ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ

DIPLOMOVÁ PRÁCE

1999/2000

Marek Uher

Prohlášení

Prohlašuji, že jsem na své diplomové práci pracoval samostatně. Všechny materiály jiných autorů, které jsem během tvorby diplomové práce použil, jsou uvedeny v seznamu použité literatury. Zároveň souhlasím s využitím této práce a jejich výsledků katedrou počítačů ČVUT FEL v Praze. Používání, kopírování, modifikace a distribuce veškerého software a příslušné dokumentace autora diplomové práce, nebo jejich částí, se řídí Obecnou veřejnou licencí GNU (GNU GPL[1], viz. příloha A.3). Používání, kopírování, modifikace a distribuce veškerého software a příslušné dokumentace jiných autorů, nebo nebo jejich částí, se řídí licenčními ujednáními jejich autorů.

UNIX je zapsaná ochranná známka X/Open Company, Ltd. DEC, PDP, VAX, VMS jsou zapsané ochranné značky Digital Equipment Corp. / Compaq, Inc. Microsoft a Windows NT jsou zapsané ochranné známky Microsoft Corporation. Novell a NetWare jsou zapsané ochranné známky Novell, Inc. X Window system je zapsaná ochranná známka MIT. Názvy ostatních produktů a firem mohou být ochrannými známkami nebo zapsanými ochrannými známkami příslušných vlastníků.

Informace obsažené v této diplomové práci jsou zveřejněny bez ohledu na jejich případnou patentovou ochranu. Jména produktů byla použita bez záruky jejich volného použití. Autor nemůže převzít právní odpovědnost ani žádnou jinou záruku za použití údajů uvedených v této diplomové práci.

Obsah přiloženého CD je k dispozici tak, jak je. Autor diplomové práce nenese žádnou právní zodpovědnost za jeho použití, za funkčnost produktu, ani neposkytuje žádné záruky.

V Kladně dne 25. ledna 2000

Marek Uher

[1]GNU General Public License

Abstract

This diploma work deals with increasing the security of internal computer network.

First part of this work is a collection of tips how to secure Intranet and Unix distributed computing environment. The information security exists to protect company's assets, the integrity of the operating system(s), the accountability of users transactions, and the proprietary nature of the information contained and managed by and for all of the above. By using the auditing tools and configuring computer system with a bit of common sense, everyone should be able to prevent about 80 percent of security related problems.

The second part of this work describes architecture, building and implementation of the new auditing and monitoring system called Wotan. This system uses the standard Unix tools to prevent and identify some security problems. Wotan is primary designed for small and medium companies with limited financial resources (health care, hospitals, schools, ...).

The final part of work is turned to results of implementing and running Wotan on the Institute of Hematology and Blood Transfusion in Prague, and to the possibility of development of Wotan in the very near future.

Poděkování

Rád bych touto cestou poděkoval všem, kteří mi pomáhali s mou diplomovou prací. Především děkuji ekonomicko-technickému náměstkovi Ústavu hematologie a krevní transfuze panu Ing. Vladimíru M. Keřkovi, který mi poskytl podporu a prostředky pro realizaci mé diplomové práce.

Dále bych chtěl poděkovat Ing. Ivo Musilovi z firmy Corpus Solutions, a.s. za poskytnuté konzultace v otázkách zabezpečení interní sítě a bezpečnosti systému Unix. Jeho rady mi pomohly při ucelení znalostí nutných pro návrh monitorovacího a auditního systému.

Děkuji pracovnicím ze Střediska vědeckých lékařských informací Ústavu hematologie a krevní transfuze paní Ing. Marii Horníkové a paní Ing. Mileně Pohlové za poskytnutou literaturu a podnětné připomínky.

Zvláštní poděkování patří vedoucímu oddělení Správy počítačového informačního systému Ústavu hematologie a krevní transfuze panu Bc. Janu Vycudilíkovi za jeho rady spojené se samotnou realizací monitorovacího a auditního systému.

Nakonec bych chtěl poděkovat své mamince za její obětavost a velkou trpělivost, bez níž by tato práce nikdy nevznikla.

Obsah

Seznam tabulek

1 Úvod

V současné době jsme svědky prudkého rozvoje výpočetní techniky ve všech oborech lidské činnosti. Stejně tak dochází k rychlému vývoji v oblasti počítačových sítí. Na sklonku 60. letech vyvinula ARPA[2] strategickou síť založenou na počítačích, která by dokázala přečkat nukleární katastrofu nebo jiné vážné katastrofy. Ze sítí vojenského projektu Ministerstva obrany Spojených států DARPA[3] se v 80. letech vyvinunula celosvětová akademická síť nazvaná Internet. V 90. letech vývoj Internetu pokračoval a došlo k jeho rozšíření v komerční a soukromé sféře. V současné době je Internet především nástrojem pro komunikaci, získávání informací a v neposlední řadě také pro elektronického obchodování.

Rychlý rozvoj Internetu v posledních letech přinesl s sebou i nové bezpečnostní problémy. Přesto se ale určitá část trhu vyvíjela ještě rychleji než Internet. Společnosti začaly využívat nové technologie používané na Internetu pro interní použití. Vznikl tak nový typ sítě – Intranet. Tento typ sítě používá stejně jako Internet přepojování paketů TCP/IP[4], publikování dokumentů v elektronickém formátu a jiné technologie vyvinuté na Internetu. V podstatě se jedná o síť typu LAN, ke které ale mohou lidé s příslušným oprávněním a přístupovými právy přistupovat z libovolného místa na světě. Bezpečnostní otázky vzniklé spolu s Intranetem jsou ještě naléhavější, než ty, které vznikají kolem sítě Internet.

Tato práce se zabývá problematikou monitorování Intranetu vystavěného na bázi unixových počítačů a detekce možných bezpečnostních problémů, které mohou nastat při jeho používání. Hlavním cílem práce však je vytvoření monitorovacího a auditního systému sítě unixových počítačů určeného pro malé a střední organizace s omezenými finančními prostředky (zdravotnictví,

[2] Advanced Research Project Agency

[3] Defenese Advanced Research Project Agency

[4] Transmission Control Protocol / Internet Protocol

školství, malé podnikové subjekty).

Diplomová práce je rozčleněna na kapitoly, které na sebe plynule navazují. Rozbor problematiky zabezpečení Intranetu a distribuovaného unixového prostředí provedený v druhé kapitole je dobrým předpokladem pro pojednání o principech auditu a monitorování unixových systémů, které jsou popsány v kapitole následující. Na tuto část navazuje popis zvoleného řešení, které bylo v rámci diplomové práce implementováno a zároveň slouží jako uživatelský návod k vytvořenému auditnímu nástroji. Další kapitola se věnuje implementaci a stručně popisuje zdrojové kódy nástroje. Jsou zde také k dispozici informace o testování programu v praxi. V závěru jsou shrnty dosažené výsledky a naznačeny možnosti dalšího rozšiřování vytvořeného nástroje.

Podklady pro svoji práci jsem čerpal z různých zdrojů. Kromě tištěných materiálů a literatury jsem využíval také informace v elektronické podobě dostupné na Internetu a v elektronických konferencích.

2 Počítačové sítě a OS Unix

V roce 1965 spojily BTL (Bell Telephone Lab) firmy AT&T své úsilí se společností General Electric Company a MIT (Massachusetts Institute of Technology) na vývoji nového operačního systému, označeného *Multics* (Multiplexed Information and Computing Sevice), [Bach86], [Brod89] a [Salu94]. Cílem bylo poskytnout široké skupině uživatelů simultánní počítačový přístup, mohutný výpočetní výkon a efektivní uložení dat.

Tento projekt skončil nezdarem. Přestože jednoduchá verze Multicsu byla provozována na počítači GE 645 od začátku roku 1969, neposkytovala ty výpočetní služby, pro které byla vyvíjena, a ani nebylo jasno, zda bude vývojových cílů dosaženo. Proto BTL svou spoluúčast na projektu ukončily [Bach86].

Po ukončení prací na projektu Multics zůstali však členové Výzkumného výpočetního centra, spadajícího do Bellových laboratoří, bez vhodného interaktivního výpočetního systému. Proto na jaře roku 1969 Ken Thompson navrhnul jednoduchý multiuživatelský systém správy souborů. Společně s Denisem Ritchiem a Ruddy Canadayem propracovali tento návrh až do rané verze jednouživatelského operačního systému. Později dostal tento operační systém název *UNIX* [Salu94].

Vznik Unixu je jedna z největších počítačových událostí. Demonstruje, že relativně malý operační systém může pracovat na různých hardwarových platformách. Unix jako první ukázal, že operační systém může být přenosný, dostupný a nezávislý na použitém hardwaru. Unix nebyl vyvinut výrobcem nabízejícím počítačové vybavení, vzniknul z požadavků individuálních uživatelů požadujících jednoduchý víceuživatelský systém, který by mohl sloužit pro každodení práci. Jednoduchost a přehlednost systému Unix také sehrála velkou roli při rozvoji počítačových sítí a komunikačních technologií jako takových.

2.1 Vývoj síťového vybavení Unixu

Vývoj operačního systénu Unix se od začátku řídil několika základními principy. Pro programové vybavení vyvinuté pro Unix platí následující pravidla:

- programy budou dělat právě jednu činnost, a tu musí dělat dobře
- programy musí být schopné navzájem spolupracovat
- programy musí být schopné pracovat s textovými kanály

Tyto principy se staly základem nového univerzálního rozhraní Unixu. Později se tyto principy uplatnily i při návrhu a tvorbě síťového vybavení [Stev90], [Salu94].

První síťovou aplikací Unixu byl `uucp` (Unix–to–Unix copy), kopírovací program mezi dvěma systémy typu Unix a všechny jeho přidružené příkazy. `uucp` byl vyvinut kolem roku 1976 a jeho první verze vydaná mimo AT&T byla součástí Unix Version 7 v roce 1978. `uucp` je dávkově orientovaný systém, používaný mezi systémy propojenými telefonní linkou nebo mezi systémy, které jsou propojeny přímo. Používal se především pro distribuci software (přenos souborů) a elektronickou poštu.

Spolu s Unix Version 7 se distribuoval i program `cu` a od té doby byl k dispozici spolu s téměř každou distribucí Unixu. Přestože tento program není nedílnou součástí síťového vybavení, nabízí možnost vzdáleného přihlašování na jiný počítačový systém. Program `cu` se obvykle používal pro spojení s jiným systémem pomocí telefoní linky. Upravená verze `cu` byla distribuována s 4.2BSD, kde byla ale přejmenována na `tip`.

V roce 1978 vyvinul Eric Schmidt síťovou aplikaci pro Unix jako součást své disertační práce na univerzitě v Berkeley. Tato jednoduchá síť se nazývala `Berknet` a byla poprvé distribuována s Berkeley Unix Version 7 pro platformu PDP-11 v distribuci 2.xBSD.

Nabízela přenos souborů, elektronickou poštu a vzdálený tisk. Byla využívána zejména na univerzitě v Berkeley. Jednotlivé počítače byly přímo propojeny linkami RS-232 s přenosovou rychlostí 9600 bps. V pozdějších verzích 4.xBSD se tento software přenesl na platformu VAX a následně byl postupně nahrazen rychlejší lokální sítí `Ethernet`, která se poprvé objevila spolu s verzí 4.2BSD.

V září roku 1980 byli Bolt, Beranek a Newman (BBN) smluvně zavázáni agenturou DARPA z ministerstva obrany USA (DoD[5]), aby vyvinuli implementaci TCP/IP protokolu pro Berkeley Unix na počítačích VAX. Verze pro systém 4.1aBSD byla připravena na podzim roku 1981. Tato verze obsahovala zkušební verze TCP/IP a `sockets` pro uzly ARPANET. Poté bylo síťové vybavení integrováno do systému 4.2BSD, který byl právě v té době ve vývoji. Systém 4.2BSD byl poprvé zveřejněn v srpnu 1983 a výrazně podpořil další rozvoj lokálních sítí, založených hlavně na technologii `Ethernet`. Systém 4.2BSD umožnil rovněž spojení počítačů na síti ARPANET, která zaznamenala velký rozvoj na počátku 80. let. V červnu roku 1986 byla do distribuce 4.3BSD zahrnuta síťová výbava *XNS* (Xerox Network System) a Internet name server. Tato verze měla přímý vliv na vznik Internetu tak, jak ho známe dnes. Další verze distribuce BSD přinášely vylepšení a podporu pro práci v síti. Verze Net–1 BSD z listopadu 1988 obsahovala pomocné programy pro vzdálený přístup. Verze 4.3-Reno BSD obsahuje implementaci *NFS* (Network Filesystem) a podporu referenčního síťového modelu OSI[6] [Stev90].

Vývoj sítí byl v té době v plném proudu i v AT&T, ovšem ubíral se jiným směrem než u BSD. Výsledky se ale bohužel obvykle nedostaly mimo AT&T. To byl velký rozdíl od prací v Berkeley, jejichž výsledky se naopak velmi rychle rozšiřovaly. Nejznámější síťový projekt v AT&T je vysokorychlostní

[5] Department of Defense

[6] Open System Interconnection

lokální síť založená na kanálu *HYPER*. Její vývoj začal kolem roku 1980 a je spojován s různými typy systémů, které provozují různé verze AT&T Unixu: VAX 780, PDP-11/70, IBM 370 a AT&T 3B20S. Experimentální verze použitého Unixu odpovídají verzím System III a System V od AT&T. Sítě byly dávkově orientované a podporovaly přenos souborů, vzdálené provádění příkazů, vzdálený tisk a elektronickou poštu.

V polovině 80. let se na trhu s AT&T Unixem objevily různé síťové balíky třetích stran. Většinou podporovaly protokol TCP/IP a byly vyvinuty výrobcem, který vyráběl propojovací hardware.

Posledním pokusem o vytvoření univerzálního rozhraní, které by umožňovalo komunikaci i prostřednictvím počítačových sítí, který provedla firma AT&T, byla implementace vstupně/výstupních proudů (I/O streams). Zpočátku nebyly zahrnovány do distribucí určených mimo vývojové prostředí Bell Labs až do verze AT&T System V Release 3.0 z roku 1986. Rozhraní transportní vrstvy *TLI* (Transport Layer Interface) bylo dodáno poprvé právě s touto verzí System V. Kolem roku 1988 se rovněž objevily různé síťové balíky nabízené třetími stranami založenými na principu TLI (především pro TCP/IP protokol). Nezaznamenaly však většího rozšíření.

2.2 Celosvětová síť Internet

Internet je soustava vzájemně propojených počítačových sítí, které jsou založeny na společné filozofii a stejných protokolech. Stupeň jejich propojení a možnosti jejich vzájemné součinnosti jsou však takové, že běžný uživatel může výsledný celek skutečně považovat za jednu jedinou síť.

Integrujícím faktorem všech sítí, sdružených v Internetu, je soustava protokolů, označovaných jako TCP/IP. Jak již bylo zmíněno výše, práce na jejich vývoji byly zahájeny kolem roku 1970 na předních výzkumných pracovištích v USA.

Práce na projektu byly sponzorovány účelovou agenturou DARPA amerického ministerstva obrany DoD. Pod dohledem DARPA byly vytvořeny první návrhy a zkušební implementace protokolů TCP/IP. Svou dnešní podobu získaly protokoly TCP/IP zhruba v letech 1977–79. V té době již síť ARPANET existovala a jejím provozovatelem byla právě agentura DARPA. Ta přibližně kolem roku 1980 začala převádět síť ARPANET na nový protokoly TCP/IP a připojovala k nově vzniklé síti další vědeckovýzkumné sítě. Síť ARPANET se tak rychle stala zárodkem a současně i páteřní sítí celé soustavy sítí – vznikajícího Internetu.

Přechod na soustavu protokolů TCP/IP byl završen v roce 1983, kdy byla ze sítě ARPANET vyčleněna síť MILNET, sloužící čistě vojenským účelům armády Spojených států. Vlastní ARPANET a k ní připojené sítě si zachovaly charakter sítí, které slouží především pro vědeckovýzkumné účely [Stev90].

Na základní síť ARPANET se následně připojovalo stále více sítí, až se prakticky přestalo hovořit o síti ARPANET a převládlo označení Internet. Internet se začal exponenciálně rozrůstat – postupně se k němu začaly připojovat nejen počítačové sítě v USA, ale i další sítě po celém světě. Internet se přetvořil do podoby, v jaké ho známe dnes.

2.3 Lokální sítě a Intranet

Internet je fenomén, který ovládá nejen počítačový průmysl, ale stále více se dotýká každého z nás, ať již pracovně nebo v soukromém životě. Proto, aby každá společnost mohla využít obrovský potenciál této sítě a technologie, je důležité si uvědomit fakt, že Internet je sítí veřejnou. To přináší řadu výhod, od možnosti výběru nejvhodnějšího lokálního poskytovatele, přes možnost efektivního propojení sítě poboček až po snadnou komunikaci se zahraničním partnerem. Zajímavá je také vysoká dostupnost veřejných dat na Internetu a naopak nízké náklady na publikování vlastních informací.

Tato otevřenost Internetu však současně znamená, že pro připojení se k němu je nezbytné zavést mechanismy ochrany vnitřní sítě – Intranetu – a zajistit případné utajení přenášených dat a soukromých informací uložených v interních databázích.

Každá instituce nebo firma, která se rozhodne v jakékoliv míře využívat služeb sítě Internet či Intranet, je postavena před otázku síťové bezpečnosti. To, jakým způsobem se s ní vypořádá, do velké míry ovlivní přínosy a případné škody způsobené podceněním rizik, která sebou tyto nové technologie přinášejí. Využití sítí Internet a Intranet, ale také množství jiných datových zdrojů, je nezbytné řešit komplexně tak, aby interní síť byla maximálně zabezpečena. Otázka bezpečnosti patří při propojování sítí mezi klíčové, a je proto nezbytné jí věnovat maximální pozornost.

Pro realizaci bezpečnostních opatření je nutné nejprve porozumět rozdílu mezi lokální sítí a Intranetem. LAN[7] je ve většině případů realizována určitým počtem stolních počítačů připojených k jednomu nebo více serverům. Koncové počítače používají server v prvé řadě k přístupu ke společným či sdíleným datům. Každý počítač je však stále jedinečný a samostatný. Jediným způsobem, jakým mohou tyto počítače při tomto uspořádání známém jako *klient-server* komunikovat, je prostřednictvím elektronické pošty nebo sdílením souborů.

Uzavřený Intranet naproti tomu uživatelům umožní používat technologie vyvinuté na Internetu ke komunikaci v rámci jedné společnosti. Uživatelé s možností vzdáleného přístupu k Intranetu mohou navíc pracovat se stejnými programy, které používají v práci, i doma. Protože uživatelé mají snadný přístup k technologiím Internetu, nabízí Intranet nenákladný způsob komunikace mezi zaměstnanci firmy. Jiným příkladem může být i možnost komunikace mezi zaměstnaci a zákazníky společnosti.

[7] Local Area Network

Další možností je použití Intranetu pro uveřejnění formulářů a informací o zdravotním stavu pacientů hospitalizovaných ve zdravotnických zařízení, k rozesílání informací o cenách svým prodejcům či dodání upgrade softwaru registrovaným uživatelům a podobně.

V současné době se čím dál více žádoucí sdílet počítače z důvodu společného využívání zdrojů způsobem, při kterém si každý počítač uchová nadále autonomii svého prostředí. Uživatel osobního počítače chce mít např. přístup k souborům uloženým na centrálně spravovaném serveru, ale chce si zachovat autonomní řízení svého osobního počítače. Již dnes řada moderních programů dovoluje přenos, údržbu a synchronizaci souborů a poskytuje i další síťové služby. Jejich použití není ale transparentní, protože uživatel si existenci sítě stále uvědomuje. Navíc, programy typu textových editorů nepracují se vzdálenými soubory stejně jako s lokálními. Uživatelé však chtějí využívat veškeré možnosti, které jim síť nabízí, na druhé straně nechtějí vědět o tom, že překračují hranice svého počítače. Tyto požadavky uživatelů může uspokojit pouze skutečné síťové distribuované výpočetní prostředí [Sun99]. Tím jsou požadavky kladené na Intranet ještě více odlišné, než požadavky kladené na Internet nebo jednoduché sítě typu LAN [Klan97].

2.4 Koncepce síťové bezpečnosti

Bezpečnost počítačových sítí je komplexním problémem. Je zřejmé, že tato problematika nemůže být řešena zcela izolovaně od ostatních oblastí bezpečnosti, jako jsou zabezpečení přístupu vnitřních uživatelů k interním serverům, správa účtů lokálních uživatelů, monitorování podezřelých událostí v interní síti nebo administrace a správa systému výpočetní techniky. Na druhou stranu můžeme říci, že vzájemná ochrana sítí by měla být navržena tak kvalitně, aby byla nezávislá na technologiích použitých uvnitř jednotlivých sítí.

Komunikace mezi těmito sítěmi by měla být zcela řízena ochrannými prvky, které tak izolují potencionální bezpečnostní problémy v jedné z připojených sítí od sítí ostatních.

V dobách, kdy nebyly počítače zapojeny do sítě, se bezpečnost zajišťovala fyzickými prostředky. K fyzickým kontrolním prostředkům patří zámky na dveřích, dozorčí u vchodu, záložní kopie důležitého softwaru a dat a fyzické prostory s výpočetní technikou volené s ohledem na minimalizaci škod způsobených případnými živelnými pohromami. Některá opatření spadající mezi výše uvedené najdou uplatnění i dnes. Přesto jsou někdy jednoduché fyzické kontroly opomíjeny, zatímco náročnější a dražší přístupy jsou propagovány, i když často přinesou z hlediska bezpečnosti znatelně menší efekt než jednoduchá fyzická opatření.

Cílem moderní informační bezpečnosti je zavádění kontrolních opatření zajišťujících *diskrétnost* (utajení), *integritu* (celistvost) a *dostupnost* (použitelnost). Mezi kontrolní prostředky lze zařadit šifrování, kontroly softwaru (vývojové kontroly, kontroly operačního systému, kontroly spustitelných programů, ...), kontroly hardware a konečně administrativní, legislativní a etické kontroly. Počítače připojené k síti, zejména pak k Internetu či rozsáhlému Intranetu, jsou vystaveny většímu ohrožení bezpečnosti, než systémy izolované. Bezpečnost sítě pak slouží ke snížení rizik souvisejících právě s připojením počítačů do sítě. Tím ale vzniká dilema – ze své podstaty stojí připojení k síti a počítačová bezpečnost obecně proti sobě. Funkcí sítě je zvýšení možnosti přístupu k počítačovým systémům, zatímco bezpečnost se snaží přístup omezit. Z toho vyplývá, že zajištění bezpečnosti síťového výpočetního prostředí představuje kompromis mezi otevřeným přístupem a absolutní bezpečností [Prib96].

Každé realizaci zabezpečení sítě musí předcházet návrh bezpečnostní koncepce. Jedná se o komplexní problematiku, kdy zanedbání jednoho prvku může zmařit vynaložené náklady a současně ohrozit cenná data.

Stejně jako chybná konfigurace nákladného ochranného systému je i volba nevyzrálého operačního systému na kritické pozici ohrožením bezpečnosti interní sítě [Kirc98].

2.5 Základní bezpečnostní principy

Při návrhu komplexních bezpečnostních opatření je velmi důležité si dopředu rozmyslet, co a jak bude chráněno. Je také potřeba si uvědomit základní bezpečnostní principy. Je možno také vycházet z některých formálních kritérií bezpečnosti (např. ITSEC, TCSEC, ITSEM a další, viz. [DoD83] a [Prib96]), ale tato kritéria vycházejí z názoru, že bezpečnost produktu, resp. systému, je měřitelná počtem a rozsahem ochranných mechanizmů zavedených během návrhu a vývoje. V praxi však bezpečnost není měřitelná pouze existencí nebo počtem bezpečnostních mechanizmů, ale spíše jejich správným nasazením a používáním. Je proto daleko lepší se při návrhu bezpečnostních opatření týkajících se počítačové sítě či systému řídit principy, která prověřila praxe. V následujících odstavcích se pokusím shrnout základní bezpečnostní principy.

- Co není povolené, je zakázané

Tento princip (v anglofonní literatuře [Sec97] označovaný jako *"default to deny principle"*) je základním principem při návrhu a tvorbě rozsáhlých kombinací přístupových práv většího počtu oprávněných uživatelů systému. Tento princip zajišťuje již v samotném základě vysokou bezpečnost navrhovaných bezpečnostních opatření. V případě omylu jsou totiž oprávněnému uživateli obvykle přidělena menší privilegia, než jaká potřebuje pro svou práci. Dá se očekávat, že takový omyl bude poměrně rychle objeven a napraven. Důsledkem tohoto omylu bude omezení činnosti uživatele, protože uživatel nebude schopen využívat služby systému v předpokládaném rozsahu.

- Nejmenší privilegia

Princip (v anglofonní literatuře [Sec97] označovaný jako *"least privileges principle"*), který stanovuje následující bezpečnostní pravidlo: "oprávněnému uživateli systému mají být přidělena jen taková privilegia, jaká nutně potřebuje pro svou činnost, a ne větší". Hlavním důvodem zavádění tohoto pravidla je minimalizace možné škody, a to jak zaviněné chybou nebo omylem oprávněného uživatele systému, tak i v případě jeho pokusu o zneužití přidělených oprávnění.

- Obecné bezpečnostní principy

Do této kategorie spadají ostatní bezpečnostní principy. Bližší specifikace těchto principů již přímo nesouvisí s mojí diplomovou prací. Případné zájemce mohu odkázat na literaturu [DoD83], [Prib96] a [Sec98]. Do této skupiny například náleží: rozdělení povinností (výkonná, kontrolní a schvalovací funkce je rozdělena mezi více subjektů), zachování bezpečnosti systému při havárii a při obnovení jeho provozu, vyloučení existence slabého bodu (nutnost odstranění jediné, nezálohovatelné anebo jinak nechráněné komponenty systému, na níž v plné míře závisí správný a bezporuchový chod systému), zajištění pravidelné revize bezpečnostních principů a pravidel.

Druhým a patrně nejdůležitějším krokem při návrhu zabezpečení sítě je vytvoření síťové bezpečnostní politiky. Tato činnost patří k nejméně oblíbeným krokům v celém postupu návrhu zabezpečení. Téměř všichni pracovníci z počítačové komunity se vždy snaží nalézt pro každý problém technické řešení. V tomto případě je ale potřeba sestavit dohromady vysvětlující text na úrovni znalostí běžného pracovníka o bezpečnostní politice a postupech. Je ale třeba vzít v úvahu, že dobře promyšlený bezpečnostní plán usnadní rozhodování o tom, co se má chránit, kolik se má do ochrany investovat a kdo bude za jednotlivé ochranné postupy zodpovědný.

2.6 Typy útočníků

V současné době vznikají nové studie (viz. [Prib96], [Chap95] a [Lhot96]), které studují vlastnosti lidí páchajících počítačovou trestnou činnost. Tyto studie mají napomoci při typování potenciálních pachatelů a preventivně potlačovat kriminalitu.

V následujícím výčtu uvedu pouze některé typy lidí páchajících počítačovou trestnou činnost:

- amatéři – lidé využívající náhodně objevenou bezpečnostní slabinu informačního systému. Využívají napadené počítače pro osobní zájmy
- hakeři, krakeři a studenti – obvykle středoškolští nebo vysokoškolští studenti, kteří se snaží získat přístup k zabezpečeným výpočetním prostředkům
- joyriders – nudící se lidé, kteří hledají zábavu. Většinou napadají informační systémy ze zvědavosti
- vandalové – lidé, které baví ničit věci, nebo lidé, kteří nemají v lásce vlastníky napadeného systému (typicky propuštěný zaměstnanec)
- score keepers – lidé napadající systém z důvodu zvýšení si vlastního sebevědomí. Snaží se napadat známé nebo výjimečné, dobře zajištěné systémy
- profesionálové – lidé, kteří jdou systematicky za svým cílem. Nejčastěji do této kategorie spadají tzv. průmysloví špioni

2.7 Typy útoků

Stejně, jako existuje mnoho typů útočníků, existuje i velké množství typů útoků. Útočník může napadnout hardware (týká se obvykle vlastního výpočetního centra – krádež, ničení počítačů střelbou, nebo ostrými předměty,

ničení počítačových sálů výbušninami, ohněm, atd.), software (úmyslná modifikace, mazání, chybná instalace, krádež,...) a zejména data (zcizení, zničení a modifikace).

Mezi nejčastější druhy útoků patří:

- vniknutí – vetřelec využívá systém jako legitimní uživatel
- znemožnění či odepření služby – jeden z možných způsobů zaměřený na zabránění v běžném využívání počítače
- krádež informace – získání důvěrných dat bez přímého použití chráněného počítače (nejčastěji odposlechem síťové komunikace)

2.8 Bezpečnost systému Unix

V předchozích odstavcích jsem obecně rozebral základní bezpečnostní principy a zranitelná místa v počítačových sítích. V této kapitole se zaměřím na bezpečnostní problémy operačního systému Unix.

Přesto, že se nedávno projevil zájem o malé a střední servery pro Internet a Intranet na platformách Microsoft Windows NT a Novell NetWare, zůstává primárním operačním systémem, který dnes řídí většinu serverů, operační systém Unix. Unix byl prvním operačním systémem používaným na Internetu a svoje vedoucí místo si zachovává i dnes. Proniká i do interních sítí malých a středních firem, které na jeho základě budují svůj Intranet [Sun99], [Corp99].

Stabilita a trvalost Unixu je jeho velkou výhodou. Bohužel, někdy naopak s sebou přináší i některé problémy. Díky tomu, že Unix je dostupný na trhu s počítačovými systémy více jak 30 let, existuje velké množství pomocných programů, které usnadňují ochranu a zabezpečení serveru běžícího pod operačním systémem Unix. Na druhou stranu jsou k dispozici stovky programů pomáhající narušitelům tyto ochrany Unixu zdolat [Hunt92], [Ches95]. Proto se v následující části textu budu podrobně věnovat nejčastějším formám útoků na známá slabá místa v bezpečnosti Unixu.

V dalším textu se předpokládá, že čtenář je seznámen se základními principy a běžnými příkazy operačního systému Unix. V textu nebude podrobně rozebírána běžná činnost administrátora systému. Z tohoto důvodu se u čtenáře také přepokládají znalosti nutné pro instalaci operačního systému typu Unix, včetně aktualizačních modifikací. Dále by měl čtenář znát a ovládat zaváděcí a ukončovací postupy systému. Též se předpokládá znalost administrace uživatelů, souborových povolení, správy souborového systému, speciálních zařízení a dalších částí systému. Potřebné znalosti nalezne případný zájemce v literatuře [Bach86], [Brod89], [Hunt92], [Lhot96], [Satr98] a [Skoc93].

2.9 Útoky na zranitelná místa v Unixu

Jakémukoliv pokusu o útok nebo průniku do informačního systému předchází výběr cíle nebo cílů, které později určují směr útoku. Při průniku do systémů Unix existuje několik nejznámějších zranitelných míst, o jejichž napadení se bude narušitel pokoušet. Tato zranitelná místa lze podle klesajícího rizika seřadit následujícím způsobem:

1. *Heslo správce systému* `root`. Heslo privilegovaného uživatele `root` je finálním cílem útoku narušitele. Získání hesla tohoto uživatele dovoluje narušiteli zcela ovládat napadený systém[8] a zcela mu umožňuje přístup ke všem jeho částem. Jediný rozdíl mezi správcem systému a úspěšným narušitelem je, že správce systému má ke stroji fyzický přístup.

2. *Přístup k účtu správce systému* `root`. Narušitel má přístup k celému systému, ale nemůže se odhlásit, protože nezná heslo a proto nemá možnost se do systému vrátit.

[8] Některé unixové systémy se dokáží účinně bránit i při tomto druhu napadení, např. NetBSD. Tento systém zavádí tzv. bezpečnostní úrovně. Tyto úrovně mimo jiné omezují pravomoce uživatele root.

3. *Heslo oprávněného uživatele systému.* Narušitel zná heslo některého oprávněného uživatele systému, které mu umožňuje přístup do systému.

4. *Přístup k účtu oprávněného uživatele systému.* Narušitel má přístup k účtu oprávněného uživatele systému, ale nemůže se do systému znovu vrátit, protože nezná heslo uživatele.

5. *Heslo nějakého uživatele systému.* Narušitel získal náhodně heslo nějakého uživatele (např. odposlechem na síti), nezná však uživatelův účet.

6. *Útoky na dostupnost služeb.* I bez úplného přístupu oprávněného uživatele systému může značná část narušitelů odstraňovat soubory, způsobit havárii systému nebo může zapříčinit mnohé další závažné škody v systému.

V další části textu některé formy útoku rozeberu detailněji a zmíním se o možných způsobech obrany.

2.9.1 Zcizení hesla oprávněného uživatele

Pro získání hesla oprávněného uživatele systému musí mít narušitel přístup k počítači, síti nebo alespoň musí být schopen využít chyby v nějaké službě. Nejjednodušší cesta do počítače obvykle vede přes lokální či vzdálený příkaz *login*. Na všech moderních systémech je potřebné k úspěšnému přihlášení potřebné zadání správného hesla v předem definovaném počtu pokusů. Některé dřívější přihlašovací programy dovolovaly nekonečné množství chybných pokusů o přihlášení. Programy pro hádání hesel byly proto omezeny pouze rychlostí modemů. Navíc neexistovalo žádné protokolování událostí v systému.

Historie obecného přihlašovacího programu je vlastně opakující se řada útoků a obran. První verze Unixu udržovaly hesla v souborech v nezašifrované formě.

Bezpečnost takového systému byla založena na utajení jména souboru s hesly. Tento soubor byl ale jinak čitelný pro každého, kdo znal jeho jméno. V nových distribucích systému Unix byla hesla ukládána zašifrovaná (používá se 8 bajtový jednosměrný hash), ale nadále čitelná pro každého. Tyto nová bezpečnostní opatření šlo obejít při použití útoku hrubou silou, nebo použitím off-line slovníkového útoku. Tyto útoky potřebovaly více procesorového času, ale rozkódování mohlo probíhat v úplně jiné části světa, než ve které byl ohrožený počítač.

Následně byl zaveden v Unixu stínový soubor s hesly a další programy pro vylepšení bezpečnosti hesel. Tato opatření přinesla určité zvýšení zabezpečení systému. Bohužel současné zavedení nových síťových služeb na bázi protokolu TCP/IP přineslo nové požadavky a svá vlastní rizika. Služby jako `FTP`[9] požadují stejnou autentizaci, ale používají oddělenou rutinu, než kterou používá standardní příkaz *login*. Tím může být léty prověřený kód z programu *login* nahrazen napodobeným a neprověřeným kódem `FTP` a do systému jsou zavlečeny další chyby. Dnes existuje velké množství odposlouchávacích programů, které mohou zachytit nezašifrované heslo z několika prvních paketů síťové relace. Existují též knihovny, které lze rychle nainstalovat do napadeného systému a nahradit nimi například standardní knihovny systémových démonů. Tyto knihovny zaznamenávají následně hesla v otevřeném tvaru na určité místo, kde si je později narušitel bez větší námahy vyzvedne.

Bezprostředním cílem mnoha útoků není přímý průlom, ale snaha o získání souboru s hesly. Mezi služby nebo programy, které byly zneužity k získání souboru s hesly, můžeme například zahrnout `FTP`, `TFTP`[10], `sendmail`, `NIS`[11], `rsh`[12], `finger`, `uucp`, `X-Window system` a mnohé další. Pro narušitele systému je útok jednoduchý – pokud je správce systému neopatrný nebo

[9] File Transport Protocol
[10] Trivial File Transport Protocol
[11] Network Information System
[12] remote shell

udělá chybu při výběru distribuce systému Unix. Obranná opatření zahrnují velkou opatrnost a konzervativní přístup k administraci softwaru. Narušiteli totiž stačí vyhrát jen jednou.

Většině popsaných bezpečnostních problémům spojenými s uživatelskými hesly a účty se dá předejít použitím kvalitních hesel. Základní pravidlo pro tvorbu hesla je jednoduché – správné heslo by nemělo obsahovat jakékoliv racionálně zjistitelné informace (např. jméno manželky, telefonní číslo, rodné číslo, části uživatelova jména, oblíbenou barvu, ...). Ideální heslo je náhodné seskupení znaků, které dává smysl pouze svému vlastníkovi.

Pro tvorbu bezpečných hesel existuje následující postup [Satr98]: Základem budoucího generovaného hesla je libovolná krátká věta. Například dvojverší z básně, název filmu a podobně. Heslo se sestavuje z prvních písmen jednotlivých slov ve větě. Například z věty "Zelené křepelky letěly Zelenou ulicí 13krát" vznikne slovo "ZklZu13k". Vzniklé heslo je náhodným seskupením znaků (malých a velkých písmen a číslic), které si uživatel ale snadno zapamatuje. Ještě bezpečnější heslo vznikne, pokud základní věta obsahuje i interpunkční znaménka.

2.9.2 Chyby v softwarovém vybavení

Nejznámější chybou v softwarovém vybavení Unixu je "Internetový červ". Dne 2. září 1988 okolo 18 hodiny postgraduální student Robbert T. Morris z Cornellovy University vypustil síťový program, který nazval *Worm*. Tento program během 24 hodin vyřadil z provozu většinu (téměř 6000) unixových počítačů hlavních národních a mezinárodních výzkumných ústavů. Trvalo další týden, než byly všechny zasažené systémy opět provozuschopné a v chodu. V lednu 1990 byl Morris postaven před soud a 4. května téhož roku odsouzen ke třem letům dozoru, 400 hodinám veřejných prací a pokutě 10 000 dolarů, [Stol90].

Jedna z cest, kterou se rozšiřoval Internetový červ, bylo zasílání nového kódu démona `finger`. Démon takovouto událost neočekával a v definovaném protokolu takováto možnost nebyla. Démon `finger` použil knihovní funkci `gets`, která neudává maximální délku vyrovnávací paměti. Internetový červ přeplnil vstupní buffer a pokračoval tak dlouho, až přepsal návratovou adresu v aktivačním záznamu funkce `gets`. Když podprogram skončil, provedl se návratový skok na adresu tohoto bufferu a začal se vykonávat v něm uložený kód. Následky této činnosti jsou popsány výše.

Tato chyba, umožňující průlom do systému a jí podobné, byly většinou distributorů Unixu již dávno opraveny. Obecný problém ale stále přetrvává - tvorba bezchybného software je problémem, který v současné době počítačová věda nedokáže řešit. Chyby jsou částečně těžce modelovatelné, protože z teoretické definice nelze zjistit, který z předpokladů, pokud nějaký, je chybný či nikoliv. Nelze sestrojit teoretický model bezchybného softwaru a tudíž nelze prokázat, zda daný software neobsahuje chyby.

V případě Internetového červa by např. většina formálních ochran Oranžové knihy [DoD83] byla naprosto k ničemu. Obecně by vysoce klasifikovaný systém omezily průlom do jedné bezpečnostní úrovně. Ale ve skutečnosti byl Internetový červ útokem na dostupnost služby. V takovém případě je ale celkem jedno, zda je útok na počítač s víceúrovňovým modelem zabezpečení proveden neklasifikovaným nebo přísně tajným procesem. Výsledek je ve všech případech stejný – systém se stáva zcela nepoužitelným.

Dobré postupy při návrhu softwaru jsou nařizovány a vynucovány vyhodnocovací agenturou. Nicméně, i nejlépe navržené systémy mají chyby. Internetový červ představuje zvláště vhodný příklad, protože ilustruje zásadní skutečnost – efekt chyby není omezen pouze na celkový výsledek nebo zneužití dané služby. Nejčastěji je systém narušen vinou jediné vadné součásti. Pro tyto případy neexistuje dokonalá ochrana.

Oranžová kniha se snaží s takovými problémy vypořádat zaměřením na

procesy a požadavky na záruky pro výše hodnocené systémy. Proto například požadavky na zabezpečení úrovně B3 zahrnují následující prohlášení [DoD83], strana 37, část 3.3.3.1.1 – Architektura systému):

> TCB[13] (Zabezpečená výpočetní základna) by měla být projektována a strukturována tak, aby použila úplný, koncepčně jednoduchý ochranný mechanizmus s přesně definovanou sémantikou. Tento mechanizmus by měl hrát ústřední úlohu při uplatňování vnitřního strukturování TCB a systému. TCB by měla spojit v jedno platné použití úrovňování, abstrakci a skrývání dat. Platná systémová technika by měla být směrována k minimalizaci složitosti TCB a vyloučení takových modulů z TCB, které nejsou z hlediska ochrany významné.

Uvádění takových doporučení a nařízení do praxe není jednoduché. Žádný programátor nebo správce systému nemůže prostě najednou začít psát bezchybný kód. Nicméně se dá učinit řada opatření ke snížení chybovosti. O této problematice se více zmiňuji v části 2.5.

Na tomto místě provedu závěrečné shrnutí poznatků o chybách softwarového vybavení: Unix je vystaven vážnému riziku útoků na programové vybavení. Proti mnoha dobře známým útokům se můžeme chránit a snížit riziko tím, že zajístíme dodržování následujících obecných pravidel:

- Vždy je nutné instalovat prověřený software přímo od výrobce Unixu nebo ze známých distribučních míst. Bezpodmínečně je nutné instalovat pravidelné opravy systému (tzv. "patche"). Pokud výrobce nebo distributor podporuje šíření programového vybavení chráněné digitálními podpisy nebo jinými šifrovacími metodami, chránícími software před nežádoucí modifikací, volíme vždy tento typ distribuce.

[13] Trusted Computing Base

- Nikdy nevytvářet ani neinstalovat skripty v shellu s propůjčenými přístupovými právy vlastníka (tzv. *setuid*, zejména ne privilegovaného uživatele `root`). Existuje velmi mnoho dobře známých technik, které mohou narušitelé použít pro získání přístupu k shellovým skriptům, které běží pod uživatelem `root`, a tím proniknout do systému.

- Při jakémkoli volání funkce knihovny či jádra (např. *system*, *exec*, *open*, ...) se vždy uvádí plné jméno souboru nebo spustitelného programu, včetně cesty.

- Protože není důvod, aby běžní uživatelé měli k souboru s nastaveným *setuid* (nebo obecně k jakémukoliv spustitelnému souboru podobné povahy) přístup *read*, je vhodné použít příkaz *chmod* a souborová oprávnění pro *setuid* nastavit na hodnotu 4711 a pro ostatní programy 751.

- Nikdy se nesmí obětovat snaha o psaní správných a ověřitelných programů pro dosažení maximální výkonosti. Program nesmí být složitý, používat nekorektní programátorské techniky a nesmí zbytečně využívat nastavení *setuid*. Ceny počítačového vybavení klesají ze dne na den, hardware je stále levnější a rychlejší. Čas věnovaný nápravě po napadení a čas uživatelů, kteří nemohou normálně vykonávat svou činnost, je naopak čím dál dražší.

2.9.3 Selhání autentizace

Některé z útoků, které jsem popisoval již dříve, často využívají i takzvané selhání autentizačního mechanizmu. Pod termínem selhání autentizačního mechanizmu se rozumí stav, ve kterém mechanizmy pro autentizaci, které by mohly být dostatečné, byly nějakým způsobem zkompromitovány. Se selháním autentizačního mechanizmu se nejčastěji setkáváme při použití některé unixové síťové služby.

Například autentizace pomocí zdrojové IP adresy může za jistých podmínek fungovat (za předpokladu, že je nainstalován dobře nakonfigurovaný firewall, který zabrání *IP spoofingu*[14]) [Uher98]. Narušitelé ale mohou použít pro přenos jistých požadavků server `portmapper`[15]. V tomto případě bude napadený cílový počítač oklamán, protože zpráva se zdála být lokálního původu, ale její skutečný původ byl jiný.

Autentizace na základě adresy selže také v případě, že zdrojový počítač není důvěryhodný. Protokol TCP/IP byl navrhován v době, kdy se využívaly zejména centrální výkonné počítače používající algoritmus sdílení času. V současné době se ale na většině sítí uplatňuje model distribuovaných výpočtů. Původní bezpečnostní mechanismy – uživatelská hesla – dnes již neposkytují v rozsáhlých sítích dostatečnou autentizaci (viz. odstavec 2.9.1).

Někdy autentizační protokol selže, protože neobsahuje správné informace. TCP/IP neidentifikuje odesílajícího uživatele. Vyšší aplikační protokoly vystavěné na bázi TCP/IP je musí získávat dodatečně nebo se bez nich musejí obejít zcela. Dokonce ani složitá kryptografická autentizace zdrojového počítače nebo jeho uživatele nemusí dostačovat. Narušený počítač nemůže provádět bezpečné šifrování, protože jeho kryptografické systémy byly zkompromitovány.

Obranou proti takovýmto druhům napadení je zavedení například systému výměny klíčů *Kerberos* (Kerberos Key Exchange). Tento nástroj byl vyvinut na MIT jako součást projektu *Athena*. Kerberos používá řadu bezpečnostních opatření a jediný důvěryhodný server pro správu přístupu k rozsáhlým distribuovaným systémům. Základní principy systému jsou jednoduché: pouze omezený počet počítačů může být opravdu bezpečný. Pro prezentaci uživatelů používá systém Kerberos tzv. *jména činitelů*

[14] Falšování IP adresy

[15] Program pro konverzi čísel programů `RPC` (Remote Procedure Call) na čísla portů protokolu DARPA (TCP/IP)

(principal names) a pro reprezentaci oblastí spravovaných důvěryhodným serverem používá *jména říší* (realm names). Nicméně, ani systém Kerberos není imunní vůči útokům. Obsahuje několik potenciálně slabých míst v autentizačním protokolu, které představují pro bezpečnost systému určité potenciální riziko [Stev90], [Klan97].

2.9.4 Selhání protokolu

V předchozích kapitolách jsem se zabýval situacemi, kdy funkce nebo služba systému Unix pracovala správně, ale selhala autentizace nebo došlo ke zkompromitování systému nebo jeho části. V této části se zaměřím na jinou skupinu bezpečnostních problémů - použité protokoly nebo postupy samy o sobě jsou již chybné. Tím je systému nebo aplikacím odepřena možnost jejich používání.

Názorným příkladem je útok na pořadová čísla síťové relace TCP [Ches95], [Klan97] nebo [Chap95]. Špatně navržený algoritmus pro generování náhodných počátečních pořadových čísel pro navazující se TCP spojení umožňuje útočníkovi použít *„Sequence–number prediction attack“*[16], nebo metodu *IP spoofingu*. Původní návrh protokolu TCP nepředpokládal použití pořadového čísla k obraně proti úmyslným útokům. Bohužel, návrh protokolu TCP je zcela nedostatečný, protože se spoléhá na autentizaci na základě IP adresy. Podobné nebo odvozené protokoly, které se obdobně jako TCP spoléhají na pořadová čísla, jsou napadnutelné stejným druhem útoku.

Částečnou ochranou je opět použití mechanismů založených na distribuci klíčů (obdobně jako systém Kerberos, viz. předchozí část 2.9.3). Tyto algoritmy jsou vystavěné na následujícím základě: navzájem komunikující počítače musí před samotným zahájením komunikace znát veřejný klíč druhé strany [Prib96]. Tato informace se nejčastěji v rozsáhlích sítích distribuje

[16] Uhodnutí pořadového čísla

pomocí `NIS`. Služba `NIS` patří mezi nezabezpečené služby využívající mechanismus `RPC`. Mechanismus `RPC` je založen na protokolu TCP/IP.

Jestliže je výměna klíčů narušena, je zkompromitován i celý autentizační protokol a následně i veškerá komunikace mezi oběma systémy. Proto je nutné použít v rozsáhlých unixových sítích důvěryhodný a prověřený kryptografický systém, který zabezpečí distribuci veřejných a soukromých klíčů, a následně i distribuci dočasných klíčů síťových relací [Prib96].

2.9.5 Nežádoucí únik informací

Velká většina protokolů při své činnosti umožňuje narušitelům získání určitých informací. Získání informací je většinou záměrem lidí, kteří tyto služby využívají. Tyto informace jsou ale také velmi často cílem agentů průmyslové špionáže (viz. kapitola 2.6 nebo [Ches95], [Prib96]), popřípadě mohou být nápomocné při průniku do zabezpečeného systému. Názorným příkladem je unixová služba `finger`. Poskytuje velmi mnoho cenných informací pro hádání hesel.

Jiným vhodným příkladem je uveřejnění čísel telefonů a kanceláří na podnikovém webovském serveru. V knize *Všichni prezidentovi muži* [Wood74] je popsán způsob, který byl v aféře Watergate použit k odvození vnitřní organizační struktury Výboru pro znovuzvolení prezidenta pomocí informací uveřejněných v jeho telefonním seznamu. Proto je velmi důležité zvážit, které informace se budou uveřejňovat a jakým způsobem.

Další službou, která obdobně jako `finger` poskytuje velké množství cenných informací je služba `DNS`[17]. Poskytuje například informace o vnitřní organizaci sítě a je proto velmi vhodná pro vyhledávání budoucích cílů vhodných pro napadení. V případě služby `DNS` je kontrola informací obtížná. Jediným řešením je omezení záznamů v `DNS` pouze na firewally a vstupní brány do interní sítě.

[17] Domain Name Service

Z těchto poznatků vyplývá, že jedinou účinnou obranou proti nežádoucímu úniku informací je důsledná a promyšlená konfigurace služby.

2.9.6 Útok na dostupnost služby

Velkou většinu služeb, které poskytuje operační systém Unix, lze používat síťově. Proto je takové služby snazší vyřadit z činnosti než služby lokálního charakteru. Typů útoků na dostupnost služby je známo mnoho druhů.

Nejtriviálnější a zároveň maximálně účinnou formou je snaha o zaplnění disků unixového serveru poštovními zprávami nebo nadměrným množstvím dat ukládaných pomocí `FTP`. Existují sice nástroje pro snížení rizika napadení tohoto druhu (např. zavedení diskových kvót nebo zavedení horní hranice velikosti poštovní zprávy), na druhou stranu je ale těžké určit horní hranici těchto limitů. Pro narušitele je velmi jednoduché místo jednoho obrovského souboru posílat stovky souborů velkých například 1MB. Navíc tento způsob napadení vyvolá na serveru spuštění velkého množství obslužných procesů (typický příklad je poštovní démon `sendmail`), které vede k zahlcení systému. Vhodnou, ale ne nejlepší, ochranou je instalace naddimenzovaného hardware (disky, operační paměť, ...) na veřejně dostupné servery. Další možnou obranou je oddělení veřejných služeb od interní síťě. To znamená v samotném návrhu topologie sítě oddělit počítače pro obsluhu pošty, `FTP`, `WWW`, `proxy služeb` a dalších služeb a umístění těchto serverů v takzvané demilitarizované zóně (*DMZ*). *DMZ* je chráněná oblast oddělená od od veřejné i interní sítě, která slouží k umístění veřejně přístupných serverů, u nichž existuje největší riziko napadení.

Další typy útoků na dostupnost služby jsou již více propracované. Nejznámějším příkladem je zasílání velkého množství falešných paketů *ICMP*[18] na počítače, u kterých chce narušitel dosáhnout narušení schopnosti komunikace pomocí protokolu TCP/IP. Typicky se jedná o *ICMP* zprávy typu

[18]Internet Control Message Protocol

destination unrechable (hostitel, síť či port je nedostupný), *redirect* (dočasná změna směrování) či zprávy měnící masku sítě. Dalším typem útoků spadající do podobné kategorie jsou útoky vedené na směrovací protokoly. Tyto útoky neslouží pro přímé vniknutí do zabezpečeného systému, ale obdobně jako útoky vedené pomocí protokolu *ICMP*, znemožňují napadenému počítači komunikaci po síti.

Nejlepší ochranou proti útokům tohoto typu je dobře nakonfigurovaný paketový filtr. Paketový filtr je v podstatě klasický směrovač, který na základě určitých, předem jasně definovaných tabulek pravidel rozhoduje, zda příchozí nebo odchozí paket propustí či nikoliv. Paketový filtr může současně fungovat jako prostředek pro záznam informací o přenesených datech, tzn. poskytuje službu pro účtování dat [Uher98].

3 Audit a monitorování Unixu

V předchozích kapitolách jsem rozebral základní typy útoků na operační systém Unix a počítačové sítě. U některých typů útoků jsem uvedl i možná bezpečnostní opatření vedoucí ke snížení či zamezení jejich výskytu.

V následující části textu se budu zabývat vlastnostmi a principy systému určeného pro audit a monitorování Unixu. Návrh a implementace auditního systému pro operační systém Unix (nebo síť unixových počítačů) je cílem mé diplomové práce.

3.1 Důvody pro zavedení auditu

V kapitole 2.9 jsem podrobně rozebral problematiku bezpečnosti operačního systému Unix a bezpečnostních otázek spojených s používáním Unixu v síťovém výpočetním prostředí. Audit představuje jednu z nejlepších metod pro odhalení nežádoucích činností v systému.

Význam slova *audit* je „zkoumat nebo vyšetřovat se záměrem prověřit“. Slovo audit v oboru výpočetní techniky je odborný termín pro činnost, která ověřuje, zda se v praxi uplatňují předem dohodnutá pravidla a postupy. Pokud se například jedná o bezpečnostní audit, rozumí se pod tímto termínem ověření, zda se v praxi uplatňují a dodržují pravidla a postupy uvedené v bezpečnostní politice, viz. kapitola 2.5. V odborné literatuře se někdy termín auditovat zaměňuje s termínem *monitorovat*. Význam slova monitorovat je „kvalitativně sledovat či kontrolovat nějakou činnost“. V oboru výpočetní techniky se termín monitorovat používá pro dlouhodobé sledování určité veličiny s a nebo bez možnosti ukládání sledovaných hodnot. Rozdíl mezi auditem a monitorováním spočívá ve schopnosti auditu získané informace prověřit vzhledem k předem daným kritériím.

Auditní záznam (*audit trail*) je semi–permanentní záznam aktivit prováděných v systému, který je spravován a udržován operačním systémem

počítače. Auditní záznamy poskytují informace využitelné například k vyhodnocení situace po úspěšném neautorizovaném průniku do systému, ale také jsou velmi nápomocné v samotném průběhu napadení.

Americké národní centrum počítačové bezpečnosti *NCSC* [19] odsouhlasilo následující definici auditního záznamu:

> Auditní záznam je chronologickým záznamem systémových aktivit dostatečný pro rekonstrukci, revizi a zkoumání posloupnosti stavů prostředí a aktivit, zúčastňujících se realizace operace, procedury nebo události v transakci od jejího počátku po její konečný výsledek.

Z této definice vyplývá, že auditní záznam je automatická metoda pro sledování veškerých transakcí, které probíhají v operačním systému nebo na síti. Pro auditní záznamy navíc platí následující obecná pravidla:

- Auditní záznam je vždy záznam všech činností, nebo určité podmnožiny činností, které na počítači probíhají
- Auditní záznamy se udržují pro prostředky nebo objekty s omezeným přístupem
- Auditní záznamy slouží k vyhodnocení rozsahu a škod po nenadálé výjímečné události (např. napadení systému, havárie disku, ...)

3.2 Struktura auditních záznamů

Obecně lze auditu podrobit libovolnou aktivitu nebo událost na síti nebo v systému. Moderní operační systému vyhovující požadavkům úrovně C2 na zabezpečení systému [DoD83], poskytují audit uživatelů, skupin i objektů na systémové úrovni.

[19] National Computer Security Center

Velká většina administrátorů síťových výpočetních prostředí používá auditní záznamy z těchto důvodů:

1. Detekce a odhalení neautorizovaného přístupu
2. Sledování a vyhodnocování činnosti oprávněných uživatelů systému
3. Sledování a vyhodnocování přístupů k serverům
4. Sledování a vyhodnocování síťového provozu na firewallu

Podle výše uvedeného popisu využití lze auditní záznamy strukturovat podle druhu informací a podle objektů, ke kterým se auditní záznamy váží. Například lze provádět audit na úrovni počítače (záznamy činosti hardware), audit na úrovni služeb (hlášení jednotlivých systémových démonů), audit na úrovni uživatelů (záznam informací o přihlášení a odhlášení uživatelů), nebo audit na úrovni specifických objektů (podrobné sledování určité části systému)

Jiným způsobem strukturování auditních záznamů je jejich rozdělení z hlediska časového plánování na:

- předběžný audit
- plánovaný audit
- audit ve stavu ohrožení

3.2.1 Předběžný audit

Jedním z bezpečnostních problémů, který přispívá ke zvýšení pravděpodobnosti napadení systému, je nedodržení bezpečnostních procedur a standardů na nově instalovaných počítačích v síti. Auditem těchto počítačů před tím, než se fyzicky připojí do sítě, lze odstranit velké množství bezpečnostních problémů. Nelze se zcela spoléhat na nastavení systému, které provádí výrobce

počítače nebo distributor použité verze Unixu. Podrobením těchto počítačů velmi přísnému bezpečnostnímu auditu lze docílit vyloučení většiny slabých míst v konfiguraci systému, programech, přístupových právech a autentizaci uživatelů. Z toho následně vyplývá znatelné snížení rizika napadení nového počítače, ale také celé sítě, do které bude počítač připojen.

3.2.2 Plánovaný audit

Pravidelně plánovaný audit může prověřit, zda jsou dodržovány bezpečnostní standardy a velmi výrazně snižuje riziko vzniku bezpečnostních problémů. Při návrhu plánování pravidelného auditu musí být do úvahy zahrnuty všechny možné faktory ovlivňující bezpečnost informačního systému (např. topologie sítě, počet připojených počítačů, citlivost uchovávaných dat, různé skupiny uživatelů, ...). Pro pravidelně plánovaný audit lze použít následující, praxí ověřená doporučení:

- audit koncových počítačů by se měl provádět každých 12 – 24 měsíců
- audit rozlehlých sítí by se měl provádět každých 24 měsíců
- audit malých sítí by se měl provádět každých 12 měsíců
- audit firewallů by se měl provádět nejméně každých 6 měsíců

3.2.3 Auditu ve stavu ohrožení

V případě vzniku neočekávané události, která nějakým způsobem narušila zabezpečení systému je nutné určit rosah škod. V této chvíli je audit patrně nejúčinnější ze všech dostupných bezpečnostních mechanismů. Audit jako jediný může prověřit integritu systému, přístupová práva souborů a adresářů nebo modifikaci binárních souborů. Operační systém Unix obsahuje velké množství souborů a adresářů, které se mohly stát cílem při úspěšném napa-

dení systému. Prověření tak velkého množství souborů a adresářů není možné provést jiným způsobem, než bezpečnostním auditem.

3.3 Využití auditních záznamů

Využití auditních záznamů představuje jednu z nejlevnějších bezpečnostních technik. Auditní záznam je ve své podstatě automatická metoda pro monitorování všech událostí a transakcí probíhajících v systému či počítačové síti. Auditní záznam tak představuje důležitý nástroj pro administraci síťového výpočetního systému.

Auditní záznam umožňuje odhalení nepatřičné činnosti v systému, kterou může vyvíjet například narušitel. Auditní záznam upozorní administrátora na probíhající útok a umožnímu provést patřičná protiopatření. Mimo to umožňuje audit monitorovat interní uživatele a prověřit, zda systém nepoškozují.

3.4 Nevýhody auditních záznamů

Auditní záznamy, jejich udržování a vyhodnocování neposkytují takovou ochranu, která by v případě výskytu mimořádné události ukončila činnost systému nebo převedla systém do bezpečnějšího stavu (*fail–safe protection*). Pokud například narušitel simuluje chování některé části systému, která je útokem vyřazena z činnosti, nebo je jiným způsobem modifikována její činnost (*active spoofing*), auditní systém nemusí takovou činnost zaznamenat. Jiným příkladem může být pasivní shromažďování informací (*passive sniffing*). Auditní systém tuto činost nemusí opět zachytit, protože narušitel se nepokouší aktivně přistupovat k žádným datům , ale pouze provádí jejich odposlech. Auditní systém naproti tomu dobře registruje aktivní formy útoků, např. *SYN flood attack*, při kterém dochází k velké záplavě `ACK` a `SYN` pakety během spojení protokolem TCP/IP.

Z těchto poznatků vyplývá, že správné použití auditního systému je pouze jednou z mnoha rozsáhlých součástí zabezpečovacího plánu informačního systému. Auditní systém nemůže nikdy nahradit bezpečnostní prvky (firewall, proxy server, viz. odstavec 2.9.6), či formální nástroje zabezpečení (bezpečnostní politiku, viz. kapitola 2.5). Naopak, jiné obranné a bezpečnostní mechanizmy nemohou zcela nahradit audit.

3.5 Standardní auditní nástroje

Unix obsahuje širokou paletu auditních a záznamových nástrojů a přidružených programů. Většina moderních unixových distribucí automaticky během instalace systému konfiguruje základní nástroje pro audit systému. Velká výhoda unixových auditních nástrojů spočívá ve schopnosti udržovat záznamové deníky (*logs*) ve fomě textových souborů, používajících kódování ASCII. Pouze malá část záznamových deníků je v binární formě (binární formát byl zvolen pro zvýšení efektivity častého přístupu k záznamovému deníku). Pro tyto záznamové deníky existují nástroje pro jejich převod do čitelného textového tvaru.

3.5.1 Auditní nástroj lastlog

Auditní nástroj `lastlog` je jedním ze základních auditních nástrojů Unixu. Slouží pro sledování místa a času posledního přihlášení jednotlivých uživatelů do systému. Tento auditní nástroj úzce spolupracuje s přihlašovacím programem `login`. Ve chvíli, kdy se uživatel přihlašuje do systému, `login` vyhledá identifikační číslo uživatele v auditním deníku nástroje `lastlog`. Na obrazovku uživatele je následně vypsán čas a místo (port terminálu, IP adresa počítače, ...) posledního přihlášení uživatele. Moderní verze Unixu navíc zobrazují informaci o tam, zda byl poslední pokus úspěšný nebo neúspěšný. To umožňuje uživateli zjistit, zda někdo nezcizil jeho uživatelský účet.

Následně program `login` provede aktualizaci auditním deníku nástroje `lastlog` tak, aby obsahoval nový čas a místo přihlášení nebo pokusu o přihlášení. Dodatečně program `login` provede aktualizaci auditních deníků *utmp*, *wtmp* anebo *btmp*.

3.5.2 Auditní nástroj who

Auditní nástroj `who` vyhodnocuje informace obsažené v auditním deníku *utmp*. Auditní deník *utmp* obsahuje informace o právě přihlášených uživatelých. Informace uložené v tomto auditním deníku se velmi rychle mění. Tento auditní deník je také zajímavý tím, že neudržuje žádnou historii. To znamená, že poté, co se uživatel odhlásí ze systému, nebude auditní deník *utmp* obsahovat žádný záznam o jeho předešlé přítomnosti a činnosti v systému.

3.5.3 Auditní nástroje last a ac

Auditní nástroj `last` vyhodnocuje informace uložené v auditních denících *wtmp* a *btmp*. Auditní deník *wtmp* udržuje informace o přihlašování a odhlašování uživatelů do a ze systému Unix. Na rozdíl od auditního deníku *utmp* si deník *wtmp* uchovává trvalé záznamy o každém úspěšném přihlášení a odhlášení všech uživatelů. Auditní deník *btmp* uchvává stejné záznamy, ale pro neúspěšné přihlášení či odhlášení všech uživatelů. Vedle těchto informací obsahuje deník *wtmp* navíc informace o startu a ukončení činnosti systému.

Pro vyhodnocování záznamů obsažených v deníku *wtmp* lze ještě využít auditní nástroj `ac`. Tento nástroj vyhodnocuje a formátuje informace obsažené v deníku *wtmp* například podle jednotlivých uživatelů, podle data a podobně. Nástroj `ac` dále vyhodnocuje celkový čas přihlášení uživatele a umožňuje tak určit, zda některý uživatel nespotřebovává příliš mnoho procesorového času.

3.5.4 Auditní nástroj syslogd

Auditní nástroj `syslogd` slouží pro záznam většiny událostí v systému, kromě záznamů o přihlašování a odhlašování uživatelů. Tento nástroj využívá mechanismus zaznamenávání zpráv (*message logging*). Je to tedy centrální referenční auditní nástroj pro zpracování výskytu všech možných událostí v systému. Tento auditní nástroj umožňuje jako jeden z mála standardních nástrojů Unixu třídit a vyhodnocovat události podle předem nastavených pravidel. Pro jednotlivé události lze definovat místo uložení zprávy, na které má auditní nástroj `syslogd` provést záznam v případě jejich výskytů. Konfigurační soubor auditního nástroje `syslogd` obsahuje záznamy, které se skládají z pole selektoru (*selector field*) a pole akce (*action field*). Pole selektoru určuje, které události má auditní nástroj `syslogd` zaznamenávat. Tento záznam obsahuje zkrácené jméno nebo typ programu (označované jako *facility*), který událost vyvolal. Dále tento záznam obsahuje úroveň naléhavosti vzniklé události (*severity level*). Z těchto informací může auditní nástroj `syslogd` vyhodnocovat události podle jejich důležitosti. Pole akce určuje činnost, která se má provést při vzniku dané události.

Zajímavou vlastností auditního nástroje `syslogd` je schopnost uchovávat systémové záznamy na více počítačích. To je velká výhoda ve chvíli, kdy dojde k havárii systému nebo neautorizovanému průniku do systému.

Většina distribucí Unixu obsahuje standardní systémové volání v jazyce C nazvané *syslog*. Toto systémové volání umožňuje programátorům ve svých programech využívat služeb auditního nástroje `syslogd`. Pro programátory, kteří programují skripty v příkazových interpretrech, je k dispozici nástroj `logger`, který plně nahrazuje systémové volání *syslog*.

3.5.5 Další standardní auditní nástroje

Operační systém Unix obsahuje další řadu standardních auditních nástrojů a deníků. Můžeme například uvést auditní deník *sulog* který obsahuje infor-

mace o používání programu `su` pro změnu uživatelské identity.

Většina unixových démonů a služeb vede o své činnosti vlastní auditní deníky. Mezi nejznámější můžeme zařadit démony `sendmail` (obsluha elektronické služby), `cron` (pravidelné spouštění naplánovaných úloh), `lpd` (obsluha tisku na tiskárnách) a mnohé další.

V neposlední řadě je nutné zahrnout do této skupiny takzvané deníky historie shellu. Tento deník uchovává informace o několika posledních provedených příkazech. Moderní unixové interprety podporují tento mechanismus pro usnadnění práce a také zvýšení bezpečnosti. Jsou známé případy, kdy někteří narušitelé vymazali v systému všechny záznamy o své činnosti a zapomněli vymazat záznamy v deníku historie shellu. Pro administrátora systému je následná rekonstrukce činnosti narušitele v systému velmi jednoduchá.

3.6 Shrnutí

Odstavce 2.5 až 2.9.6 popisovaly velké množství různých typů napadení a průniků do síťových informačních systémů, které nějakým způsobem ohrožují uložená data či samotný počítač. Kapitola 3 ukázala, že auditní záznamy poskytují jeden z nejlepších prostředků pro detekci možných bezpečnostních či jiných vážných problémů, které mohou v systému vzniknout. Detekce takovýchto problémů je možná nejen poté, co se projeví plnou měrou, ale již na samém začátku jejich vzniku. Auditní záznamy zachycují všechny aktivity, které se v počítači vyskytnou. Některé operační systémy, jako například Microsoft Windows NT nebo Novell NetWare používají obecně pouze jeden soubor pro auditní záznam. Naproti tomu Unix používá pro auditní záznamy více souborů, z nichž některé poskytují duplicitní informace. Výhoda tohoto způsobu ukládání auditních záznamů je zřejmá na první pohled. V případě poškození jednoho nebo více auditních deníků je možné zpětně rekonstruovat činnost systému a provedené transakce ze zbylých záznamů v nepoškozených denících.

Je důležité monitorovat a zaznamenávat všechny typy neautorizovaných přístupů nebo podezřelých událostí v systému. Nejdůležitější povinností administrátora systému je však monitorování a vyhodnocování auditních záznamů pro objekty s omezeným přístupem, protože tyto objekty představují nejrizikovější oblast. Je také důležité nastavit na monitorovaných systémech různé úrovně auditu podle toho, jaké druhy informací jsou na serverech uchovávány. Je zbytečné například podrobovat rozsáhlému auditu pracovní stanici, která slouží pouze pro čtení elektronické pošty a neobsahuje žádná důležitá data. Na druhou stranu je velkým rizikem nezapnout veškeré auditní záznamy na poštovním serveru, který je připojen do veřejné sítě Internet a představuje tak velmi vhodný cíl pro proniknutí do interní zabezpečené sítě.

Auditní záznamy se nejčastěji vyhodnocují po úspěšném neautorizovaném průniku do systému. V takovém případě slouží pro stanovení rozsahu způsobených škod nebo ke zjištění, která data byla zkompromitována nebo zcizena. Operační systém Unix má v tomto případě velkou výhodu. Narušitel obvykle modifikuje záznamy v jednom auditním deníku a zapomene (nebo není schopen) pozměnit ostatní související informace o systému [Stol90]. Touto promyšlenou strategií uchovávání duplicitních záznamů v různých auditních denících dosahuje Unix většího stupně zabezpečení. Další možné uplatnění naleznou auditní záznamy při pravidelné kontrole systému. Vzhledem k tomu, že mnoho oprávněných uživatelů systému si neuvědomuje, že je zaznamenávána jejich činnost, auditní záznamy tak napomáhají při odhalování autorizovaných, ale nepatřičných aktivit.

Je velmi důležité auditní záznamy pravidelně udržovat (kontrolovat jejich integritu), analyzovat jejich obsah a zálohovat mimo monitorovaný systém. Po neoprávněném průniku do systému poskytuje auditní záznam jedinou možnost, jak zjistit způsob, kterým se narušitel do systému dostal, a také, co v něm provedl.

Náklady spojené s instalací nových bezpečnostních opatření, nebo náklady spojené s náhradou škod, jsou často velice vysoké. Společnost War Room Research se sídlem v Baltimore, Maryland, USA uvádí, že u 67% společností, které nahlásily neautorizovaný průnik do svého informačního systému, dosáhly náklady na nápravu škod 50 000 $. Ještě více alarmující je skutečnost, že 27% těchto společností vynaložilo na nápravu škod více jak 500 000 $. Sledování a vyhodnocování auditních záznamů představuje jednu z nejlevnějších alternativ bezpečnostních opatření, která zajišťuje dostatečnou úroveň zabezpečení systému.

4 Popis řešení a instalace

V zadání diplomové práce je požadováno navržení a implementace aplikace, která umožní audit a monitorování sítě unixových počítačů pro malé a střední organizace s omezenými finačními prostředky. Systém by měl být navržen tak, aby sjednotil důležité funkce, které jsou obsaženy v různých komerčních či volně dostupných systémech. Systém by měl umožňovat monitorování základních funkcí a prostředků sítě unixových počítačů (vytížení CPU, správa paměti, informace o diskových svazcích, informace o procesech, informace o uživatelích, informace o systémové konfiguraci, ...).

V předchozí kapitole byly popsány principy auditu v operačním systému Unix. Je tedy třeba navrhnout a implementovat program splňující výše uvedené požadavky. Součástí systému musí být grafické uživatelské rozhraní realizované pomocí webovského prohlížeče (tzn. s použitím jazyka HTML[20] a rozhraní CGI[21]).

4.1 Moderní auditní systémy

Pro zvýšení zabezpečení systému Unix v současné době vznikají stále nové auditní nástroje, které poskytují administrátorům neocenitelné služby při správě systému a aplikaci bezpečnostních opatření. Na Internetu lze nalézt mnoho auditních nástrojů pro operační systém Unix, dostupných zdarma nebo za úhradu. Tyto auditní systémy pokrývají různé části monitorování systému a slouží pro různé účely. Lze je rozdělit do dvou skupin.

První skupinu tvoří systémy vhodné pro nalezení chyby v konfiguraci. Některé z nich je možno použít při budování firewallu, jiné jsou vhodné pro detekci útoků. Další zase slouží k určení stávající úrovně zabezpečení instalovaného operačního systému. Můžeme například uvést systém `NADIR`[22], který po-

[20] HyperText Markup Language
[21] Common Gateway Interface
[22] Network Anomaly Detection and Intrusion Reporter

skytuje administrátorům systému rozšíření standardních auditních záznamů. Systém `NADIR` byl speciálně vyvinut pro rozlehlé sítě superpočítačů v Los Alamos National Laboratory. Je založen na pravidlech, které automaticky detekují průnik do systému. Slouží tedy pro zvýšení zabezpečení systému. Mezi nástroje podobného charakteru lze zařadit systémy `Titan`, `Stalker`, `WatchDog`, `TAMU`, `COPS`, `Tripwire`, `ISS`[23], `Satan`, `SPY`[24] nebo `NetSonar`. Více informací o těchto a dalších auditních nástrojích nalezne případný zájemce v [Sans99].

Druhou skupinu tvoří nástroje, které slouží pro usnadnění administrace většího počtu unixových počítačů zapojených v síti. Například auditní systém `SysMon` je navržen pro získávání a udržování informací ze sítě unixových počítačů. Získané informace slouží jako databáze dostupných prostředků na síti. Poskytuje administrátorům okamžitý přehled o fukčnosti a činnosti jednotlivých unixových počítačů. Systém `SysMon` není tedy určen pro zvýšení zabezpečení systému, ale spíše jako pomocný nástroj pro usnadnění každodenní práce administrátorů. Dalším příkladem může být auditní systém `Big Brother`. Slouží obdobně jako `SysMon` k monitorování využití a dostupnosti různých částí unixových systémů. Jiným příkladem auditních systémů, které můžeme zařadit do této skupiny, jsou systémy `Nessus`, `NCARP` nebo `NFR`[25].

4.2 Architektura auditního systému Wotan

Při návrhu architektury nového auditního systému, který byl později implementován v rámci mé diplomové práce, jsem vycházel z modulární struktury systému `Titan`. Auditní systém jsem ale nekoncipoval jako systém sloužící pro zvýšení zabezpečení systému, ale jako systém, který lze použít při moni-

[23] Internet Security Systems
[24] Security Profile Inspector
[25] Network Fly Recorder

torování dostupnosti a využití různých částí unixových počítačů zapojených v síti. Tento systém přesto převzal některé rysy auditních nástrojů spadajících do první skupiny uvedených v části 4.1.

Nově navrhovaný auditní systém jsem pojmenoval `Wotan`. Auditní systém `Wotan` je kolekce malých a funkčně ucelených nástrojů (modulů), využívajících standardního programového vybavení pro administraci unixových systémů. Tato myšlenka vychází z původní filosofie operačního systému Unix, viz. odstavec 2.1. `Wotan` ve velké míře používá standardní nástroje a nesnaží se do systému přidávat velké množství nového neprověřeného software. Některé auditní systémy zcela obcházejí dostupné programové vybavení a zanášejí do systému velké množství (z hlediska bezpečnosti) neprověřeného kódu. To je v rozporu s jejich vlastním posláním – snižování rizik zranitelnosti zabezpečení systému, viz. odstavec 2.9.2.

Základní funkce jednotlivých modulů systému `Wotan` jsou následující:

- Shromažďování a analýza informací o oprávněných uživatelích systému. Modul *User and group monitoring* poskytuje informace o účtech jednotlivých uživatelů, skupinových účtech a provádí monitorování uživatelské aktivity v systému.

- Shromažďování a analýza informací týkající se konfigurace sítě. Modul *Network off-line monitoring* analyzuje konfiguraci a využívání síťových zařízení, protokolů a směrování.

- Shromažďování a analýza informací získaných monitorováním souborových systémů. Modul *Filesystems monitoring* provádí sledování využití diskových svazků a souborových systémů.

- Shromažďování a analýza informací o systémových prostředcích. Modul *System monitoring* analyzuje informace týkající se instalovaných zařízení, interních struktur jádra, řízení přidělování paměti, vstupně/výstupního systému a dalších systémových komponent.

Auditní systém `Wotan` provádí svou činnost ve dvou krocích. V prvním kroku nejprve `Wotan` ve stanovený čas provede audit systému a nashromáždí dostupné informace. Z tohoto hlediska ho lze tedy zařadit mezi nástroje pro plánovaný audit, viz. odstavec 3.2.2. Pro audit systému obsahuje distribuce nástroje `Wotan` tzv. *auditního agenta*, viz. obr. 5 v příloze A.1. Tento agent se instaluje na jednotlivé počítače v síti, které mají být monitorovány, viz. obr. 4 v příloze A.1. Agent při svém spuštění nejprve načte a vyhodnotí konfigurační soubor, který řídí jeho činnost. Popis tohoto konfiguračního souboru bude uveden později.

Následně agent vyhledá inicializační modul a vykoná v něm uložené příkazy. Inicializační modul se vždy nahrává jako první. V současné verzi systému `Wotan` slouží tento modul pouze pro nastavení časové známky (*timestamp*) auditního běhu.

Po provedení příkazů uvedených v inicializačním modulu se podle nastavené konfigurace zavedou jednotlivé moduly pro audit systému. Nejprve jsou vykonány příkazy pro audit systému uložené v modulech pro sledování informací o účtech oprávněných uživatelů, skupinových účtech a monitorování uživatelské aktivity v systému. Tyto informace jsou důležité pro kontrolu integrity databáze oprávněných uživatelů a detekci případných pokusů o neautorizovanou modifikaci této databáze. Získané informace také poskytují přesný obraz o činnosti právě přihlášených uživatelů v systému. Z tohoto hlediska se v podstatě jedná o rozšíření informací, které poskytuje standardní auditní deník *utmp* nástroje `who`.

Jako další se vykonají příkazy uložené v modulech pro sledování konfigurace sítě. Sledovány jsou informace o síťových zařízeních, protokolech (TCP, UDP[26], ARP[27], ...), směrování, otevřených síťových relací, atd. Tyto informace jsou často cílem útoků nasměrovaných proti dostupnosti využívání

[26] User Datagram Protocol
[27] Address Resolution Protocol

počítače. Při modifikaci těchto informací dojde často k úplnému vyřazení systému z činnosti (zejména pokud poskytuje nějaké síťové služby, nebo naopak je na nějaké síťové službě zcela závislý). Pomocí těchto informací lze například odhalit některé aktivní útoky popisované v části 2.9.4 nebo 2.9.6. Je proto nutné kontrolovat integritu těchto informací.

Následně jsou vykonány příkazy uložené v modulech pro sledování informací o využití diskových svazků a souborových systémů. Některé typy útoku na dostupnost služby, např. snaha o zaplnění disků unixového počítače poštovními zprávami nebo nadměrným množstvím dat, jsou velice účinné a nebezpečné. Modul pro sledování využití diskových svazků a souborových systémů je velice užitečný při detekci právě takovéhoto typu útoku.

Jako poslední jsou vykonány příkazy uložené v modulech pro sledování systémových prostředků. Monitorovány jsou instalovaná zařízení, vytížení CPU, interní struktury jádra, inicializační skripty systému, běžící procesy, řízení přidělování paměti, vstupně/výstupní systém a další části systému. Monitorováním těchto prostředků lze také detekovat některé typy útoků. Jestliže se například v systému, ve kterém obvykle neběží mnoho procesů a není alokováno větší množství paměti, tyto hodnoty znatelně zvýší, může to znamenat neautorizovaný průnik do systému a následné zneužívání jeho prostředků. Moduly pro sledování systémových prostředků mohou detekovat právě takovéto typy útoků a mohou být nápomocné při zjistění rozsahu způsobených škod nebo návrhu nutných protiopatření.

Všechny získané informace auditní agent v poslední fázi mezi sebou prováže a vytvoří mezi nimi křížové odkazy. Tím je jeho činnost na monitorovaném počítači ukončena. Nashromážděná data jsou připravena k následnému vyhodnocení. Ve druhém kroku se provede analýza těchto údajů. Analýza získaných informací se provádí v režimu *off–line*. To znamená, že vyhodnocení informací se uskuteční teprve ve chvíli, kdy administrátor přistupuje k získaným datům prostřednictvím rozhraní CGI.

Pravidla pro vyhodnocování získaných informací jsou pevně zabudována do rozhraní CGI a nelze je měnit. Administrátor přesto může vyhodnocování údajů řídit. Vyhodnocování pravidel lze ovlivnit nastavením referenčních hodnot, které jsou uloženy v konfiguračním souboru. Tento konfigurační soubor se načítá při každém přístupu k nashromážděným datům prostřednictvím rozhraní CGI. Při dosažení nebo překročení těchto referenčních hodnot je administrátor na tuto událost upozorněn. Příklad konfiguračního souboru s nastavenými referenčními hodnotami je uveden v příloze A.2.

Auditní systém `Wotan` je dále doplněn o síťový monitor v režimu *on–line* (tzn. umožňuje monitorování v reálném čase). Tento monitor zjišťuje informace o vytížení systému, aktuálním počtu přihlášených uživatelů a době, po kterou je systém v činnosti nebo naopak je z nějakého důvodu z činnosti vyřazen. Dále tento monitor sleduje propustnost sítě a dostupnost síťových služeb instalovaných na jednotlivých počítačích. Tyto sledované hodnoty jsou pro chod sítě unixových počítačů patrně nejpodstatnější. Z tohoto důvodu jsem při návrhu architektury auditního systému `Wotan` zvolil pro tuto část režim *on–line*, aby byly neustále k dispozici aktuální hodnoty.

Poslední částí systému `Wotan` je auditní nástroj pro sledování standardního rozšíření databáze MIB[28] protokolu SNMP[29] pro operační systém Linux. Také tento modul pracuje v reálném čase. Navíc obsahuje jednoduchý databázový server pro uložení a údržbu takto získaných informací.

SNMP monitor jsem zahrnul do návrhu architektury auditního systému `Wotan` pro zvýšení dostupnosti informací o monitorovaných prostředcích sítě unixových počítačů. Pokud by byl například na koncovém počítači auditní agent z nějakého důvodu vyřazen z činnosti, je nutné poskytnout administrátorovi alternativní nástroj pro monitorování tohoto počítače. Proto poskytuje SNMP monitor obdobné nebo stejné informace jako auditní agent. Tím

[28]Management Information Base

[29]Simple Network Management Protocol

je dosaženo většího stupně zabezpečení. SNMP monitor je dostupný pouze pro počítače s operačním systémem Linux. Operační systém Solaris žádné obdobné rozšíření databáze MIB standardně neposkytuje.

Po důkladném rozvážení jsem se rozhodl navrhnout auditní systém `Wotan` tak aby nevyžadoval pro svou činnost propůjčení přístupových práv privilegovaného uživatele `root`. Musí tedy využívat pouze nástroje dostupné běžným uživatelům.

4.3 Distribuce systému Wotan

Distribuce systému `Wotan` je rozdělena do dvou samostatných částí. První část tvoří systém pro audit sítě unixových počítačů. Tato část obsahuje auditního agenta, jehož funkce a činnost byly popisovány v předchozí části textu, a další pomocné programy. V současné době je tento agent implementován pro operační systémy Sun Solaris ve verzi 2.5, 2.5.1, 2.6 a Solaris 7 pro platformy Sparc a Intel. Agent byl také přenesen do operačního systému Linux s jádrem 2.0.x na platformě Intel. Pro odladění agenta jsem zvolil linuxové distribuce Debian 2.0 „Hamm“ a Debian 2.1 „Slink“. Distribuce Debian Linux je řazena mezi nejlépe propracované linuxové distribuce. Při návrhu této distribuce kladou její autoři důraz zejména na stabilitu a bezpečnost. Z těchto důvodů byla distribuce Debian Linux vybrána pro interní servery ÚHKT[30] na platformě Intel. Interní servery ÚHKT obsahují řadu citlivých dat, například mzdové a osobní údaje zaměstanců, finační účetnictví, databáze hospitalizovaných pacientů, informace o vyšetřeních, diagnózách, lécích, registr dárců krve včetně dárců vzácných krevních skupin a vyřazených dárců – např. s diagnózou AIDS, a mnohé další. Proto jsou na operační systémy použité na serverech v interní síti ÚHKT kladeny velmi přísné požadavky.

Verze auditního agenta pro Linux s jádrem 2.2.x je zatím ve vývoji a ve stádiu alfa testů. Současné verze distribuce Debian Linux jsou vystavěny

[30] Ústav hematologie a krevní transfuze v Praze

nad verzí jádra 2.0.x. Nová distribuce, vystavěná nad jádrem 2.2.x, bude k dispozici nejprve koncem ledna a to ve verzi *frozen* (to znamená, že vývoj distribuce je pozastaven a provádí se pouze odstraňování chyb). Vydání stabilní verze distribuce Debian Linux 2.2 „Potato“ se dá očekávat v průběhu roku 2000. Do této doby nelze v plné míře provádět vývoj agenta pro jádro 2.2.x, protože systémové knihovny a obslužné programy plně nepodporují novou verzi jádra. Nejvážnějším problémem jsou provedené změny a úpravy ve struktuře pseudo–souborového systému `/proc` nového jádra.

Z uvedených důvodů lze používat auditní systém `Wotan` jen na počítačích, na kterých je nainstalována mimo již dříve zmíněných také distribuce Linuxu RedHat 5.x, Caldera OpenLinux 1.x nebo jiná linuxová distribuce s jádrem 2.0.x. Zvolená distribuce Linuxu musí plně splňovat normu *SVID*[31]. Bez splnění této podmínky nemusí auditní agent korektně pracovat.

Druhá část distribuce systému `Wotan` je tvořena sadou HTML dokumentů a CGI skriptů. Sada CGI skriptů byla naprogramována a odladěna na webovském serveru Apache verze 1.3 pod operačním systémem Linux. CGI skripty jsou nezávislé na použité distribuci Linuxu a architektuře procesoru. Jedinou nutnou podmínkou pro správný chod aplikace je plná kompatibilita distribuce Linuxu s normou *POSIX*[32] (tzn. musí být k dispozici nástroje pro zpracování textu `awk`, `sed`, `grep`, `tr`, ...).

4.4 Instalace auditního systému Wotan

Instalace auditního systému `Wotan` bude v budoucnu detailně popsána v instalačním manuálu. Tento manuál bude sestaven po skončení testování nástroje v praxi a odstranění všech známých chyb současné verze distribuce auditního systému `Wotan-1.0`. Do finální verze auditního systému `Wotan` bude také nutné zapracovat připomínky a návrhy uživatelů.

[31] System V Interface Definition

[32] Portable Operating System Interface

Tato část textu proto obsahuje pouze nejpodstatnější informace převzaté ze vznikajícího manuálu k auditnímu systému `Wotan`. Je proto možné, že znění některých částí textu bude v budoucnosti přeformulováno nebo bude zcela vypuštěno a nahrazeno textem novým.

4.4.1 Instalace auditního agenta

Systém `Wotan` je určen pro audit v síti unixových počítačů. V těchto sítích se nejčastěji využívá pro usnadnění administrace uživatelských účtů systém `NIS` a pro centrální správu domácích adresářů uživatelů souborový systém `NFS`. Instalace auditního systému `Wotan` plně využívá obou technologií.

Nejprve je nutné před samotnou instalací auditního agenta zřídit uživatelský účet a domácí adresář (můžeme použít příkazy `useradd` nebo `admintool` v závislosti na používané verzi Unixu). Je potřeba zajistit, aby domácí adresář nového uživatele byl dostupný na všech monitorovaných počítačích. Z tohoto důvodu je nejvhodnější zřídit adresář na jednom z hlavních serverů a zajistit jeho následný export pomocí `NFS` na ostatní monitorované počítače. Pokud hlavní server běží pod operačním systémem Sun Solaris 2.x nebo Solaris 7 je vhodné použít `Secure NFS`, pokud je jeho podpora dostupná.

Pro zakládaný uživatelský účet nastavíme například následující hodnoty: pro uživatele zvolíme login name *wotan*, UID[33] nastavíme na hodnotu 7407. Pro tohoto uživatele dále zřídíme speciální skupinu *wotan* s GID[34] rovným 7407. Členem této skupiny bude pouze uživatel *wotan* a nikdo jiný. Je velmi důležité, aby na všech monitorovaných počítačích měl uživatel *wotan* shodné UID a GID. Pro zajištění této podmínky využijeme zmiňovaný systém `NIS`, který umožňuje centrální správu uživatelských účtů a jejich následný export a distribuci v síti.

Pokud chceme provádět správu auditního systému `Wotan` z uživatelské

[33] User IDentifier
[34] Group IDentifier

úrovně, potom uživateli *wotan* přiřadíme login shell a heslo. V opačném případě nastavíme login shell na `/bin/false` a do souboru `/etc/shadow` umístíme do položky *encrypted password* (viz. `man shadow(5)`) hvězdičku. Tím znemožníme komukoliv se na účet uživatele *wotan* přihlásit. Administrace systému `Wotan` je pak možná pouze z účtu privilegovaného uživatele `root` pomocí příkazu `su` *wotan*, který změní UID a GID uživatele `root` na uživatele *wotan*.

Uživateli *wotan* přidělíme domácí adresář. Do takto vytvořeného domácího adresáře uživatele *wotan* nakopírujeme distribuci auditního systému `Wotan` z přiloženého CD pomocí příkazu

```
cp -R /cdrom/Wotan-1.0 /home/wotan
```

Adresář `/cdrom` je místo připojení jednotky CD-ROM, které se liší podle použité verze Unixu. Adresář `/home/wotan` představuje domácí adresář uživatele *wotan*, ve kterém vznikne zadáním výše uvdeného příkazu podadresář `Wotan-1.0`. Příkazem `ls` zjistíme přístupová práva, vlastníka a skupinu u všech souborů a podadresářů v adresáři `Wotan-1.0`. Vlastníkem všech souborů by měl být uživatel *wotan*. Skupina vlastníka by měla být nastavena na skupinu *wotan*. Pokud tomu tak není, v domácím adresáři uživatele *wotan* zadáme příkaz

```
chown -R wotan:wotan Wotan-1.0
```

Tímto příkazem docílíme správného nastavení vlastníka všech souborů a podadresářů v adresáři `Wotan-1.0`. Přístupová práva pro soubory by měla být nastavena u spustitelných souborů na hodnotu 750, u datových souborů na 640 a u podadresářů na 751. Pokud tomu tak není, zadáním příkazu `chmod 750` na spustitelné soubory (umístěné vždy v podadresářích `bin`),

`chmod 640` na datové a další soubory a konečně `chmod 751` na adresáře docílíme požadovaného nastavení. Tyto kontroly je potřeba provést vzhledem k různým nastavením implicitní hodnoty `umask` (masky pro zakládání souborů a adresářů) na různých systémech a vzhledem k přístupovým právům na připojeném CD. Některé unixové systémy připojují souborový systém CD tak, že u všech souborů jsou přístupová práva nastavena na hodnotu 555 (tzn. všechny soubory jsou spustitelné), nebo naopak jsou nastavena na hodnotu 444 (tzn. všechny soubory se chovají jako datové). Při kopírování se tato přístupová práva zachovávají a je potřebné je tedy podle výše uvedených pokynů změnit.

Pokud chceme používat SNMP rozšíření auditního systému `Wotan`, je nutné dodatečně změnit vlastníka a skupinu pro podadresář `Wotan-1.0/database` na vlastníka a skupinu, pod kterým běží webovský server (nejčastěji je to uživatel *www-data*, *www*, *web* nebo *nobody*). Tohoto nastavení dosáhneme zadáním příkazu

```
chown -R www-data:www-data Wotan-1.0/database
```

v domácím adresáři uživatele *wotan*. Při zadání příkazu použijeme správné jméno uživatele, pod kterým je www serveru spouštěn na daném počítači. V posledním kroku zadáme v domácím adresáři uživatele *wotan* příkaz

```
ln -s Wotan-1.0 Wotan
```

Tento příkaz slouží pro vytvoření symbolického odkazu na aktuální verzi auditního systému `Wotan`. To umožňuje používat různé verze auditního systému `Wotan`. Pokud chceme používat jinou verzi distribuce, stačí pouze zrušit starý symbolický odkaz a spustit příkaz `ln` tak, aby vznikl nový odkaz na adresář s novou verzí auditního systému. Tento krok navíc usnadní následnou

konfiguraci auditního systému `Wotan` a jeho pozdější administraci.

V domácím adresáři uživatele *wotan* se nyní nachází nainstalovaná aktuální verze auditního nástroje `Wotan`. V dalším textu bude pro větší přehlednost proměnná `$HOME` označovat domácí adresář uživatele *wotan* a proměnná `$WOTAN` adresář `$HOME/Wotan`. Pro snadnější pochopení funkce a konfigurace auditního systému `Wotan` následuje popis struktury podadresářů a krátké vysvětlení významu jednotlivých souborů v těchto podadresářích. Odkazy jsou uváděny relativně k adresáři `$WOTAN`.

Distribuce auditního systému `Wotan-1.0` má následující adresářovou strukturu:

- `agents/` – tento adresář obsahuje distribuci auditních agentů pro podporované operační systémy a platformy.
- `bin/` – tento adresář obsahuje pomocné programy auditního systému `Wotan`.
- `database/` – v tomto adresáři jsou umístěny centrálně spravované databáze informací. V současné době je dostupná podpora pouze pro databáze modulu SNMP. Ostatní databáze jsou spravované lokálně na jednotlivých monitorovaných počítačích.
- `doc/` – v tomto adresáři je uložena dokumentace auditního systému `Wotan`.
- `etc/` – tento adresář obsahuje všechny globální konfigurační soubory.
- `hosts/` – v tomto adresáři se zřizují podadresáře pro všechny monitorované systémy. Postup přidání nového počítače je popsán dále.
- `html/` – tento adresář obsahuje HTML dokumenty, CGI skripty a další soubory pro uživatelské rozhraní.

- `include/` – v tomto adresáři jsou uloženy hlavičkové soubory nutné pro překlad některých implementačně závislých částí auditního systému `Wotan`.
- `lib/` – tento adresář obsahuje knihovny funkcí.
- `src/` – v tomto adresáři jsou umístěny zdrojové texty implementačně závislých částí auditního systému `Wotan`.

Aby auditní systém `Wotan` správě pracoval, je po jeho instalaci nutná jeho rekonfigurace. V adresáři `$WOTAN/etc` se nalézají všechny hlavní konfigurační soubory. Soubor `Wotan.cfg` obsahuje globální nastavení nutná pro spuštění auditního agenta. Tento soubor je dobře okomentovaný. Pro každou konfigurační možnost je připojen krátký popis této možnosti a jsou uvedeny možné hodnoty, kterých může tato možnost nabývat. Modifikaci tohoto souboru lze provést v jakémkoliv textovém editoru, který ukládá text bez rozšířeného formátování (např. `vi`, `joe` nebo `emacs`). Modifikovat lze pouze hodnoty u jednotlivých možností, nelze však modifikovat jejich symbolická jména. Upravené nastavení se projeví od chvíle, kdy se uloží nová konfigurace.

Konfigurační soubor `Wotan.hosts` obsahuje seznam monitorovaných počítačů. Jeho formát je odvozen od formátu souboru `/etc/hosts`. Pro každý počítač obsahuje soubor `Wotan.hosts` jednu řádku s těmito položkami:

`<jméno počítače (hostname)> <komentář>`

Pokud chceme přidat do seznamu monitorovaných počítačů další počítač, zaneseme nejprve nový záznam do zmíněného souboru. Následně zkopírujeme podadresář z adresáře `$WOTAN/agents/`, který obsahuje verzi auditního agenta určenou pro operační systém a architekturu daného počítače do adresáře `$WOTAN/hosts/<hostname>`. Jméno počítače (*host name*) zjistíme v Solarisu pomocí příkazu `hostname` a v Linuxu pomocí příkazu `hostname -s`.

Po zkopírování patřičné verze agenta pro daný operační systém a hardwarovou platformu je nutné opět překontrolovat vlastníka, skupinu a přístupová práva u všech souborů a podadresářů v nově vzniklém adresáři, zda-li obsahují patřičné hodnoty. Pokud došlo k modifikaci vlastníka, skupinu nebo přístupových práv, je nutné upravit jejich hodnoty podle výše uvedeného popisu.

Jako příklad můžeme uvést instalaci auditního agenta na systému Sun Solaris. Příkaz

```
cp -R $WOTAN/agents/SunOS/Solaris-7/x86 $WOTAN/hosts/quark
```

nainstaluje příslušnou verzi agenta pro operační systém Sun Solaris 7 pro počítač *quark* s procesorem Intel a kompatibilní. Tento příkaz způsobí pouze přenesení příslušných podadresářů a souborů pro danou platformu. Nejdůležitější je přidání záznamu do souboru `Wotan.hosts`, který zařadí počítač mezi monitorované počítače. Pokud se jméno počítače zařadí pouze do souboru `Wotan.hosts` a nezkopíruje se pro něj patřičná verze agenta, bude tento systém ignorován.

Stejnou instalační proceduru použijeme na všech zbývajících počítačích. Pro zpřehlednění dalšího zápisu zavedeme novou proměnnou `$AGENT`, která označuje adresář `$WOTAN/hosts/<hostname>`.

V tomto kroku zbývá ještě překontrolovat lokální konfigurační soubor `$AGENT/etc/Wotan.cfg`, který ovlivňuje nastavení agenta na konkrétním počítači. Ve většině případů to není potřeba. Implicitní nastavení téměř vždy funguje správně. K ověření správnosti instalace využijeme vestavěný test auditního agenta. Po zadání příkazu

```
$AGENT/bin/wotan -d
```

proběhne test a administrátor je upozorněn na případné problémy. Většina problémů je způsobena špatným nastavením cesty k spustitelným souborům v konfiguračním souboru `$AGENT/etc/Wotan.cfg`, nebo absencí externích programů. Používané externí programy jsou součástí standardní distribuce odpovídající verze Unixu, nebo se jedná o volně dostupné nástroje. Stačí je tedy pouze nainstalovat do systému a znovu spustit vestavěný test pro ověření korektnosti instalace auditního agenta.

V posledním kroku se provede na všech počítačích instalace souboru pro démona `cron`, který zajišťuje periodické spouštění auditního systému `Wotan`. Tento soubor se nachází v adresáři `$AGENT/etc` a jmenuje se `Wotan.crontab`. Soubor lze opět upravit podle vlastních potřeb v libovolném textovém editoru. Vlastní instalace souboru se provede příkazem

```
crontab $AGENT/etc/Wotan.crontab
```

na všech monitorovaných počítačích. Pokud vše proběhlo správně, je instalace auditního systému `Wotan` dokončena. Nyní zbývá nainstalovat uživatelského rozhraní, které dovoluje nashromážděná data vyhodnocovat.

4.4.2 Instalace uživatelského rozhraní

Jak již bylo uvedeno, uživatelské rozhraní je tvořeno sadou CGI skriptů a HTML dokumentů. Instalaci tohoto rozhraní vyžaduje počítač s operačním systémem Linux a webovským serverem Apache verze 1.3.

Zkopírování všech souborů nutných pro správnou funkci grafického uživatelského rozhraní proběhlo již při instalaci auditního agenta. Soubory se nalézají v adresáři `$WOTAN/html`. Nyní je potřeba správně umístit tyto soubory do adresáře, ve kterém je nalezne webovský sever. V případě severu `Apache` je to nejčastěji adresář `$HOME/public_html`.

Pro usnadnění instalace je k dispozici instalační skript nazvaný

`wotan.install.gui`. Tento skript provede celou instalaci automaticky za předpokladu, že je správně nainstalována distribuce auditního agenta systému `Wotan`, viz. odstavec 4.4.1. V domácím adresáři uživatele *Wotan* stačí zadat příkaz

```
$WOTAN/bin/wotan.install.gui
```

Pokud je pro vystavované HTML dokumenty a CGI skripty použit jiný adresář, než `$HOME/public_html`, lze instalačnímu skriptu tento adresář zadat jako parametr. Pro ilustraci předpokládejme, že dokumenty určené pro vystavení na webovském serveru jsou v adresáři `$HOME/www`. Instalace se potom provede následujícím příkazem

```
$WOTAN/bin/wotan.install.gui --directory www
```

Instalační skript použije pro instalaci adresář `$HOME/www`.

Během instalace je uživateli položeno několik doplňujících otázek, na které odpovídá stiskem kláves `y` nebo `n` a následně klávesou `Enter`. Tyto otázky slouží pro ověření správnosti přednastavených voleb automatické instalace.

Pro zjednodušení dalšího zápisu zavedeme proměnou `$WWW`, která označuje adresář, ve kterém se nacházejí naistalované HTML soubory a CGI skripty (např. `$HOME/public_html`).

V následujícím kroku instalace je nutné prověřit nastavení konfigurace uživatelského rozhraní. Konfigurační soubory řídí nejen nastavení uživatelského rozhraní, ale slouží pro zadání referenčních hodnot. Tyto hodnoty se využívají při vyhodnocování nashromážděných dat. První soubor nazvaný `Wotan.Audit.cfg` ovlivňuje nastavení uživatelského rozhraní a referenčních hodnot pro moduly v režimu *off–line*. Tento soubor se nachází v adresáři `$WWW/cgi-etc`. Je velmi podrobně okomentován. U každé konfigurační mož-

nosti je opět připojen její krátký popis a jsou zde uvedeny možné hodnoty, kterých může tato možnost nabývat. Modifikaci tohoto souboru lze provést v jakémkoliv textovém editoru. Upravené nastavení se projeví od chvíle, kdy se tento soubor uloží.

Pokud chceme používat *on–line* monitor, který využívá rozšíření MIB Linuxu, je nutné nejprve přeložit SNMP managera. V adresáři `$WOTAN/bin` je uložen soubor `Makefile` pro řízení překladu managera, který je součástí distribuce auditního systému `Wotan-1.0`. Sestavení binárního souboru provedeme zadáním následujícího příkazu v adresáři `$WOTAN/bin`

```
make all
```

Po úspěšném překladu se v tomto adresáři vytvoří SNMP manager s názvem `wotan.snmp`. Tento program poskytuje základní funkce pro manipulaci s objekty v MIB na vzdálených systémech.

Instalaci *on–line* monitoru dokončíme úpravou konfiguračního souboru `Wotan.SNMP.cfg`, který se nachází v adresáři `$WWW/cgi-etc`. Úprava tohoto souboru je opět velice jednoduchá, protože je dobře okomentovaný a jsou v něm podrobně vysvětleny všechny možné volby a nastavení.

Pokud se během instalace nevyskytla chyba, je instalace auditního systému `Wotan-1.0` dokončena. V posledním kroku prověříme funkčnosti nové instalace pomocí webovského prohlížeče. Pokud je vše v pořádku, auditní systém `Wotan` začne vyhodnocovat nasbíraná data. V opačném případě je nutné znovu projít všechny uvedené konfigurační soubory a přesvědčit se, zda obsahují správná a platná nastavení.

5 Implementace

Tato kapitola se věnuje popisu implementace auditního systému `Wotan` a popisuje strukturu zdrojových kódů.

K implementaci auditního agenta a CGI skriptů byl zvolen příkazový interpret `TENEX c-shell` (`tcsh`). Pro hardwarově závislé částí byl použit jazyk C. Systém byl implementován a odladěn pro překladače jazyka C `gcc`, jenž je dostupný pro většinu unixových distribucí. Jako základní platformu pro vývoj a ladění všech aplikací jsem zvolil Sun Solaris 7 na platformě Intel. Pro vývoj a ladění SNMP managera byl použit Debian Linux 2.1.

5.1 Auditní agent

Auditní agent je samostatná aplikace, která plně využívá standardních unixových nástrojů. Funkcí auditního agenta je shromažďování důležitých dat získaných z operačního systému. Jeho činnost lze popsat následujícím způsobem:

```
StartAgent
ParseCommandLineArgs
LoadInitModule
     StartAudit
          LoadAccountModules
          LoadFSModules
          LoadNetModules
          LoadSystemModules
     StopAudit
StopAgent
```

Funkce `StartAgent` slouží pro inicializaci vnitřních proměnných auditního agenta. Současně tato funkce načítá konfigurační soubor, který ovlivňuje chování auditního agenta na konkrétním počítači. Funkce dále do standard-

ního systémového auditního deníku nástroje `syslogd` (viz. odstavec 3.5.4) zaznamená čas a datum, kdy byl auditní agent `Wotan` spuštěn. Tento záznam může například administrátora systému upozornit na skutečnost, že systém je monitorován neoprávněnou osobou, nebo dochází k monitorování systému mimo naplánovaný čas.

Funkce `ParseCommandLineArgs` třídí a vyhodnocuje parametry, které byly předány auditnímu agentovi z příkazové řádky. Pokud byl dán agentovi pokyn pro spuštění vestavěného testu, je provedena funkce `RunDependencyTest`. Tato funkce nebude více popisována, protože slouží pouze pro ověření správnosti instalace a na samotnou činnost auditního agenta nemá přímý vliv.

Funkce `LoadInitModule` zavede inicializační modul a vykoná v něm uložené příkazy. Tento modul je zaváděn vždy jako první. Aktuální verze inicializačního modulu obsahuje pouze příkaz pro nastavení časové známky auditního běhu. V budoucnu do něj mohou být umístěny další příkazy, jejichž vykonání bude nutné během startu auditního agenta.

Funkce `StartAudit` vyhodnotí podle nastavení v konfiguračním souboru pořadí zavádění jednotlivých auditních modulů. Tato funkce vytvoří seznamy modulů (`MODULES_LIST`), které se následně procházejí a zavádějí se v nich zařazené moduly. Konfigurační soubor ovlivňuje pořadí modulů pouze v rámci jedné skupiny modulů. Pořadí skupin je pevně stanoveno a nedá se změnit. Skupiny modulů jsou v současné verzi auditního agenta čtyři:

- Skupina **ACCOUNT** – slouží pro audit účtů oprávněných uživatelů, skupinových účtů a uživatelské aktivity v systému.
- Skupina **FS** – slouží pro audit využití diskových svazků a souborových systémů.
- Skupina **NET** – slouží pro audit síťového subsystému.
- Skupina **SYSTEM** – slouží pro audit systémových prostředků.

Skupina modulů	**Externí používané programy**
ACCOUNT	id w who
FS	df mount procinfo
NET	arp ifconfig netstat rarp route
SYSTEM	bogomips dmesg free iostat ipcs lsmod modinfo procinfo prtconf psrinfo ps swap uptime vmstat

Tabulka 1: Externí programy používané auditními moduly

Pro každou skupinu modulů je tedy následně k dispozici jeden seznam: `ACCOUNT_MODULES_LIST`, `FS_MODULES_LIST`, `NET_MODULES_LIST` a `SYSTEM_MODULES_LIST`. Pro zavádění modulů v rámci jednotlivých seznamů jsou k dispozici funkce `LoadAccountModules`, `LoadFSModules`, `LoadNetModules` a `LoadSystemModules`. Bylo nutné implementovat pro každý seznam modulů jinou zaváděcí funkci vzhledem k tomu, že moduly v rámci jednotlivých skupin mají jinou vnitřní strukturu. Jejich vnitřní struktura je plně podřízena jejich funkci a je navržena pro maximálně efektivní vykonávání příkazů.

Funkce pro zavádění modulů se nejprve pokusí vyhledat jednotlivé moduly, které jsou uvedeny v seznamu a následně vykonají všechny uvedené příkazy v nalezených modulech. Moduly jsou na sobě navzájem nezávislé. Pokud by tedy došlo k poškození nebo odmazání jednoho modulu, auditní systém přesto bude nadále pokračovat ve své činnosti.

Jednotlivé moduly využívají při shromažďování dat standardní nástroje pro administraci Unixu. Výčet těchto nástrojů je uveden výše v tabulce č. 1. Pro zpracování výstupu jednotlivých programů je použit jazyk pro zpracování textu `awk`. Výsledek zpracování je uložen ve zformátovaném tvaru do souborů. Tyto soubory jsou umístěny v podadresářích podle příslušnosti k dané

skupině modulů. Pro jednorázové vyhodnocení získaných dat lze například využít rozhraní CGI dodávané s distribucí auditního nástroje `Wotan`, jehož implementace bude popsána v části 5.2. Pro dlouhodobé vyhodnocování, popřípadě sestavení grafů, lze například použít nástroj `mrtg`[35]. Tento nástroj umožňuje generování HTML dokumentů, které obsahují grafy uložené jako obrázky ve formátu GIF[36].

Funkce `StopAudit` ukončí auditní běh a zruší jednotlivé seznamy modulů. Činnost auditního agenta je ukončena voláním funkce `StopAgent`, která provede záznam času a data ukončení auditního běhu do systémového auditního deníku nástroje `syslogd` a následně provede úklidové akce.

Auditní agent systému `Wotan` nevyžaduje pro svou činnost propůjčení přístupových práv privilegovaného uživatele `root`. To je podmínka, kterou jsem si dal ve fázi návrhu nově vznikajícího auditního systému, viz. část 4.2. Tuto podmínku se mi podařilo při implementaci beze zbytku splnit.

5.1.1 Zdrojové kódy auditního agenta

Jak již bylo uvedeno, auditní agent systému `Wotan` je implementován v příkazovém interpretu `TENEX c-shell`. Z tohoto důvodu není nutné provádět překlad systému při jeho instalaci. Interpret kompiluje a provádí skripty až ve chvíli jejich spuštění. Soubory, které obsahují zdrojové kódy jsou tedy současně i binárními spustitelnými soubory. Interpret `TENEX c-shell` jsem zvolil pro jeho dostupnost na většině unixových platforem. Tento interpret vedle své dostupnosti navíc poskytuje příkazy vhodné pro práci s textem. Pokud se ale systém osvědčí v praxi, bude vhodné provést jeho implementaci v programovacím jazyce `Perl`. `Perl` například odstraní nutnost implementace různých funkcí pro zavádění modulů v rámci jednotlivých skupin a sníží zatížení systému během auditu. Současná verze auditního agenta, implemen-

[35] Multi Router Traffic Grapher
[36] Graphics Interchange Format

tovaná v `TENEX c-shell`, slouží proto spíše jako základ pro budoucí rychlejší a stabilnější verze.

Kód auditního agenta je uložen v souboru `$AGENT/bin/wotan`. Tento zdrojový kód je ve velké většině shodný pro všechny platformy, které auditní systém `Wotan` podporuje. Při implementaci však vyvstaly některé problémy, které si vyžádaly úpravu zdrojových kódů a jejich následné rozdělení podle jednotlivých platforem. Tyto problémy jsou dány drobnými rozdíly v různých verzích operačních systémů a programového vybavení, které se s těmito verzemi operačních systémů dodává.

Pro svoji správnou činnost vyžaduje auditní agent moduly, které provádějí vlastní audit systému. Tyto moduly jsou uloženy v adresáři `$AGENT/modules`. Moduly jsou rozděleny do skupin, které bylu popsány v předchozí části 5.1. Zdrojový kód těchto modulů je ve velké většině implementován v jazyce pro zpacování textu `awk`. Moduly fungují jako filtry formátující výstupní informace získané od externích programů. Všechny zdrojové kódy modulů jsou důkladně okomentovány a jsou přehledné.

5.2 Uživatelské rozhraní

Grafické uživatelské rozhraní je implementováno sadou HTML dokumentů a CGI skriptů určených pro webovský server `Apache 1.3` pro operační systém Linux. Ukázky uživatelského rozhraní jsou zobrazeny v příloze A.1 obr. 1 až obr. 3. CGI skripty jsou, obdobně jako auditní agent, implementovány v interpretu `TENEX c-shell`. Pro formátování výstupu v jazyce HTML byl opět využit jazyk pro zpracování textu `awk`.

5.2.1 Zdrojové kódy uživatelského rozhraní

CGI skripty se při instalaci ukládají do adresáře `$WWW/cgi-bin`. Tyto skripty realizují nejen samotné interaktivní uživatelské rozhraní auditního systému `Wotan`, ale také slouží pro samotný audit. Skripty jsou rozděleny podle

funkce do několika skupin. První skupinu tvoří skripty, jejichž název začíná `Wotan.Audit.*`. Tato skupina skriptů slouží k vyhodnocování dat, které nashromáždil auditní agent. Při vyhodnocování se uplatňují referenční hodnoty, o nichž se zmiňuji v odstavci 4.2, viz. obr. 5 v příloze A.1. Druhou skupinu tvoří skripty, jejichž název začíná `Wotan.Monitor.*`. Tyto skripty slouží pro *on–line* monitorování sítě, viz. obr. 7 v příloze A.1. Poslední skupinu tvoří skripty, jejichž jméno začíná `Wotan.SNMP.*`. Skripty v rámci této skupiny realizují vyhodnocování dat získaných od SNMP managera `wotan.snmp`, viz. obr. 6 v příloze A.1.

Pro zpřehlednění zdrojových kódů jsem často používané části kódu uložil do několika samostatných knihoven. Tyto knihovny slouží pro práci s CGI rozhraním (knihovna `Wotan.CGI.lib`), pro uložení a údržbu dat získaných SNMP managerem (knihovna `Wotan.DB.lib`) a pro manipulaci s objekty v MIB na monitorovaných linuxových systémech (knihovna `Wotan.SNMP.lib`). Při instalaci se knihovny ukládají do adresáře `$WWW/cgi-lib`.

Pro překlad SNMP managera určeného pro monitorování rozšíření MIB pro Linux je potřebná knihovna CMU[37] SNMP (ve verzi 3.5 a vyšší). Překlad a instalace SNMP managera byla popsána v odstavci 4.4.2. Pro správnou činnost *on–line* síťového monitoru a příslušných CGI skriptů je nutné mít v systému nainstalovány nástroje `tcpblast` (ve verzi FreeBSD/rzm 961003 nebo vyšší) a nástroj `nmap` (ve verzi 2.2 nebo vyšší). Tyto nástroje jsou zahrnuty do distribuce auditního nástroje `Wotan` a jsou uloženy v adresáři `contrib` na přiloženém CD. Součástí jejich distribuce je přesný popis pro překlad a instalaci do systému.

5.3 Testování systému

Auditní systém `Wotan` byl testován a laděn na interní počítačové síti ÚHKT. K testování byl využit centrální server s operačním systémem Sun Solaris 2.6

[37] Carnegie Mellon University

na platformě Sparc, a osobní počítače s operačním systémem Debian Linux 2.1 a Caldera OpenLinux 1.2. Obě distribuce Linuxu používaly jádro ze stabilní řady 2.0.x. Pro otestování správné funkce CGI skriptů v praxi, byl použit webovský server Apache 1.3.3 dodávaný standardně s distribucí Debian Linux.

Uživatelské rozhraní bylo testováno ve webovském prohlížeči *Netscape Communicator* ve verzi 4.x pro Solaris, Linux a MS Windows. Dále bylo uživatelské rozhraní odzkoušeno v prohlížeči *Internet Explorer* ve verzi 5.0. Rozhraní jsem testoval i v řádkovém prohlížeči *Lynx* ve verzi 2.8, který je velmi oblíben v unixové komunitě. K formátování dat v HTML dokumentech se nejčastěji využívají tabulky. Bohužel, webovský prohlížeč *Lynx* nezvládl zobrazení větších tabulek správně. Do příští verze systému hodlám proto rozdělit HTML dokumenty a CGI skripty do dvou skupin. První skupina bude určena pro zobrazování k grafických prohlížečích (*Netscape*, *Amaya*, *IE*, ...), druhá skupina pak bude optimalizována pro zobrazení v textově orientovaných prohlížečích. K verifikaci samotného HTML kódu jsem využil webovský prohlížeč a editor *Amaya* vyvíjený WWW konsorciem, který umožňuje důkladné otestování syntaxe jazyka HTML.

Ve fázi alfa a hlavně beta testů systému `Wotan` jsem provedl krátké porovnání funkčnosti a rychlosti s komerčním systémem X^{ni} firmy Fastlane Software Systems, Inc. Déledobější porovnání nebylo možné, jednak proto, že poskytnutá trial license pro systém X^{ni} vypršela za 7 dní a také proto, že testování, odladění a odstranění chyb v systému `Wotan` bylo časově velmi náročné.

6 Závěr

Systémů, které provádějí audit unixové sítě, bylo popsáno a navrženo mnoho. Existují velmi kvalitní komerční produkty, jejichž cena může dosáhnout až několik desítek tisíc dolarů. Tyto systémy jsou velice sofistikované, ale jsou mimo finanční možnosti převážné většiny zdravotních a školských zařízení, s výjimkou těch největších. Existuje také velké množství volně dostupných systémů. Tyto systémy jsou však často speciálně navrženy pro konkrétní organizace, nebo pokrývají pouze určitou část auditu či monitorování systému. Z těchto uvedených důvodů jsou tyto auditní a monitorovací systémy ve většině případů pro malé a střední organizace s omezenými finačními prostředky nedostupné nebo neúčelné.

Často však i malá a střední zdravotnická zařízení (mezi která patří také ÚHKT) uchovávají velké množství velmi citlivých dat. Právě pro takovéto případy jsem navrhl a implementoval tento auditní a monitorovací systém, který sice neposkytuje všechny možnosti jako velké komerční nebo volně dostupné systémy, ve spojení s dalšími ochranými prvky je však schopen za nevelké vložené náklady poskytnout poměrně vysoký stupeň zabezpečení sítě, což odpovídá kritériu ekonomické efektivity (maximální výstupy při minimálních vstupech).

Výhodou systému, vytvořeného v rámci mé diplomové práce, je jeho jednoduchost. Tato jednoduchost mu umožňuje snadné rozšíření o různé funkce a moduly poskytující nové služby, které původní auditní systém nepodporoval, ale které si vyžádali uživatelé nebo podmínky provozu. Další výhodou je vzájemná nezávislost mezi jednotlivými moduly, což umožňuje uživateli si přizpůsobit systém dle vlastních požadavků.

Během budování auditního a monitorovacího systému jsem narazil na vážnější problémy při výběru vhodných standardních nástrojů, které by bylo možné využít. Některé nástroje vyžadovaly například pro svou činnost pro-

půjčení přístupových práv privilegovaného uživatele. To bylo ale v rozporu s podmínkou, kterou jsem si dal při návrhu architektury nově vznikajícího auditního systému.

Operační systém Unix obsahuje několik standardních administrátorských nástrojů, ale žádný z nich není pro účely komplexního auditu optimální. O výhodách a nevýhodách těchto nástrojů jsem se zmínil v kapitole 3. Během práce na budovaném auditním systému jsem měl možnost srovnání s volně dostupnými nebo komerčními implementacemi některých auditních systémů pro Unix.

Konfigurace a vytváření pravidel pro audit unixových systémů není jednoduché. Z tohoto důvodu se hotový auditní systém velmi špatně testuje a prakticky neexistuje možnost úplného otestování všech jeho částí. Částečně lze nový auditní systém prověřit porovnáním s implementací jiného auditního systému, ale toto testování není kompletní.

Pro prověření všech možných způsobů selhání činnosti nebo nedostupnosti prostředků sítě unixových počítačů, bych musel provést velký počet různorodých testů. To však není reálné, neboť nikdy nemohu do těchto testů zahrnout všechny možné způsoby selhání sítě nebo výskyt nenadálých situací, které chod sítě unixových počítačů ohrožují.

Proto jsem pro testování velkého seznamu možných způsobů selhání činnosti unixové sítě použil metodu pokusu a omylu - provedl jsem simulaci určitého počtu možných reálných ohrožení. Testování a doladění auditního systému je dlouhodobá záležitost. Během této doby je síť vystavuje potenciálně velkému riziku, neboť získaná a vyhodnocená data nemusí dávat přesné informace o tom, co se na síti děje.

I přes tyto problémy s testováním je auditní systém funkční a poskytuje určitou úroveň zabezpečení vnitřní sítě ÚHKT. V budoucnu je možno, v případě rozšiřování interní sítě ÚHKT, využít pro zvýšení jejího zabezpečení kombinaci více auditních systémů, které jsou určeny pro rozsáhlejší instalace.

Po zprovoznění auditního systému se budu zabývat propojením jeho činnosti na další interní bezpečnostní opatření. Jako příklad mohu uvést provázání auditního systému a firewallu, jehož implementace byla součástí mé bakalářské práce. Velkou pozornost bude nutno věnovat také dalšímu seznámení uživatelů s bezpečnostními principy pro práci v počítačové síti. Je známo, že 80% všech neautorizovaných průniků a útoků je provedeno z vnitřní sítě. Již dnes probíhají na danou problematiku rozsáhlé diskuse mezi vedením ústavu, správou počítačové sítě a uživateli.

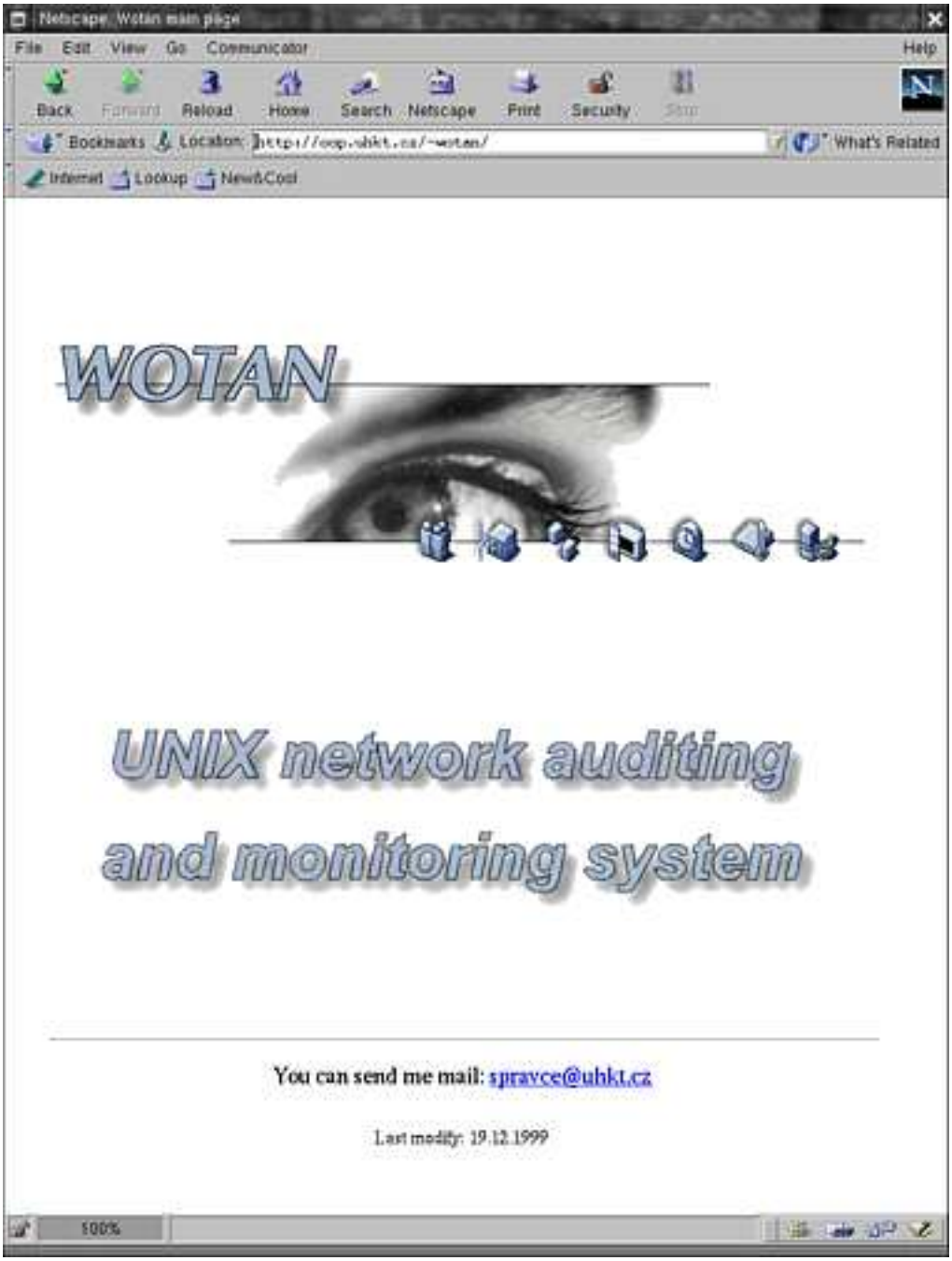
Pokud se auditní systém osvědčí v praxi, bude vhodné provést jeho implementaci v programovacím jazyce Perl. O výhodách implementace auditního systému v Perlu jsem se zmínil v odstavci 5.1.1. Dále bude nutné zavést ochranné prvky pro přenos dat po síti od auditního agenta na webovský server, na kterém se provádí jejich vyhodnocování. Stejně tak bude nutné zabezpečit přenos informací z webovského serveru, na kterém jsou uloženy CGI skripty realizující uživatelské rozhraní auditního systému, a webovským prohlížečem administrátora systému. Pro jazyk Perl existují knihovny, které umožňují použít například `SSL`[38] pro šifrovaný přenos dat po síti, nebo `S-HTTP`[39] pro zabezpečený přenos dat mezi serverem a klientem při použití protokolu `HTTP`. Další možnou úpravou stávajícího auditního systému je podpora většího počtu distribucí Unixu pro různé platformy.

[38] Secure Socket Layer

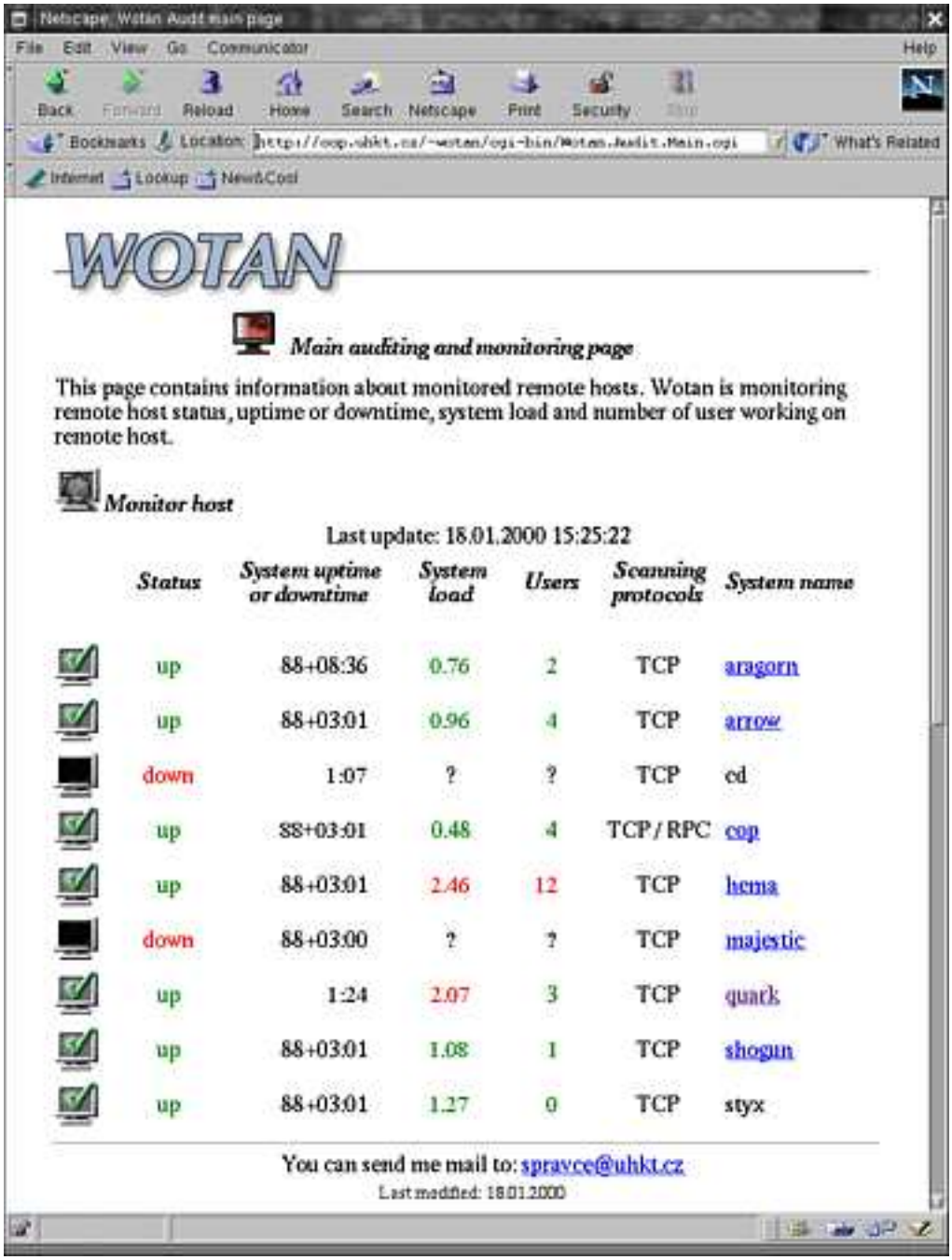
[39] Secure Hypertext Transport Protocol

A Přílohy

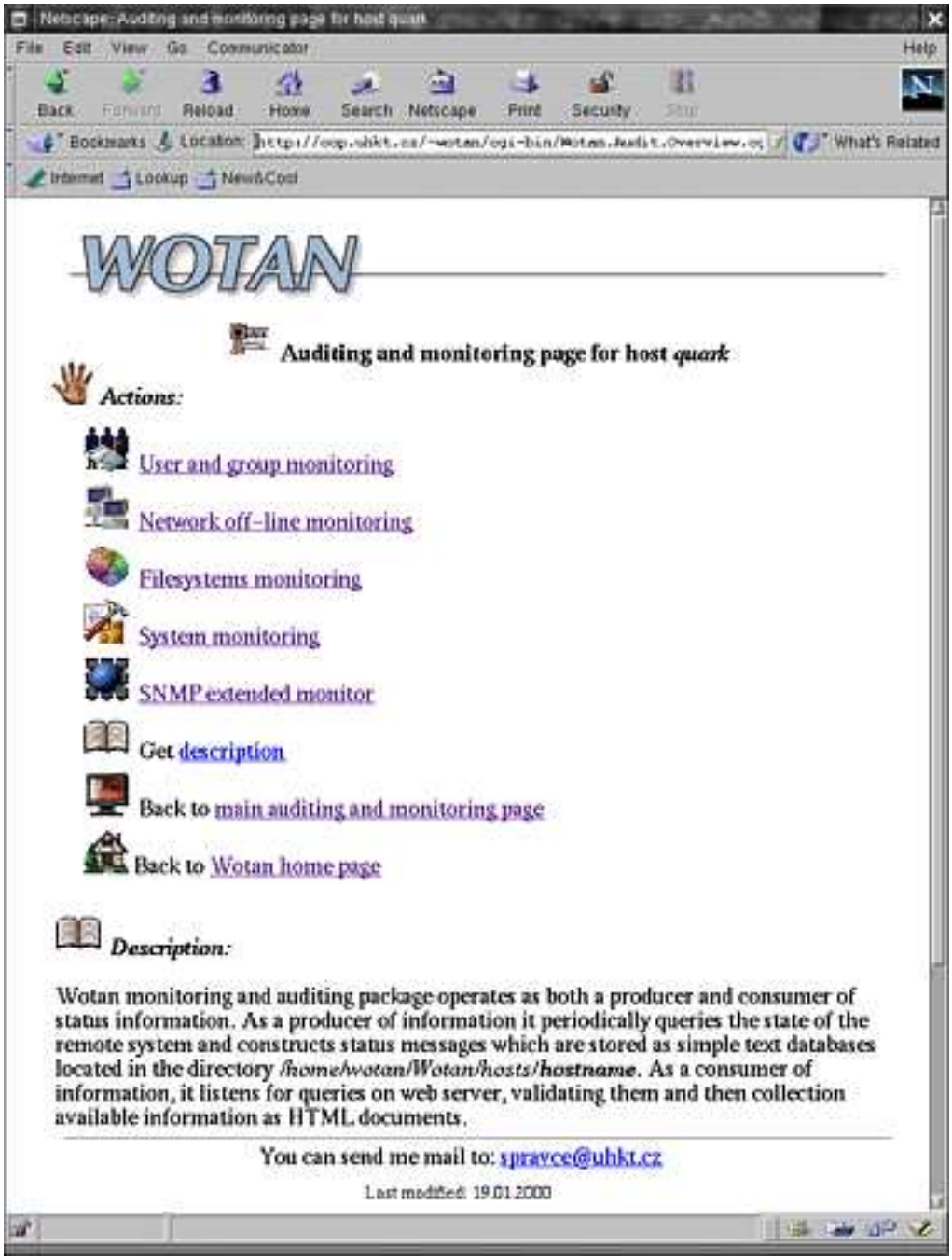
A.1 Obrazová příloha



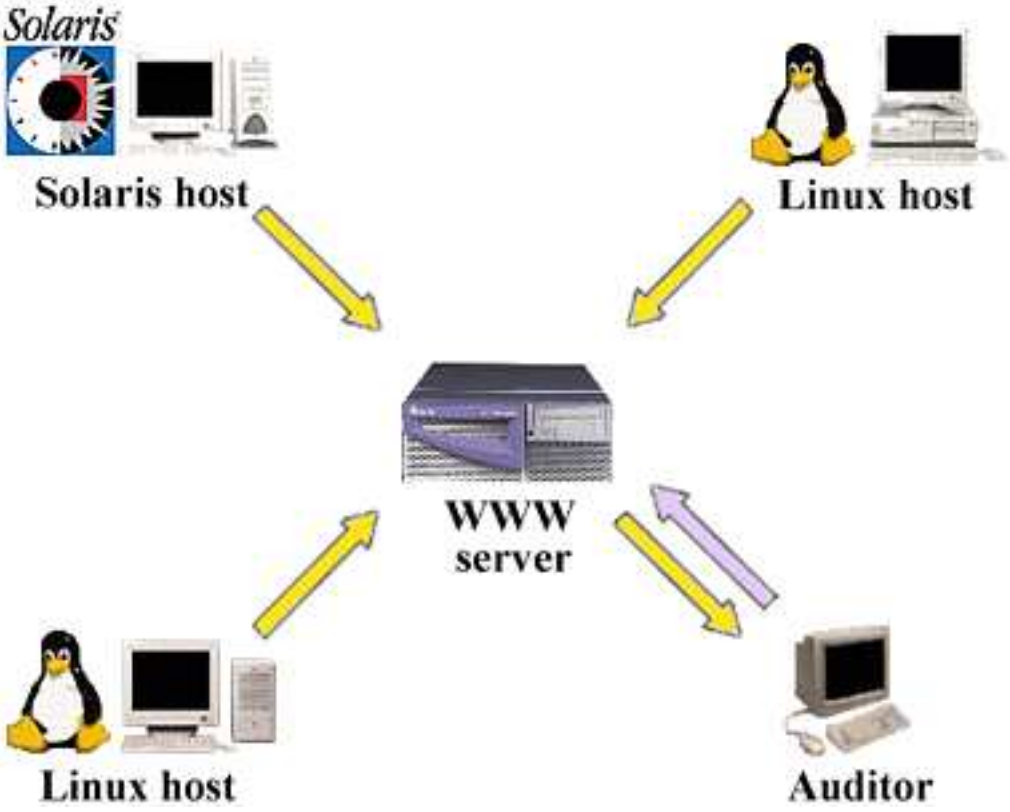
Obr. 1: Úvodní stránka systému Wotan



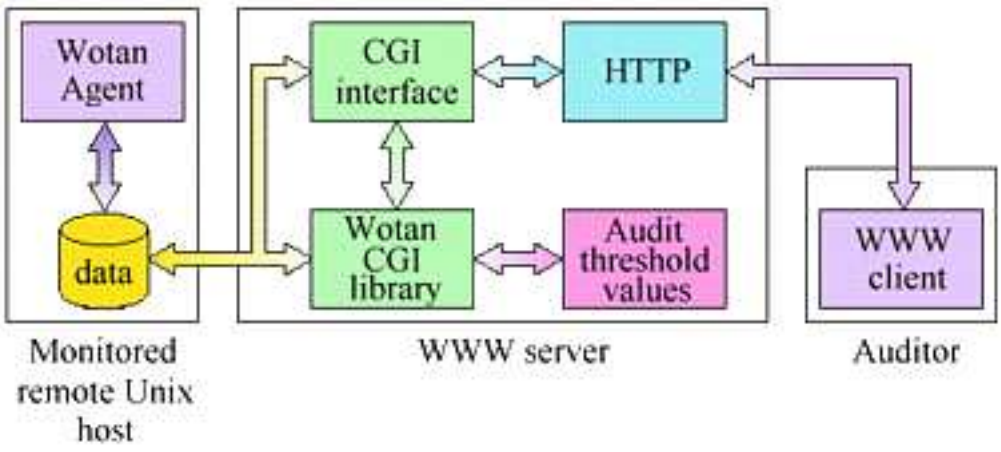
Obr. 2: Hlavní monitorovací a auditní stránka



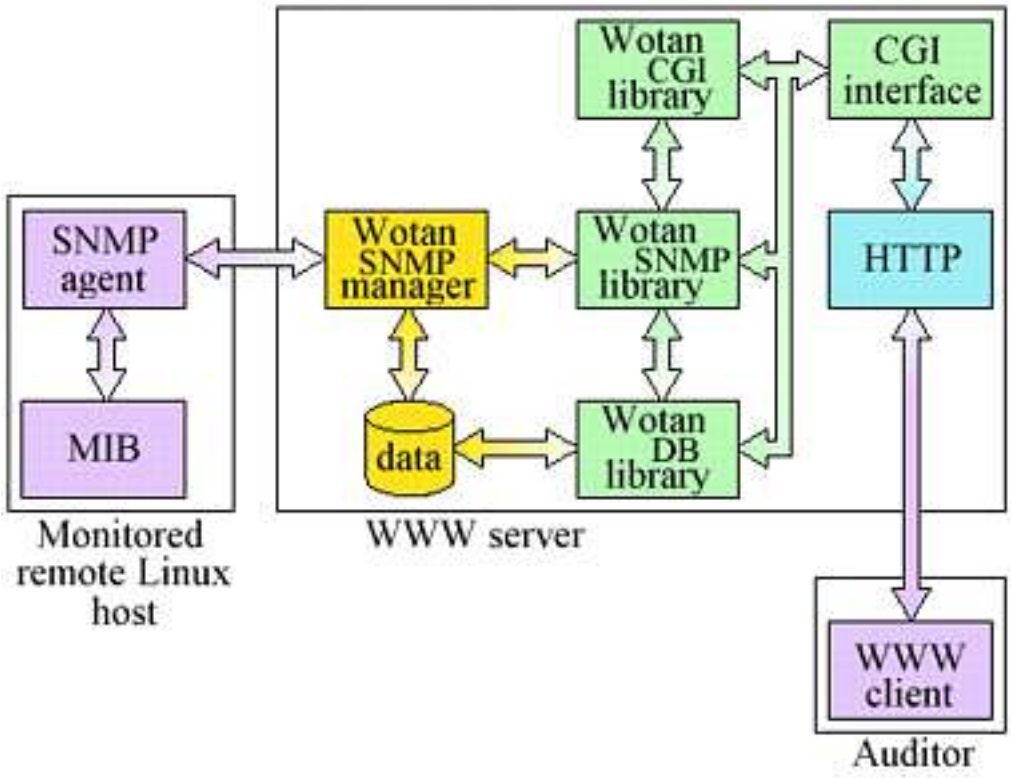
Obr. 3: Výčet možných akcí pro sledovaný počítač



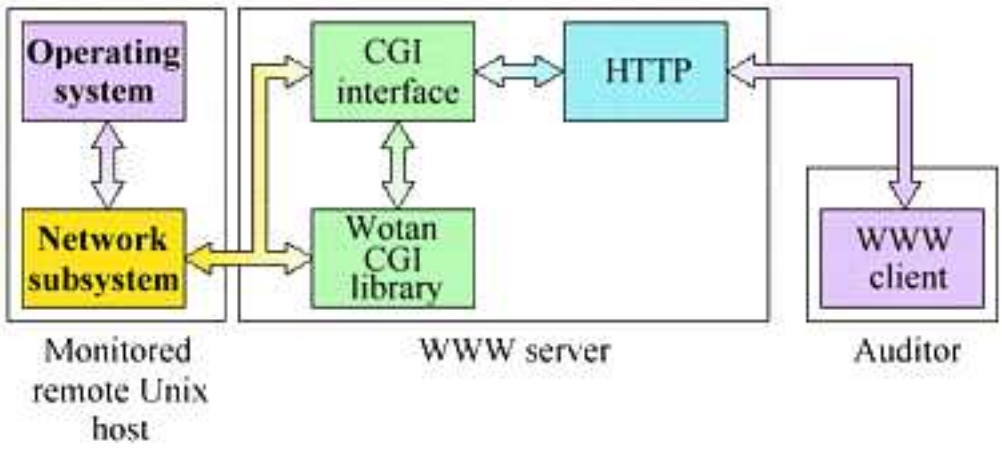
Obr. 4: Princip činnosti systému Wotan



Obr. 5: Architektura auditního agenta



Obr. 6: Architektura on–line SNMP monitoru



Obr. 7: Architektura on–line síťového monitoru

A.2 Příklad konfiguračního souboru

```
#
# Wotan.Audit.cfg v0.3
# (c)1999 Marek Uher, Marek.Uher@uhkt.cz
#
# Configuration file for audit CGI module
#

#
# Begin of configuration
#
if ("$?AUDITCONF" == "0") then
set AUDITCONF

#
# You can edit value of variables
# (! DO NOT EDIT names of variables !)
#

#
# Set Wotan home directory
#
set WOTAN_HOME = "/home/wotan/Wotan"

#
# Wotan audit administrator e-mail
#
set AUDIT_ADMIN = "spravce@uhkt.cz"

#
# Web site (URL) with Wotan AUDIT HTML relating images
#
set AUDIT_IMG_URL = "http://cop.uhkt.cz/~wotan/html/Images"

#
# Web site (URL) with Wotan AUDIT HTML documents
#
set AUDIT_DOC_URL = "http://cop.uhkt.cz/~wotan/html"
```

```
#
# Web site (URL) with Wotan AUDIT CGI binaries
#
set AUDIT_CGI_URL = "http://cop.uhkt.cz/~wotan/cgi-bin"

#
# Network exploration tool and security scanner
# (Compatibile with nmap v.2.2.x)
#
set NETWORK_SCANNER = "/usr/local/bin/nmap"

#
# RPC timeout (in seconds)
#
set WOTAN_RPC_TIMEOUT = "1"

#
# Network throughput timeout (in seconds)
#
set WOTAN_NET_TIMEOUT = "10"

#
# Remote host maximum load
#
set WOTAN_MAX_LOAD = "3"

#
# Remote host maximum user sessions
#
set WOTAN_MAX_USERS = "25"

#
# Remote host maximum usage disk capacity
# (value of variable is in %, <= 100)
#
set WOTAN_MAX_CAPACITY = "85"

#
# End of configuration
#
endif
```

A.3 Obecná veřejná licence GNU

GNU general public licence – český překlad

Verze 2, červen 1991

A.3.1 Preambule

Licence pro většinu programového vybavení jsou navrženy tak, že vám odebírají právo jeho volného sdílení a úprav. Smyslem Obecné veřejné licence GNU je naproti tomu zaručit volnost sdílení a úpravy volného programového vybavení – pro zajištění volného přístupu k tomuto programovému vybavení pro všechny jeho uživatele. Tato Obecná veřejná licence GNU se vztahuje na většinu programového vybavení nadace Free Software Foundation a na jakýkoli jiný program, jehož autor se přikloní k jejímu používání. (Některé další programové vybavení od Free Software Foundation je namísto toho pokryto Obecnou knihovní veřejnou licencí GNU.) Můžete ji rovněž použít pro své programy.

Pokud mluvíme o volném programovém vybavení, máme na mysli volnost, nikoliv cenu. Naše Obecná veřejná licence je navržena pro zajištění toho, že můžete volně šířit kopie volného programového vybavení (a účtovat si za tuto službu, pokud chcete), že obdržíte zdrojový kód anebo jej můžete získat, pokud chcete, že můžete tento software měnit nebo jeho části použít v nových programech; a že víte, že tyto věci smíte dělat.

Abychom mohli vaše práva chránit, musíme vytvořit omezení, která zakáží komukoli vám tato práva odepírat nebo vás žádat, abyste se těchto práv vzdal. Tato omezení se promítají do jistých povinností, kterým musíte dostát, pokud šíříte kopie dotyčného programového vybavení anebo ho modifikujete.

Například, šíříte-li kopie takového programu, ať již zdarma nebo za poplatek, musíte poskytnout příjemcům všechna práva, která máte sám. Musíte zaručit, že příjemci rovněž dostanou anebo mohou získat zdrojový kód. A musíte jim ukázat tyto podmínky, aby znali svá práva.

Vaše práva chráníme ve dvou krocích: (1) autorizací programového vybavení, a (2) nabídkou této licence, která vám dává právoplatné svolení ke kopírování, šíření a modifikaci programového vybavení.

Kvůli ochraně každého autora i nás samotných chceme zaručit, aby každý chápal skutečnost, že pro volné programové vybavení nejsou žádné záruky. Je-li programové vybavení někým jiným modifikováno a posláno dále, chceme, aby příjemci věděli, že to, co mají, není originál, takže jakékoliv problémy vnesené jinými se neodrazí na reputaci původních autorů.

Konečně, každý volný program je neustále ohrožen softwareovými patenty. Přejeme si zamezit nebezpečí, že redistributoři volného programu obdrží samostatně patentová osvědčení a tím učiní program vázaným. Abychom tomu zamezili, deklarovali jsme, že každý patent musí být buď vydán s tím, že umožňuje volné užití, anebo nesmí být vydán vůbec.

Přesná ustanovení a podmínky pro kopírování, šíření a modifikaci jsou uvedeny dále.

A.3.2 Ustanovení a podmínky pro kopírování, distribuci a modifikaci

§0

Tato licence se vztahuje na kterýkoliv program či jiné dílo, které obsahuje zmínku, umístěnou v něm držitelem autorských práv, o tom, že dílo může být šířeno podle ustanovení Obecné veřejné licence GNU. V dalším textu znamená „Program“ každý takový program nebo dílo a „dílo založené na Programu“ znamená buď Program samotný anebo každé jiné dílo z něj odvozené, které podléhá autorskému zákonu: tím se míní dílo obsahující Program nebo jeho část, buď doslovně anebo s modifikacemi, popřípadě v překladu do jiného jazyka. (Nadále je překlad zahrnován bez omezení pod pojmem „modifikace“.) Každý uživatel licence je označován jako „vy“.

Jiné činnosti než kopírování, šíření a modifikace nejsou pokryty touto licencí; sahají mimo její rámec. Akt spuštění programu není omezen a výstup z Programu je pokryt pouze tehdy, jestliže obsah výstupu tvoří dílo založené na Programu (nezávisle na tom, zda bylo vytvořeno činností Programu). Posouzení platnosti předchozí věty závisí na tom, co Program dělá.

§1

Smíte kopírovat a šířit doslovné kopie zdrojového kódu Programu tak, jak jste jej obdržel, a na libovolém médiu, za předpokladu, že na každé kopii viditelně a náležitě zveřejníte zmínku o autorských právech a absenci záruky; ponecháte nedotčené všechny zmínky vztahující se k této licenci a k absenci záruky; a dáte každému příjemci spolu s Programem kopii této licence.

Za fyzický akt přenesení kopie můžete žádat poplatek a podle vlastního uvážení můžete nabídnout za poplatek záruční ochranu.

§2

Můžete modifikovat vaši kopii či kopie Programu anebo kterékoliv jeho části, a tak vytvořit dílo založené na Programu, a kopírovat a rozšiřovat takové modifikace či dílo podle platné podmínky sekce §1 za předpokladu, že splníte všechny tyto podmínky:

a. Modifikované soubory musíte opatřit zřetelnou zmínkou uvádějící, že jste soubory změnil a datum každé změny.

b. Musíte umožnit, aby jakékoliv vámi publikované dílo či rozšiřované dílo, které obsahuje zcela nebo jen zčásti Program nebo jakoukoli jeho část, popřípadě je z Programu nebo jeho části odvozeno, mohlo být jako celek bezplatně poskytnuto každé třetí osobě v souladu s ustanoveními této licence.

c. Pokud modifikovaný program pracuje tak, že čte interaktivně povely, musíte zjistit, že při nejběžnějším způsobu jeho spuštění vytiskne nebo zobrazí hlášení zahrnující příslušnou zmínku o autorském právu a uvede, že neexistuje žádná záruka (nebo popřípadě, že záruku poskytujete vy), a že uživatelé mohou za těchto podmínek Program redistribuovat, a musí uživateli sdělit, jakým způsobem může nahlédnout do kopie této licence. (Výjimka: v případě, že sám program je interaktivní, avšak žádné takové hlášení nevypisuje, nepožaduje se, aby vaše dílo založené na Programu takové hlášení vypisovalo.)

Tyto požadavky se vztahují k modifikovanému dílu jako celku. Pokud lze identifikovat části takového díla, které zřejmě nejsou odvozeny z Programu a mohou být samy o sobě rozumně považovány za nezávislá a samostatná díla,

pak se tato licence a její ustanovení nevztahují na tyto části, jsou-li šířeny jako nezávislá díla. Avšak jakmile tyto části rozšiřujete jako části celku, jímž je dílo založené na Programu, musí být rozšiřování tohoto celku podřízeno ustanovení této licence tak, že povolení poskytnutá dalším uživatelům se rozšíří na celé dílo, tedy na všechny jeho části bez ohledu na to, kdo kterou část napsal.

Smyslem tohoto paragrafu tedy není získání práv na dílo zcela napsané vámi ani popírání vašich práv vůči němu; skutečným smyslem je výkon práva na řízení distribuce odvozených nebo kolektivních děl založených na Programu.

Pouhé spojení jiného díla, jež není na Programu založeno, s Programem (anebo dílem založeným na Programu) na paměťovém nebo distribučním médiu neuvazuje toto jiné dílo do působnosti této licence.

§3

Můžete kopírovat a rozšiřovat Program (nebo dílo na něm založené, viz. sekce §2 v objektové anebo spustitelné podobě podle ustanovení sekcí §1 a §2 výše, pokud splníte některou z následujících náležitostí:

a. Doprovodíte jej zdrojovým kódem ve strojově čitelné formě. Zdrojový kód musí být rozšiřován podle ustanovení sekcí §1 a §2., a to na médiu běžně používaném pro výměnu programového vybavení; nebo

b. Doprovodíte je písemnou nabídkou nejméně na tři roky, podle níž poskytnete jakékoli třetí straně, za poplatek nepřevyšující vaše výdaje vynaložené na fyzickou výrobu zdrojové distribuce, kompletní strojově čitelnou kopii odpovídající zdrojovému kódu, jenž musí být šířen podle ustanovení sekcí §1 a §2 na médiu běžně používaném pro výměnu programového vybavení; nebo

c. Doprovodíte jej informacemi, které jste dostal ohledně nabídky na poskytnutí zdrojového kódu. (Tato alternativa je povolena jen pro nekomerční šíření a jenom tehdy, pokud jste obdržel program v objektovém nebo spustitelném tvaru spolu s takovou nabídkou, v souladu s položkou b výše.)

Zdrojový kód k dílu je nejvhodnější formou díla z hlediska jeho případných modifikací. Pro dílo ve spustitelném tvaru znamená úplný zdrojový kód veškerý zdrojový kód pro všechny moduly, které obsahuje, plus jakékoli další soubory pro definici rozhraní, plus dávkové soubory potřebné pro kompilaci a instalaci spustitelného programu. Zvláštní výjimkou jsou však ty programové komponenty, které jsou normálně šířeny (buď ve zdrojové nebo binární formě) s hlavními součástmi operačního systému, na němž spustitelný program běží (tj. s překladačem, jádrem apod.). Tyto komponenty nemusí být šířeny se zdrojovým kódem, pokud ovšem komponenta sama nedoprovází spustitelnou podobu díla.

Je-li šíření objektového nebo spustitelného kódu činěno nabídkou přístupu ke kopírování z určitého místa, potom se za distribuci zdrojového kódu počítá i nabídnutí ekvivalentního přístupu ke kopírování zdrojového kódu ze stejného místa, byť přitom nejsou třetí strany nuceny ke zkopírování zdrojového kódu spolu s objektovým.

§4

Nesmíte kopírovat, modifikovat, poskytovat sublicence anebo šířit Program jiným způsobem než výslovně uvedeným v této licenci. Jakýkoli jiný pokus o kopírování, modifikování, poskytnutí sublicence anebo šíření Programu je neplatný a automaticky ukončí vaše práva daná touto licencí. Strany, které od vás obdržely kopie anebo práva v souladu s touto licencí, však nemají své licence ukončeny, dokud se jim plně podřizují.

§5

Není vaší povinností tuto licenci přijmout, protože jste ji nepodepsal. Nic jiného vám však nedává možnost kopírovat nebo šířit Program nebo odvozená díla. V případě, že tuto licenci nepřijmete, jsou tyto činnosti zákonem zakázány. Tím pádem modifikací anebo šířením Programu (anebo každého díla založeného na Programu) vyjadřujete své podřízení se licenci a všem jejím ustanovením a podmínkám pro kopírování, modifikování a šíření Programu a děl na něm založených.

§6

Pokaždé, když redistribuujete Program (nebo dílo založené na Programu), získává příjemce od původního držitele licence právo kopírovat, modifikovat a šířit Program v souladu s těmito ustanoveními a podmínkami. Nesmíte klást žádné překážky výkonu zde zaručených příjemcových práv. Nejste odpovědný za vymáhání dodržování této licence třetími stranami.

§7

Jsou-li vám z rozhodnutí soudu, obviněním z porušení patentu nebo z jakéhokoli jiného důvodu (nejen v souvislosti s patenty) uloženy takové podmínky (ať již příkazem soudu, smlouvou nebo jinak), které se vylučují s podmínkami této licence, nejste tím osvobozen od podmínek této licence. Pokud nemůžete šířit Program tak, abyste vyhověl zároveň svým závazkům vyplývajícím z této licence a jiným platným zůvazkům, nesmíte jej v důsledku toho šířit vůbec. Pokud by například patentové osvědčení nepovolovalo bezplatnou redistribuci Programu všemi, kdo vaším přičiněním získají přímo nebo nepřímo jeho kopie, pak by jediný možný způsob jak vyhovět zároveň patentovému osvědčení i této licenci spočíval v ukončení distribuce Programu.

Pokud by se za nějakých specifických okolností jevila některá část tohoto paragrafu jako neplatná nebo nevynutitelná, považuje se za směrodatnou rovnováha vyjádřená tímto paragrafem a paragraf jako celek se považuje za směrodatný za jiných okolností.

Smyslem tohoto paragrafu není navádět vás k porušování patentů či jiných ustanovení autorského práva, anebo tato ustanovení zpochybňovat; jediným jeho smyslem je ochrana integrity systému šíření volného programového vybavení, které je podloženo veřejnými licenčními předpisy. Mnozí lidé poskytli své příspěvky do širokého okruhu programového vybavení šířeného tímto systémem, spolehnuvše na jeho důsledné uplatňování; záleží na autorovi/dárci, aby rozhodl, zda si přeje šířit programové vybavení pomocí nějakého jiného systému a žádný uživatel licence nemůže takové rozhodnutí zpochybňovat.

Smyslem tohoto paragrafu je zevrubně osvětlit to, co je považováno za důsledek plynoucí ze zbytku této licence.

§8

Pokud je šíření či použití Programu v některých zemích omezeno buď patenty anebo autorsky chráněnými rozhraními, může držitel původních autorských práv, který svěřuje Program do působnosti této licence, přidat výslovně omezení pro geografické šíření, vylučující takové země, takže šíření je povoleno jen v těch zemích nebo mezi těmi zeměmi, které nejsou tímto způsobem vyloučeny. Tato licence zahrnuje v tomto případě takové omezení přesně tak, jako bylo zapsáno v textu této licence.

§9

Free Software Foundation může čas od času vydávat upravené nebo nové verze Obecné veřejné licence. Takové nové verze se budou svým duchem podobat současné verzi, v konkrétních věcech se však mohou lišit s ohledem na nové problémy či zájmy.

Každé nové verzi je přiděleno rozlišující číslo verze. Pokud Program specifikuje číslo verze, která se na něj vztahuje, a „všechny následující verze“ , můžete se podle uvážení řídit ustanoveními a podmínkami buďto oné konkrétní verze anebo kterékoliv následující verze, kterou vydala Free Software Foundation. Jestliže Program nespecifikuje číslo verze této licence, můžete si vybrat libovolnou verzi, kterou kdy Free Software Foundation vydala.

§10

Pokud si přejete zahrnout části Programu do jiných volných programů, jejichž distribuční podmínky jsou odlišné, zašlete autorovi žádost o povolení. V případě programového vybavení, k němuž vlastní autorská práva Free Software Foundation, napište Free Software Foundation; někdy činíme výjimky ze zde uvedených ustanovení. Naše rozhodnutí bude vedeno dvěma cíli; zachováním volné povahy všech odvozenin našeho volného programového vybavení a podporou sdílení a opětovného využití programového vybavení obecně.

§11

ZÁRUKA SE NEPOSKYTUJE

VZHLEDEM K BEZPLATNÉMU POSKYTNUTÍ LICENCE K PROGRAMU SE NA PROGRAM NEVZTAHUJE ŽÁDNÁ ZÁRUKA, A TO V MÍŘE POVOLENÉ ZÁKONEM. POKUD NENÍ PÍSEMNĚ STANOVENO JINAK, POSKYTUJÍ DRŽITELÉ AUTORSKÝCH PRÁV POPŘÍPADĚ JINÉ STRANY PROGRAM „TAK, JAK JE“, BEZ ZÁRUKY JAKÉHOKOLIV DRUHU, AŤ VÝSLOVNÉ NEBO VYPLÝVAJÍCÍ, VČETNĚ, ALE NIKOLI JEN, ZÁRUK PRODEJNOSTI A VHODNOSTI PRO URČITÝ ÚČEL. POKUD JDE O KVALITU A VÝKONNOST PROGRAMU, LEŽÍ VEŠKERÉ RIZIKO NA VÁS. POKUD BY SE U PROGRAMU PROJEVILY ZÁVADY, PADAJÍ NÁKLADY ZA VŠECHNU POTŘEBNOU ÚDRŽBU, OPRAVY ČI NÁPRAVU NA VÁŠ VRUB.

§12

V ŽÁDNÉM PŘÍPADĚ, S VÝJIMKOU TOHO, KDYŽ TO VYŽADUJE PLATNÝ ZÁKON, ANEBO KDYŽ TO BYLO PÍSEMNĚ ODSOUHLASENO, VÁM NEBUDE ŽÁDNÝ Z DRŽITELŮ AUTORSKÝCH PRÁV ANI ŽÁDNÁ JINÁ STRANA, KTERÁ SMÍ MODIFIKOVAT ČI ŠÍŘIT PROGRAM V SOULADU S PŘEDCHOZÍMI USTANOVENÍMI, ODPOVĚDNÝ ZA ŠKODY, VČETNĚ VŠECH OBECNÝCH, SPECIÁLNÍCH, NAHODILÝCH NEBO NÁSLEDNÝCH ŠKOD VYPLÝVAJÍCÍCH Z UŽÍVÁNÍ ANEBO NESCHOPNOSTI UŽÍVAT PROGRAMU (VČETNĚ ALE NIKOLI JEN, ZTRÁTY NEBO ZKRESLENÍ DAT, NEBO TRVALÝCH ŠKOD ZPŮSOBENÝCH VÁM NEBO TŘETÍM STRANÁM, NEBO SELHÁNÍ FUNKCE PROGRAMU V SOUČINNOSTI S JINÝMI PROGRAMY), A TO I V PŘÍPADĚ, ŽE TAKOVÝ DRŽITEL AUTORSKÝCH PRÁV NEBO JINÁ STRANA BYLI UPOZORNĚNI NA MOŽNOST TAKOVÝCH ŠKOD.

KONEC UNSTANOVENÍ A PODMÍNEK

A.3.3 Jak uplatnit tato ustanovení na vaše nové programy

Pokud vyvinete nový program a chcete, aby byl veřejnosti co nejvíce k užitku, můžete toho nejlépe dosáhnout tím, že jej prohlásíte za volné programové vybavení, které může kdokoliv redistribuovat a měnit za zde uvedených podmínek.

K tomu stačí připojit k programu následující údaje. Nejbezpečnější cestou je jejich připojení na začátek každého zdrojového souboru, čímž se nejúčinněji sdělí vyloučení záruky; a v každém souboru by pak měla být přinejmenším řádka s „copyrightem“ a odkaz na místo, kde lze nalézt úplné údaje.

> <řádka se jménem programu a nástinem toho, co dělá>
> Copyright ©19xx jméno autora
>
> Tento program je volné programové vybavení; můžete jej šířit a modifikovat podle ustanovení Obecné veřejné licence GNU, vydávané Free Software Foundation; a to buď verze 2 této licence anebo (podle vašeho uvážení) kterékoli pozdější verze.
>
> Tento program je rozšiřován v naději, že bude užitečný, avšak BEZ JAKÉKOLI ZÁRUKY; neposkytují se ani odvozené záruky PRODEJNOSTI anebo VHODNOSTI PRO URČITÝ ÚČEL. Další podrobnosti hledejte v Obecné veřejné licenci GNU.
>
> Kopii Obecné veřejné licence GNU jste měl obdržet spolu s tímto programem; pokud se tak nestalo, napište o ni Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Připojte rovněž informaci o tom, jak je možné se s vámi spojit elektronickou a papírovou poštou.

Pokud je program interaktivní, zařiďte, aby se při startu v interaktivním módu vypsalo hlášení podobné tomutu:

```
Gnomovision verze 69, Copyright (C)19xx jméno autora.
Gnomovision je ABSOLUTNĚ BEZ ZÁRUKY; podrobnosti
se dozvíte zadáním show w. Jde o volné programové
vybavení a jeho šíření za jistých podmínek je vítáno;
podrobnosti získáte zadáním show c.
```

Hypotetické povely show w a show c by měly zobrazit příslušné pasáže Obecné veřejné licence. Odpovídající povely ovšem nemusí být právě show w a show c; mohou to být třeba stisky tlačítka na myši nebo položky v menu – cokoliv, co se do vašeho programu hodí.

Pokud je to nutné, měl byste také přimět svého zaměstnavatele (jestliže pracujete jako programátor) nebo představitele vaší školy, je-li někdo takový, k tomu, aby podepsal „zřeknutí se autorských práv“. Zde je vzor; jména pozměňte:

> Yoyodyne, a.s., se tímto zříká veškerého zájmu o autorská práva k programu ‘Gnomovision’ (překladač s nakladačem) napsaného Jamesem Hackerem.
>
> <Tomáš Složitý — podpis>, 1. dubna 1989
> Tomáš Složitý, více než prezident

Tato Obecná veřejná licence neumožňuje zahrnutí vašeho programu do jiných než volných programů. Je-li váš program knihovnou podprogramů, můžete zvážit, zda je užitečné umožnit sestavování i vázaných aplikačních programů s vaší knihovnou. V takovém případě použijte Obecnou knihovní licenci GNU namísto této licence.

Literatura

[Bach86] M. J. Bach: **Principy operačního systému Unix**.
Nakladatelství Prentice Hall, 1986. Český překlad vydavatelství Softwarové Aplikace a Systémy, Praha, 1993.
ISBN 80-901507-0-5

[Brod89] J. Brodský, L. Skočovský: **Operační systém Unix a jazyk C**.
Nakladatelství SNTL, Praha, 1989.
ISBN 80–03–00049–1

[Burk97] R. Burk, D. B. Horvath: **UNIX System Administrator's Edition Unleashed**.
Vydavatelství SAMS, Indianapolis, 1997.
ISBN 0–672–30952–1

[Corp99] Materiály firmy **Corpus Solutions, a.s.**.
(`http://www.corpus.cz`)

[DoD83] Department of Defense: **Kritéria hodnocení zabezpečených počítačových systémů**.
Český překlad Orange Book – Trusted Computer System Evaluation Criteria, CSC–STD–001–83. Nakladatelství BEN – technická literatura, Praha, 1994

[Dost98] L. Dostálek, A. Kabelová: **Bezpečnost na internetu**.
(`http://info.pvtnet.cz/bezpec.htm`)

[Hunt92] C. Hunt: **Konfigurace a správa sítí TCP/IP**.
Nakladatelství O'Reilly & Associates, Inc., 1992. Český překlad vydavatelství Computer Press, Praha, 1997.
ISBN 80–7226–024-3

[Chap95] D. B. Chapman, E. D. Zwicky: **Firewally – principy budování a udržování**.
Nakladatelství O'Reilly & Associates, Inc., 1995. Český překlad vydavatelství Computer Press, Praha, 1998.
ISBN 80–7226–051–0

[Ches95] W. R. Cheswick, S. M. Bellovin: **Firewally a bezpečnost Internetu, aneb jak zahnat lstivého hackera**.
Nakladatelství Addison–Wesley Publishing Company, 1995. Český překlad vydavatelství Science, Veletiny, 1998.
ISBN 80–86083–01–2

[Klan97] L. Klander: **Hacker proof**.
Nakladatelství Jamsa Press, 1997. Český překlad vydavatelství UNIS Publishing, Praha, 1998.
ISBN 80–86097–15–3

[Kern78] B. W. Kernighan, D. M. Ritchie: **Programovací jazyk C**.
Nakladatelství Prentice Hall, 1978. Slovenský překlad vydavatelství ALFA, Bratislava, 1989.
ISBN 80–05–00154–1

[Kirc98] J. Kirch: **Microsoft Windows NT versus UNIX**.
Článek věnovaný porovnání MS Windows NT a Unixu. Do češtiny přeložil Hanuš Adler.
(`http://www.penguin.cz/~had/unix-nt/`)

[Lhot96] L. Lhotka: **Server v Internetu**.
Nakladatelství KOPP, České Budějovice, 1996.
ISBN 80–85828–65–0

[Peek97] J. Peek, T. O'Reilly, M. Loukides: **UNIX power tools**.
Nakladatelství O'Reilly & Associates, Inc., Sebastopol, 1997.
ISBN 1–56592–260–3

[Perf97] SysAdmin editors: **UNIX Performance Tuning**.
Nakladatelství R&D Books, Lawrence, 1997.
ISBN 0–87930–470–7

[Perf99] Časopis **Performance Computing**.
Vydavatelství Miller Freeman, Inc., San Francisco, 1999.
Rubrika Daemons & dragons.

[Pet98] J. Peterka: **Archiv článků o internetu**.
(`http://ksi.ms.mff.cuni.cz/~peterka/archiv/`)

[Prib96] J. Přibyl, J. Kodl: **Ochrana dat v informatice**.
Vydavatelství ČVUT, Praha, 1996.
ISBN 80–01–01664–1

[Root99] Server **Root Shell**.
Články věnované slabým místům a průnikům do systému Unixu.
(`http://www.rootshell.com`)

[Ryb90] J. Rybička: **LaTeX pro začátečníky**.
Druhé brožované vydání. Nakladatelství KONVOJ, Brno, 1999.
ISBN 80–85615–77–0

[Salu94] P. H. Salus: **A Quarter Century of UNIX**.
Nakladatelství Addison–Wesley Publishing Company, 1994.
Český překlad části textu časopis BYTE, nakladatelství McGraw–Hill, české vydání, Praha, 1994, číslo 10 (říjen).

[Sam99] Server **SysAdmin**.
Články věnované správě systému Unix.
(`http://www.samag.com/`)

[Sans99] Materiály organizace **SANS**.
(`http://www.sans.org`)

[Satr98] P. Satrapa, J. A. Randus: **LINUX – Internet server**.
Druhé upravené vydání, vydavatelství Neokortex, Praha, 1998.
ISBN 80–902230–3–6

[Sec97] SysAdmin editors: **UNIX Security**.
Nakladatelství R&D Books, Lawrence, 1997.
ISBN 0–87930–471–5

[Sec98] Časopis **Data security management**.
Vydavatelství Tate International, s.r.o., Praha, 1998.
Jozef Vyskoč: Základní bezpečnostní principy.

[Ser99] Časopis **Server/Workstation Expert**.
Vydavatelství Computer Publishing Group, Newton Cenetr, 1998, 1999.
Rubrika System administration.

[Skoc93] L. Skočovský: **Principy a problémy operačního systému UNIX**.
Vydavatelství Science, Veletiny, 1993.
ISBN 80–901475–0–x

[Sol97] Sun Education Services: **Solaris 2.x System Administration, Student Guide**.
Sun Microsystems, Inc., Mountain View, 1997.

[Stev90] W. R. Stevens: **Programování sítí operačního systému Unix**.
Nakladatelství Prentice Hall, 1990. Český překlad vydavatelství Science, Veletiny, 1994.
ISBN 80–901475–3–4

[Stok99] Server **Stokely Consulting**.
Články věnované zabezpečení systému Unix.
(`http://www.stokely.com/`)

[Stol90] C. Stoll: **Kukaččí vejce**.
Nakladatelství Pocket Books, New York, 1990. Český překlad vydavatelství Mladá fronta, Praha, 1997.
ISBN 80–204–0594–1

[Sun97] Časopis **SUNEXPERT**.
Vydavatelství Computer Publishing Group, Newton Cenetr, 1997, 1998.
Rubrika System administration.

[Sun99] Materiály firmy **Sun Microsystems, s.r.o.**.
(`http://www.sun.cz`)

[Sys98] Časopis **SysAdmin**.
Vydavatelství Miller Freeman, Inc., San Francisco, 1998, 1999.
Články věnované počítačové a síťové bezpečnosti.

[Sysw99] Server **Sysadmin**.
Články věnované administrátům systému Unix.
(`http://www.sysadmin.com/`)

[Uher98] M. Uher: **Zabezpečení podnikové sítě (Intranetu)**.
Bakalářská práce, ČVUT FEL, Praha, 1998.

[Unix97] Časopis **Unix review's**.
Vydavatelství Miller Freeman, Inc., San Francisco, 1997.
Rubrika Daemons & dragons.

[Unix99] Server **Unix Guru Universe**.
Články věnované zabezpečení systému Unix.
(`http://www.ugu.com/ugu/`)

[Wets97] J. R. Wetsch: **Distributed UNIX System Administration**.
Nakladatelství R&D Books, Lawrence, 1999.
ISBN 0–87930–540–1

[Wels96] M. Welsh, L. Kaufman: **Používáme Linux**.
Nakladatelství O'Reilly & Associates, Inc., 1996. Český překlad vydavatelství Computer Press, Praha, 1997.
ISBN 80–7226–001–4

[Wood74] C. Woodward, R. Bernstein: **All the President's Men**.
Nakladatelství Simon & Schuster, New York, 1974.

[Zak148] **Zákon č.148/1998 Sb.**
Zákon č.148/98 - *Zákon o ochraně utajovaných skutečností* v platném znění, Sbírka zákonů České republiky.

[Zem93] P. Zemánek: **Základy operačního systému UNIX**.
Česká informatická společnost, Praha, 1993.