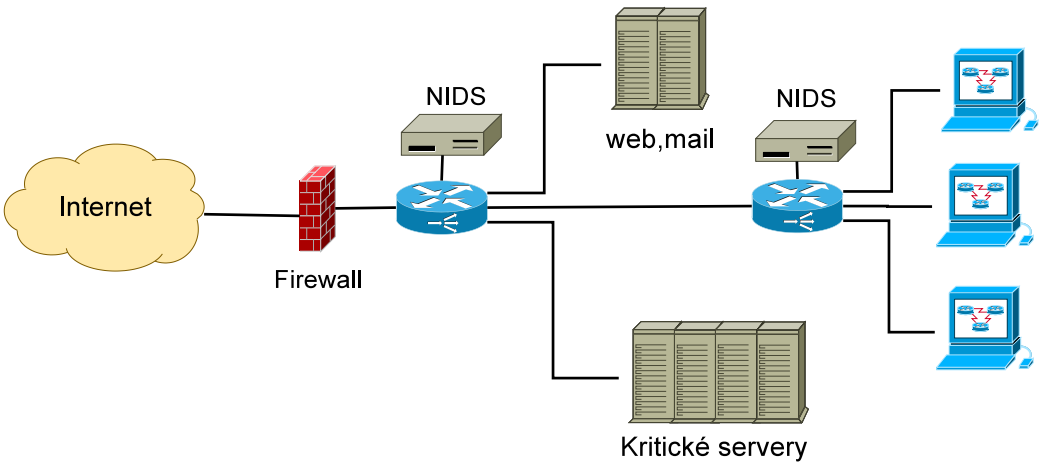
Systémy pro detekci podezřelého provozu mohou být zapojeny mimo vnitřní síť chráněnou firewallem, jak je zachyceno na obrázku 2 nebo mohou být vloženy přímo do vnitřní sítě podle obrázku 3.

V případě umístění NIDS mimo vnitřní síť je analyzován všechen provoz vstupující do vnitřní sítě. Základní výhodou tohoto zapojení je, že je potřeba pouze jedno zařízení, které může sloužit velkému počtu hostitelských počítačů. To znamená, že správa a aktualizace databáze signatur, ale i dlouhodobé zajištění správné konfigurace zařízení je relativně jednoduché. Na druhou stranu útoky vyvolané z prostoru chráněného firewallem není možné nijak detekovat, neboť nebezpečný provoz neprochází přes NIDS. Je také potřeba poznamenat, že systém v tomto zapojení detekuje i nebezpečné incidenty, které jsou blokovány firewallem. Je tak generováno velké procento falešných poplachů.



Obrázek 2: Zapojení NIDS mimo vnitřní síť.

Bezpečnost se výrazně zvyšuje, pokud je NIDS umístěn do vnitřní sítě tak, že analyzuje provoz procházející přes všechny vnitřní linky. Zapojení NIDS do vnitřní sítě ukazuje obrázek 3. NIDS je umístěn ke každému přepínači, směrovači a na hranici lokální sítě. V tomto zapojení už systém neanalyzuje provoz, který je blokován firewallem. Redukuje se tak výskyt falešných poplachů. Nicméně nevýhodou je nutnost použití velkého počtu zařízení v různých místech sítě, což znamená kromě velkých pořizovacích nákladů i značné úsilí vynaložené na udržení aktuální konfigurace pro všechna zařízení. Proto se umístění NIDS do vnitřní sítě používá pouze tam, kde je vyžadována velká bezpečnost síťové infrastruktury.



Obrázek 3: Zapojení NIDS do vnitřní sítě.

Syntaxe	Význam	Příklad
`^`	Vzor je hledán na začátku vstupních dat	Výraz `^AB` znamená, že `AB` se má hledat na začátku vstupních dat. Bez `^` se hledá `AB` kdekoliv ve vstupních datech.
`|`	Vazba nebo	Výraz `A|B` znamená `A` nebo `B`.
`.`	Libovolný znak	
`?`	Kvantifikátor pro jeden nebo žádný znak	Výraz `A?` znamená `A` nebo prázdný řetězec.
`*`	Kvantifikátor libovolného počtu znaků	Výraz `A*` znamená libovolný počet `A`.
`{ }`	Opakování	Výraz `A{100}` znamená 100 `A`.
`[ ]`	Třída znaků	Výraz `[lwf]` znamená znaky `l`,`w` nebo `f`.
`[^]`	Cokoliv s výjimkou	Výraz `[^\n]` znamená všechny znaky s výjimkou znaku `\n`.

Tabulka 1: Výrazové prostředky pro popis vzorů v programu Snort a L7 Dekodér.

5 Systém Snort

Pro detekci nebezpečného provozu se nejvíce používá volně šiřitelný program Snort. Díky velké podpoře komunity lidí, kteří se zabývají bezpečností počítačových sítí, obsahuje tento program rozsáhlou databázi pravidel popisující signatury útoku. Program Snort je implementován modulárně, což umožňuje využít pro detekci řadu různých modulů, které se také označují jako preprocesory. Mezi nejdůležitější preprocesory programu Snort patří rekonstrukce síťových toků, neboť síťová a transportní vrstva je fragmentovaná do jednotlivých paketů a signatury útoků se mohou nacházet na rozhraní dvou po sobě jdoucích paketů. V programu Snort jsou ale i preprocesory na sledování rozložení běžného provozu a detekci anomálií, což umožňuje odhalit i DoS útoky nebo skenování portů.

Každý známý útok je popsán signaturou, která se skládá z porovnání vybraných položek z hlaviček paketů až do úrovně síťové vrstvy ISO/OSI. Současně se analyzuje pomocí vyhledávání vzorů datová část paketů. Z těchto poznatků vychází i struktura pravidel systému Snort. Příklad zápisu jednoho konkrétního pravidla je uveden níže.

```
alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111
(content: "|00 01 86 a5|"; msg: "external mountd access";)
```

Je vidět, že každé pravidlo se skládá z hlavičky a těla. V hlavičce je nejprve uvedena akce, která se má vykonat pokud přijatý paket odpovídá pravidlu. Následuje porovnání konkrétních položek z hlaviček paketů. Jedná se o typ protokolu (tcp), zdrojovou IP adresu s maskou (!192.168.1.0/24), číslo zdrojového portu (any), cílová IP adresa s maskou (192.168.1.0/24) a cílový port (111). Na hlavičku navazuje tělo pravidla, které je uvedeno v kulatých závorkách. V těle pravidla jsou uvedená podrobná kritéria, která musí paket splňovat, aby se jednalo o útok. Je možné specifikovat vzory nebo regulární výrazy, které se mají hledat v datech paketů. Pro každý vzor je možné uvést, na jaké pozici v paketu se musí nacházet, maximální vzdálenost mezi vzory a další podobné informace. Možnosti popisu vzorů v programu Snort shrnuje tabulka 1. Kromě požadavků na výskyt vzorů nebo regulárních výrazů mohou být v těle pravidla uvedeny i požadavky na nastavení některých položek v TCP/IP hlavičce, jako je například pořadové číslo paketu nebo TCP flag.

Díky klasifikaci políček z hlavičky paketů je možné omezit analýzu obsahu paketu na vyhledání pouze takových vzorů, které mají pro daný paket smysl. Například pokud přijde paket s číslem portu 80 (webová služba), není potřeba hledat vzory určené pro protokol SMTP. S využitím klasifikace hlaviček paketů je tak omezen počet falešných poplachů.

Protokol	Regulární výraz
smtp	`^220[\x09-\x0d -~]* (e?smtp|simple mail)`
pop3	`^(\+ok |-err )`
ftp	`^220[\x09-\x0d -~]*ftp`
jabber	`<stream:stream[\x09-\x0d ][ -~]*[\x09-\x0d ]xmlns=['']jabber`

Tabulka 2: Příklady regulárních výrazů z databáze programu L7 dekodér.

6 L7 dekodér

Program L7 dekodér je jedním z nejznámějších volně dostupných programů, který slouží k detekci aplikačních protokolů. Primárně je program určen pro QoS systémy, ale je možné jej použít i pro blokování tunelovaného provozu. Detekce tunelovaného provozu je založena na vyhledávání regulárních výrazů v datech aplikační vrstvy. Současně s programem je volně k dispozici i databáze regulárních výrazů, které identifikují různé protokoly a používají stejné výrazové prostředky jako program Snort 1. Příklady regulárních výrazů pro běžně používané protokoly jsou uvedeny v tabulce 2.

V současně době jsou v databázi obsaženy řádově desítky různých protokolů. Jedná se jak o standardizované protokoly, tak i o konkrétní aplikační protokoly. Navíc k aplikačním protokolům existuje více regulárních výrazů, ze kterých může uživatel vybírat. Jednotlivé výrazy se liší zejména v přesnosti detekce a výpočetní náročnosti. L7 dekodér je implementován jako součást jádra operačního systému, ale i v podobě uživatelské aplikace. V obou případech dosahuje velmi malé propustnosti. I když se regulární výrazy hledají jenom v prvních paketech spojení, není možné program použít ani pro 100 Mb sítě, neboť samotní autoři programu uvádějí, že dosáhli maximální propustnost 20Mb/s, což je pro dnešní multi-gigabitové sítě nedostačující. Dalším problémem je, že L7 dekodér neřeší fragmentaci paketů na síťové a aplikační vrstvě. Vzory, které jsou rozděleny do více paketů, nejsou detekovány.

I přes uvedené nevýhody se regulární výrazy programu L7 dekodér často používají v IDS systémech pro detekci tunelovaného provozu, kdy útočník skryje svoji komunikaci pod nějakou povolenou službu. Detekci aplikačních protokolů na základě regulárních výrazů používá například IDS systém Bro [7].

7 Shrnutí

Každým rokem přibývají nové bezpečnostní hrozby a metody útočníků se vyvíjejí z jednoduchých do velice důmyslných a propracovaných forem. V současné době hojně rozšířené paketové filtry nejsou schopny odhalit některé typy útoků. Proto se v oblasti síťové bezpečnosti stále více uplatňují systémy NIDS. Pro NIDS existují dva základní přístupy realizace, které se liší v charakteru použitých detekčních metod. Jeden přístup je založený na hledání signatur, druhý přístup se snaží pomocí různých metod detekovat anomálie a odchylky od normálního síťového provozu. Nejvíce rozšířené a komerčně úspěšné se staly systémy, které využívají k detekci útoků signatury. Systémy založené na hledání anomálií jsou spíše předmětem výzkumu a počítá se s postupnou integrací jejich funkcí do stávajících systémů.

NIDS systém musí při detekci signatur vykonat s příchodem každého paketu celou řadu činností, které jsou výpočetně velmi náročné. Pokud má systém pracovat na multi-gigabitových rychlostech, stává se realizace NIDS velmi komplikovanou. Nejvíc výpočetně náročná je operace hledání regulárních výrazů, která zabírá více než 90 % výpočetního výkonu a limituje výkonnost NIDS na konvenčních procesorech řádově na stovky Mb/s. V zařízeních je proto snaha urychlit tyto časově kritické operace pomocí vhodného algoritmu nebo hardwarové architektury.

Za poslední rok bylo identifikováno více než 240 milionů nových nebezpečných programů, což je 100 % nárůst oproti roku 2008, a předpokládá se, že tento trend bude i nadále pokračovat. Pro nové algoritmy a hardwarové architektury tak není výzvou jenom dosažení vysoké rychlosti vyhledávání, ale dosažení vysoké propustnosti zejména pro rozsáhlé databáze signatur, kdy limitujícím faktorem může být rychlost a kapacita dostupných pamětí nebo množství hardwarových prostředků.

Reference

[1] Alfred V. Aho and Margaret J. Corasick. Efficient String Matching: An Aid to Bibliographic Search. *Communications of the ACM*, 18(6):333–340, 1975.

[2] Paul Barford, Jeffery Kline, David Plonka, and Amos Ron. A signal analysis of network traffic anomalies. In *IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 71–82, New York, NY, USA, 2002. ACM.

[3] Beate Commentz-Walter. A string matching algorithm fast on the average. In *Proceedings of the 6th Colloquium, on Automata, Languages and Programming*, pages 118–132, London, UK, 1979. Springer-Verlag.

[4] Sarang Dharmapurikar and Vern Paxson. Robust tcp stream reassembly in the presence of adversaries. In *SSYM'05: Proceedings of the 14th conference on USENIX Security Symposium*, pages 5–5, Berkeley, CA, USA, 2005. USENIX Association.

[5] Sarang Dharmapurikar, Haoyu Song, Jonathan Turner, and John Lockwood. Fast packet classification using bloom filters. In *ANCS '06: Proceedings of the 2006 ACM/IEEE symposium on Architecture for networking and communications systems*, pages 61–70, New York, NY, USA, 2006. ACM.

[6] J.E. Dickerson and J.A. Dickerson. Fuzzy network profiling for intrusion detection. In *Fuzzy Information Processing Society, 2000. NAFIPS. 19th International Conference of the North American*, pages 301–306, 2000.

[7] Holger Dreger, Anja Feldmann, Michael Mai, Vern Paxson, and Robin Sommer. Dynamic application-layer protocol analysis for network intrusion detection. In *USENIX-SS'06: Proceedings of the 15th conference on USENIX Security Symposium*, Berkeley, CA, USA, 2006. USENIX Association.

[8] Levent Ertöz, Eric Eilertson, Aleksandar Lazarevic, Ar Lazarevic, Pang ning Tan, Vipin Kumar, Paul Dokas, and Jaideep Srivastava. Minds - minnesota intrusion detection system. In *Next Generation Data Mining*, Boston, 2004. MIT Press.

[9] John E. Hopcroft and Jefferey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Adison-Wesley Publishing Company, Reading, Massachusets, USA, 1979.

[10] Seong Soo Kim and A. L. Narasimha Reddy. Statistical techniques for detecting traffic anomalies through packet header data. *IEEE/ACM Trans. Netw.*, 16(3):562–575, 2008.

[11] Petr Kobierský, Jan Kořenek, and Libor Polčák. Packet header analysis and field extraction for multigigabit networks. In *Proceedings of the 2009 IEEE Symphosium on Design and Diagnostics of Electronic Circuits and Systems*, pages 96–101. IEEE Computer Society, 2009.

[12] Eddie Kohler, Jinyang Li, Vern Paxson, and Scott Shenker. Observed structure of addresses in ip traffic. In *IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 253–266, New York, NY, USA, 2002. ACM.

[13] Jack Koziol. *Intrusion Detection with Snort*. Sams, Indianapolis, IN, USA, 2003.

[14] Jan Kořenek and Martin Košek. Flowcontext: Flexible platform for multigigabit stateful packet processing. In *2007 International Conference on Field Programmable Logic and Applications*, pages 804–807. IEEE Computer Society, 2007.

[15] Jan Kořenek and Viktor Puš. Memory optimization for packet classification algorithms. In *Proceedings of the 5th ACM/IEEE Symposium on Architectures for Networking and Communications Systems*, Association for Computing Machinery, pages 165–166. Association for Computing Machinery, 2009.

[16] Jan Kořenek and Viktor Puš. Memory optimization for packet classification algorithms in fpga. In *Proceedings of the 13th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems*, pages 297–300. IEEE Computer Society, 2010.

[17] Christopher Kruegel, Darren Mutz, William Robertson, and Fredrik Valeur. Bayesian event classification for intrusion detection. In *ACSAC '03: Proceedings of the 19th Annual Computer Security Applications Conference*, page 14, Washington, DC, USA, 2003. IEEE Computer Society.

[18] Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining anomalies using traffic feature distributions. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 217–228, New York, NY, USA, 2005. ACM.

[19] Inc. McAfee. McAfee IntruShield Network IPS Appliances datasheet. Dokument dostupný na http://www.mcafeesecurity.com, 2005.

[20] Jelena Mirkovic, Sven Dietrich, David Dittrich, and Peter Reiher. *Internet Denial of Service: Attack and Defense Mechanisms*. Prentice Hall PTR, 2005.

[21] Viktor Puš and Jan Korenek. Fast and scalable packet classification using perfect hash functions. In *FPGA '09: Proceeding of the ACM/SIGDA international symposium on Field programmable gate arrays*, pages 229–236, New York, NY, USA, 2009. ACM.

[22] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. *SIGMOD Rec.*, 29(2):427–438, 2000.

[23] Symantec Inc. Internet Security Threat Report, 2010.

[24] TippingPoint X505. http://h10144.www1.hp.com/products/security/index.htm, 2010.

[25] Alfonso Valdes and Keith Skinner. Adaptive, model-based monitoring for cyber attack detection. In *RAID '00: Proceedings of the Third International Workshop on Recent Advances in Intrusion Detection*, pages 80–92, London, UK, 2000. Springer-Verlag.