Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství

Bakalářská práce

Linuxový proxy server v sítích Active Directory

Autor: **Michal Šťovíček**

Vedoucí práce: Ing. Marek Pícka, Ph.D.

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Linuxový proxy server v sítích Active Directory" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 10.3.2013 _____

Poděkování

Rád bych touto cestou poděkoval vedoucímu mé bakalářské práce panu Ing. Markovi Píckovi, Ph.D. za doporučení a cenné rady, které mi pomohly k sepsání této bakalářské práce.

Linuxový proxy server v sítích Active Directory

Souhrn

Tato bakalářská práce se zabývá instalací operačního systému Debian GNU/Linux a konfigurací potřebných aplikací pro běh tohoto systému v roli proxy serveru. Nejprve popisuje teoretické znalosti, nutné k pochopení praktické části a dále již konkrétní kroky instalace konfigurace jednotlivých aplikací. V závěru autor popisuje, zda se povedlo splnit všechny zamýšlené cíle.

Klíčová slova

Proxy, Samba, Linux, Windows, Active Directory, Server

Linux Proxy Server in an Active Directory domain

Summary

This bachelor thesis deals with installation of Debian GNU/Linux Operating System and configuration of applications which are necessary for running this Operating System as Proxy Server. At first, theoretical knowledge is described, which is necessary to understand the practical part of this bachelor thesis. Then there is the practical part, in which concrete steps are described, how to install Debian with applications and how to configure these applications. At the end of this thesis the author has written whether or not the goals of the bachelor thesis were accomplished.

Keywords

Proxy, Samba, Linux, Windows, Active Directory, Server

Obsah

Úvod

Tato bakalářská práce nese název Linuxový proxy server v doméně Active Directory. Téma bakalářské práce si autor zvolil kvůli zájmu o operační systémy GNU/Linux a také možnosti využít tohoto projektu v reálném prostředí základní školy, kde je zaměstnancem.

V této době, kdy je internet plný nevhodného obsahu, je potřeba zajistit řešení, které by takovýto obsah dokázalo spolehlivě filtrovat. Poměrně zajímavou možnost filtrování spatřuje autor práce v nasazení proxy serveru. Proxy server byl dříve využíván hlavně z důvodu šetření přenesených dat tím, že pokud si více klientů vyžádalo přístup na jednu internetovou stránku, uložil si ji do své paměti a klientům ji poskytoval po lokální síti. V dnešní době, kdy je rychlost internetových připojení v řádech Megabitů za sekundu, již toto nemá velký význam. Proto se používá proxy jako firewall, který umí blokovat přístup k různým webům, nebo spouštění souborů.

Bakalářská práce je rozdělena do dvou základních částí. První část je teoretická, která napomáhá čtenáři k pochopení daného tématu a lepší orientaci v druhé, praktické části bakalářské práce.

V první kapitole bakalářské práce jsou vypsány a vysvětleny pojmy vyskytující se v praktické části. Kapitola se věnuje částečně i operačním systémům Windows od společnosti Microsoft, neboť tento systém bude pracovat jako hlavní server a bude obsahovat databázi skupin a uživatelů. V následující kapitole bude popsáno praktické řešení včetně popsaných ukázek konfiguračních souborů jednotlivých aplikací.

Cíl práce

Cílem této bakalářské práce je nakonfigurování počítače s operačním systémem GNU/Linux v roli proxy serveru a zajištění spolupráce tohoto serveru s Windows serverem 2008r2 v roli doménového řadiče se službou Active Directory. Proxy server by měl být následně nasazen v prostředí základní školy o celkovém počtu 120 PC.

Metodika

Bakalářská práce je založena na analýze požadavků školy, ve které by mělo být řešení zavedené a studiu odborné literatury, zabývající se operačními systémy GNU/Linux a Microsoft Windows server.

V první kapitole této bakalářské práce budou popsány pojmy, které je nutno znát k pochopení praktické části.

Druhá kapitola bude praktická, bude v ní popsáno stávající řešení a následně navrženo nové řešení podle požadavků školy. Nakonec dojde ke konfiguraci a nasazení nového řešení a jeho následného vyhodnocení.

1. Literární rešerše

První kapitola se věnuje několika nezbytným pojmům, které je třeba vymezit pro správné pochopení praktické části této bakalářské práce. V praktické části bude použito mnoho pojmů, ovšem některé jen okrajově.

1.1. Operační systém Linux

Linux byl původně pouze jádrem operačního systému, které bylo vytvořeno Linusem Torvaldsem při jeho studiu na vysoké škole. Vzorem mu byl operační systém MINIX. Původně byl Linux vyvíjen jen jako koníček, ale po zveřejnění první verze jádra na internetu začal být o Linux velký zájem. Ze začátku byl využíván projekt GNU, zabývající se vývojem volně šiřitelného software a Linux využíval (a nadále využívá) i licenci tohoto projektu GNU/GPLv2.

V současné době pojem Linux označuje celé distribuce operačního systému. Jen několik z nich je stále označováno GNU/Linux kvůli používání některých aplikací z projektu GNU. Tyto distribuce mohou být buďto volně šiřitelné, nebo placené. U placených distribucí se ovšem platí pouze za podporu, neboť jádro operačního systému i ostatní aplikace, kromě těch, které si společnosti, prodávající distribuce, samy napíší, musí být šířeny pod licencí GPLv2, která říká, že kód musí být dostupný zdarma a každý si ho může upravovat podle svého uvážení.

Linux používá monolitické jádro, což znamená, že jádro řídí procesy, vstupy, výstupy a tak dále. Jinými slovy obsluhuje veškerá systémová a hardwarová volání.

V Linuxu může být použito několik různých souborových systémů. Nejznámější je Ext3, což je nástupce souborového systému Ext2, obohacený o žurnál. Souborový systém Ext3 je obousměrně kompatibilní se svým předchůdcem, což umožňuje přechod mezi těmito souborovými systém bez nutnosti zálohy dat. Ext3 dále nabízí uživateli zvolit, zda chce vysoký výkon, nebo co nejvyšší odolnost proti nekonzistenci dat. Dále je tu střední možnost, která je také výchozí.

1.1.1. Grafické prostředí Linuxu

Grafické prostředí v Unixových systémech není, na rozdíl od Windows, integrováno do jádra. Toto prostředí běží jako samostatná aplikace a díky tomu stoupá i odolnost systému proti haváriím. Pokud v takovém systému selže grafické prostředí, jádro systému běží dále nezávisle na tomto selhání. V Linuxu se používá uživatelské rozhraní X-windows, které definuje způsob, jakým se budou chovat a vykreslovat objekty. Pro vytvoření uživatelsky přívětivého prostředí se k rozhraní X-windows používají správci oken. Nejznámější a též nejpoužívanější správci jsou KDE a GNOME.

1.2. Operační systém Windows

Windows je operačním systémem společnosti Microsoft. Tento systém je placený a vychází ve verzích pro desktopy a pro servery. Verze pro osobní počítače je ve světě nejrozšířenějším operačním systémem. Momentálně je k dispozici verze s označením Windows 8, která je zaměřena na zařízení s dotykovým displayem. Serverový typ Windows má nyní označení Windows server 2013, ovšem v našem případě bude použita starší verze Windows server 2008 r2.

Tento operační systém používá, na rozdíl od Unixových systémů, takzvané mikrojádro. Toto jádro operačního systému poskytuje jen minimum služeb a komunikuje s dalšími službami. Služby pak zajišťují řízení procesů a podobně.

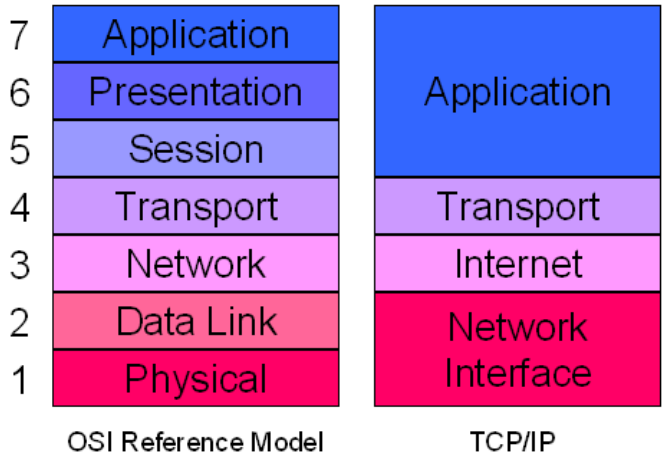
Windows používá systém souborů NTFS (New Technology File Systém), který umožňuje žurnálování, přidělování práv souborům, vytváření diskových kvót a v neposlední řadě kompresi na úrovni souborového systému.

1.3. Protokoly

Následující podkapitola se bude zabývat několika vybranými druhy protokolů, které budou použity v praktické části bakalářské práce.

1.3.1. TCP/IP

TCP/IP je skupina protokolů, z nichž nejznámější jsou právě TCP (Transmission Control Protocol) a IP (Internet Protocol). Tato sada také udává, jak by se měly stavět a jak by měly fungovat počítačové sítě. Obsahem TCP/IP je i členění softwaru na jednotlivé vrstvy, úlohy těchto vrstev a také jaké by měly tyto vrstvy plnit úkoly. Tím se má na mysli použití protokolů na jednotlivých vrstvách.



Obrázek 1: Srovnání referenčního modelu ISO/OSI a modelu TCP/IP

V práci budou zmíněny protokoly http, ftp a jejich zabezpečené verze https a ftps, které si nyní představíme trochu podrobněji.

1.3.2. Protokoly http a https

Zkratka http značí hyper text transfer protocol. Tento protokol je bezstavový. To znamená, že probíhá komunikace klient-server tak, že klient odešle dotaz na server, a ten následně odešle odpověď klientovi. O jakékoli zachovávání informací o stavu se stará sám klient. Protokol http se řadí do vrstvy aplikační, takže řeší pouze věci spojené s www a neřeší transport dat. Předpokladem pro funkci transportu je existence transportního protokolu, kterým může být například TCP.

Protokol http má ještě zabezpečenou verzi, označovanou https. Tento protokol zasílá zprávy pomocí SSL (Secure Socket Layer), nebo přes TLS (Transport Layer

Security) spojení. Toto zabezpečení výrazně ztěžuje odposlech posílaných zpráv útočníkem. Standardně pracují protokoly http na portu 80 a https na portu 443.

1.3.3. Protokoly ftp a ftps

Třípísmenná zkratka ftp je synonymem pro File Transfer Protocol. Tento protokol se v dnešní době začíná nahrazovat kvůli bezpečnosti. Slouží, jak už název napovídá, k přenosu souborů. Na serveru, kde běží server ftp jsou nasdílena data a klient se může pomocí ftp klienta připojit k tomuto serveru. Dále už může pomocí příkazů kopírovat soubory z ftp serveru, nebo naopak na server. Zabezpečená verze protokolu ftp se liší pouze tím, že pro přístup a přenos dat používá SSH (Secure Shell).

SSH se také používá pro přístup a správu vzdáleného počítače. Je možné přes něj spouštět různé příkazy, programy a tak podobně. SSH pracuje s šifrováním veřejného klíče: *„ Secure Shell je založen na technologii šifrování veřejným klíčem. Pracuje podobně jako bezpečnostní schránky v bance. Abyste schránku otevřeli, potřebujete dva klíče. V případě šifrování potřebujete dva matematické klíče, veřejný a soukromý. Váš veřejný klíč můžete uveřejnit např. na internetových stránkách, nechat si jej natisknout na tričko nebo umístit na billboard v rušné části města. Každý, kdo o to požádá, může dostat kopii veřejného klíče. Váš soukromý klíč však musíte maximálně zabezpečit. Jde o skutečné zabezpečení dat, která chcete zašifrovat. Kombinace veřejný klíč + soukromý klíč by měly být naprosto ojedinělé a neměly by se opakovat.“*[1]

1.3.4. Kerberos

Protokol **Kerberos** slouží k ověřování identity v nezabezpečené síti. K autentizaci uživatelů je potřeba důvěryhodná třetí strana, jejíž roli představuje centrální autentizační server. Tento server, nazývaný též KDC (Key Distribution Center), má na starost databázi uživatelů a přiděluje jim takzvané Tickety (lístky). Tento lístek je přenášen po síti místo

[1] SHAH, Steve. *Administrace systému Linux: překlad čtvrtého vydání*. 1. vyd. Praha: Grada, 2007, s. 296.

hesla, neobsahuje tajné informace a má omezenou životnost, po jejímž skončení si uživatel může zažádat o nový.

Ověřování pomocí protokolu Kerberos probíhá takto:

1) Klient pomocí hesla prokáže svou identitu serveru KDC
2) Server KDC vydá klientovi lístek TGT (Ticket-Granting Ticket) a klient na základě tohoto lístku získá přístup ke službě TGS (Ticket-Granting Service). Služba TGS je součástí ověřování v řadiči domény.
3) Služba TGS vydá klientovi lístek služby, který slouží k prokázání identity uživatele vůči službě v síti a zároveň k prokázání této služby uživateli.

Více o protokolu Kerberos se můžete dočíst v knize *Kerberos: definitive guide*[2]

1.4. Proxy server

Proxy server je software, který byl dříve používán kvůli pomalému připojení k síti internet. Klientské stanice přistupují k proxy serveru a ten si podle požadavku uloží data do své paměti. Pokud si klient vyžádá informaci, která je uložena v paměti proxy serveru, je mu poskytnuta tato kopie bez nutnosti přistupovat na internet. Tato funkce je velmi výhodná v případě pomalého připojení k internetu a více klientských stanic, které tuto jedinou linku sdílejí. Stručný popis proxy serveru by mohl vypadat takto:

„Proxy server je server (počítačový systém, nebo aplikace), který se chová jako prostředník pro požadavky klienta, který hledá věci na jiném serveru. Klient se připojí k proxy serveru, vyžádá si určitou službu (například připojení, soubor, webovou stránku…) a proxy server tento požadavek zhodnotí dle svých pravidel.

Koncept proxy serveru byl vynalezen v raných dobách distribučních systémů, primárně za účelem zjednodušení a kontroly jejich komplexnosti. V dnešní době slouží většina proxy serverů jako tzv. webové proxy, které dovolují připojení do WWW (World Wide Web).

Proxy servery mají několik různých využití, mezi něž patří například:

- *mohou z bezpečnostních důvodů poskytovat serverům za nimi anonymitu*
- *mohou urychlovat přístup ke zdrojům (díky cachingu)*

[2] GARMAN, Jason a Charles PERKINS. *Kerberos: definitive guide*. Vyd. 1. New York: O´Reilly, 2003, 253 s. ISBN 05-960-0403-6.

- *mohou aplikovat přístupovou politiku ke službám a obsahu (například pro blokování nežádoucích stránek)*
- *mohou přistupovat ke stránkám, které jsou zakázané nebo filtrované určitým poskytovatelem připojení či institucí*
- *mohou obejít bezpečností či rodičovské filtry*
- *mohou obejít internetové filtry a dostat se tak ke stránkám, které blokuje např. vláda*
- *mohou sledovat posílaný obsah a detekovat tak včas malware*
- *mohou sledovat odchozí obsah a zabránit tak například ztrátě dat"* [3]

1.5. DNS

Zkratka **DNS** značí Domain Name System. Internetové protokoly rozpoznávají jednotlivá zařízení v síti pomocí adres IP, které jsou 32-bitové, rozdělené na 4 části po osmi bitech a zapsaných dekadicky. Tato čísla ovšem nejsou zapamatovatelná. Z tohoto důvodu bylo vytvořeno právě DNS. Je to velmi objemná distribuovaná databáze se souborem opatření, zajišťující jedinečnost symbolických názvů domén. Díky této databázi je možné překládat symbolické názvy na adresy IP a zpět.

1.6. DHCP

Pod pojmem **DHCP** rozumíme Dynamic Host Control Protocol. Ten zajišťuje přidělení IP adresy, masky podsítě, výchozí brány (tzv. gateway) a DNS serverů.

„Počítač při startu kontaktuje DHCP-server a vyžádá si od něj potřebné informace. DHCP-server přiděluje IP-adresy dynamicky. Má vymezený rozsah adres a z něho rozdává a bere zpět. Klientovi server adresu propůjčuje jen na určitou dobu. Klient musí před uplynutím doby zažádat o obnovení IP-adresy. To brání tomu, aby vypnuté respektive havarované počítače blokovaly adresy. Protokol klientovi umožňuje adresy se vzdát. Klient by tak měl učinit při vypínání.

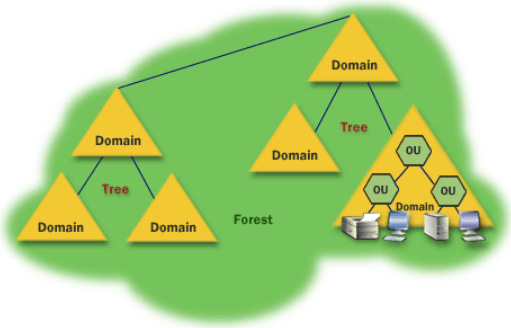
[3] Slovníček pojmů: Proxy server. [online]. [cit. 2012-12-29]. Dostupné z: http://hosting.blueboard.cz/slovnicek-pojmu/proxy-server

Protokol DHCP umožňuje též zavádění operačního systému ze sítě. Je rozšířením staršího protokolu BOOTP a je s ním zpětně kompatibilní. Rozšíření je například právě ve zmiňovaném přidělování IP-adres jen na určitou dobu.“[4]

1.7. Active Directory

Active Directory je adresářová služba protokolu LDAP (Lightweight Directory Access Protocol), která je implementovaná společností Microsoft pro použití v prostředí operačního systému Windows. Tato služba umožňuje nastavovat politiku, instalovat software, nebo aktualizace najednou klientským stanicím v organizační struktuře. V nabídce je možnost vytvářet takzvané organizační jednotky, kterými mohou být například doménoví uživatelé, nebo takzvané kontejnery, které mohou obsahovat jiné objekty. Všechny informace jsou uchovávány v databázovém souboru. Struktura Active Directory je rozdělena na fyzickou a logickou.

1) Logická struktura: tvoří ji Forest (les), Trees(stromy), Domény a organizační jednotky (OU-Organization Units). Hlavní je Les, který může obsahovat více stromů. Strom potom tvoří jedna, nebo více domén. Doména se dále skládá z organizačních jednotek a uvnitř těchto jednotek se již nacházejí objekty. Objektem může být například počítač, tiskárna, uživatel.



Obrázek 2: Logická struktura Active Directory

[4] SCHNIKOW, K. T. DHCP: Dynamic Host Control Protocol. *DHCP* [online]. 2006, s. 1, 15-2-2006 [cit. 2013-03-04]. Dostupné z: http://www.abclinuxu.cz/slovnik/dhcp

2) Fyzická struktura: je tvořena z doménových řadičů a sítí/podsítí. Doménový řadič je server, na kterém je Active Directory, nebo je jeho část. Sítě a podsítě jsou tvořeny nějakým rozsahem adres a ve většině případů se toto rovná síti LAN.

Doména je stavební kámen logické struktury Active Directory. Doména není omezena místem, můžou do ní být připojeny zařízení ze všech poboček společnosti. Přístup k objektům v doméně je řízen pomocí Access Control Listu, což je seznam oprávnění, který určuje kdo má právo přistupovat k objektu a jaké s ním může provádět operace. Tato oprávnění nemohou přejít na jiné domény.

Pojem Organizační jednotka označuje nejmenší jednotku v logické struktuře Active Directory. Organizační jednotky se do sebe dají vnořovat a tak se díky nim dá vytvářet požadovaná hierarchická struktura. Tato struktura se obvykle vytváří tak, aby co nejvíce usnadnila orientaci v dané organizaci.

Strom je seskupení jedné, či více domén. Vzniká připojením podřízené domény (child domain) k rodičovské doméně (root domain).

Doménový řadič je v našem případě počítač se systémem Windows server 2008 r2, který obsahuje lokální doménovou databázi. Jedna doména může mít i více doménových řadičů, z nichž každý obsahuje úplnou repliku celé databáze. Pokud provedeme změnu v databázi jednoho řadiče, spustí se automatická replikace na všech ostatních řadičích v doméně, aby každý obsahoval stejné informace. Jeden řadič může být zároveň součástí jen jedné domény. Doménový řadič také zprostředkovává autentifikaci uživatelů.

Globální katalog je služba, která umožňuje vyhledávat objekty uvnitř jiné domény v rámci jednoho aresáře. Jedná se o centrální repositář, který obsahuje informace o objektech z celého stromu, nebo lesa. Dále podává informace o členství v univerzálních skupinách, nutných pro přihlášení.

1.8. Samba

Software **Samba** představuje balík několika aplikací, které implementují protokol SMB (Server Message Block) do systémů typu UNIX. Tento balík aplikací je obsažen ve většině distribucí Linuxu, nebo je možné ho stáhnout z internetových stránek samby.

Samba se používá pro sdílení souborů a tiskáren mezi Unixovými operačními systémy a OS Windows. V nejnovější verzi již Samba dokáže plnit roli primárního řadiče domény Active Directory, nebo se stát plnohodnotným členem této domény. Samba realizuje své služby dvěma démony:

„*Kód serveru se systémem Samba se v podstatě skládá ze dvou démonů: smbd a nmbd. Démon smbd obsluhuje vlastní sdílení souborových systémů a tiskových služeb pro klienty. Nejprve se spojí s portem 139 a naslouchá požadavkům. Při každé autentizaci klienta se smbd zkopíruje; originál se vrátí na port 139 pro nové požadavky a kopie obsluhuje připojení klienta. Tato nová kopie také změní svůj aktuální identifikátor uživatele z uživatele root na autentizovaného uživatele. (Jestliže se např. uživatel svoboda autentizuje u smbd, nová kopie bude běžet s právy uživatele svoboda, nikoli s právy superuživatele.) Kopie zůstává v paměti tak dlouho, dokud trvá připojení klienta.*

Démon nmbd je odpovědný za obsluhu požadavků jmenného serveru NetBIOS. Nejprve se spojí s portem 137; na rozdíl od smbd však nevytvoří pro každý dotaz instanci sama sebe. Kromě požadavků jmenného serveru zpracovává nmbd také požadavky od hlavních prohlížečů, prohlížečů domén a serverů WINS.“[5]

Samba se konfiguruje pomocí jediného konfiguračního souboru, nebo je k dispozici webové rozhraní SWAT (Samba Web Administration Tool), které je ovšem primárně vypnuto z bezpečnostních důvodů. Tento nástroje je v sambě až od verze 2.0, pracuje i bez webového serveru, jakým je například Apache.

[5] SHAH, Steve. *Administrace systému Linux: překlad čtvrtého vydání*. 1. vyd. Praha: Grada, 2007, 426 s. ISBN 978-80-247-1694-7.

2. Praktická část

Následující kapitola a její podkapitoly se již řadí mezi praktickou část bakalářské práce a autor se v nich bude zabývat praktickým řešením daných cílů práce.

2.1. Základní škola Rudná

Škola je rozdělena na dvě budovy, spojené chodbou, v novější je instalován server s Microsoft Windows server 2008 r2, na kterém je nainstalována služba Active Directory, která slouží pro přihlašování uživatelů, dále nezbytný DNS server a také server DHCP. Vzhledem k tomu, že v celé škole je přibližně 120 počítačů a z toho více, než polovina je volně přístupná žákům, je potřeba filtrovat přístup na různé nepřípustné internetové stránky.

K tomuto účelu se výborně hodí proxy server. Tento server dokáže filtrovat protokoly http, ftp a dále zabezpečené varianty https a ftps. Také je zde možnost zakázat například spouštění video souborů, spustitelných souborů, obrázků a tak dále. Zajímavým řešením je použití proxy serveru Squid, který je dostupný pod licencí GNU/GPL. Tento software je dostupný pro různé operační systémy včetně Microsoft Windows, ovšem v tomto případě je vhodnější použít systém GNU/Linux, protože je také pod licencí GPL a také z důvodu úspory financí na nákup hardwaru. Díky tomu, že se GNU/Linux dá nainstalovat přesně podle požadavku uživatele a tudíž i bez grafického rozhraní, které je v tomto případě zbytečné, dále je možné nakonfigurovat jádro systému tak, aby obsahovalo pouze nutné součásti, umožňuje použití výrazně slabšího hardwaru, než by bylo nutné pořídit pro běh Windows.

2.2. Výběr hardwaru

Výběr hardwaru pro server je poměrně jednoduchý. Linux není náročný na procesor, ani operační paměť, takže je možné použít například starší počítač, vyřazený z provozu. Tento počítač by ovšem neměl mít slabší konfiguraci, než uvádí výrobce použité distribuce operačního systému. Také je důležité, aby počítač obsahoval dvě síťové karty. Server by

měl oddělit celou síť od připojení k internetu tak, aby veškerá komunikace šla skrze něj. V praxi to vypadá například tak, že modem ADSL je připojen ze svého ethernetového portu do první síťové karty serveru a druhá karta tohoto serveru je připojena do switche v lokální síti.

V tomto případě bude využit vyřazený server, obsahující hardware:

- cpu: Intel Pentium 4 s frekvencí 1,8 GHz
- operační paměť: 1024 DDR s frekvencí 400 MHz
- pevné disky: Western Digital o kapacitě 500 GB v raidovém poli kvůli zrcadlení
- ostatní: server má jednu integrovanou síťovou kartu a jednu přidanou v PCI slotu, dále DVD mechaniku a disketovou jednotku

Tento server má parametry vyšší, než je potřeba, ale vzhledem k tomu, že jiný počítač není v tuto chvíli k dispozici, bude použit tento.

2.3. Výběr distribuce

Vybrat si distribuci v nepřeberném množství může být složité. Každá z nich má své výhody a nevýhody. Také záleží na tom, jak moc je uživatel seznámen s unixovými operačními systémy.

Pro začátečníky jsou zde distribuce jako je Ubuntu, které si čím dál tím více získává na oblibě pro svou jednoduchost a orientaci na obyčejné uživatele, ale pro některé nevyhovující grafické prostředí Unity. Ubuntu vychází z jedné z nejoblíbenějších distribucí Debian. Debian je unikátní v tom, že používá svůj instalační nástroj aptitude (dříve apt-get), se kterým výborně řeší různé závislosti mezi balíčky a je velmi snadné ho používat.

Mezi další oblíbené distribuce patří CentOS a OpenSUSE. CentOS vychází z komerčního RedHat Linuxu, ovšem oproti RedHatu obsahuje novější aktualizace aplikací, které ještě nemusí být odladěny a tak mohou v některých případech nečekaně ukončit svou práci. Tato distribuce také používá instalační nástroj RPM, který je v Linuxových systémech nejběžnější.

Distribuce OpenSUSE zase vychází z komerčního SUSE Linuxu, který vydává společnost Novell. Volně šiřitelná i komerční verze SUSE mají vydařený grafický instalátor Yast, pomocí kterého se zároveň konfiguruje systém.

Toto byly nejhlavnější distribuce, i když je ještě spousta dalších známých jako například Arch Linux, Mandriva a pro velmi pokročilé uživatele Gentoo, nebo Slackware. Škála distribucí je tedy opravdu velká a záleží na uživateli, co preferuje.

V tomto případě se uchýlíme k distribuci Debian ve verzi stable, která obsahuje již prověřené a stabilní balíčky aplikací. Sice má o něco složitější instalační postup než například Ubuntu, ale je zde možnost už při instalaci konfigurovat vše včetně jádra systému. Vzhledem k rychlosti připojení k internetu, která je zde nyní 24 Mbps přes linku VDSL, bude použita síťová instalace, která má pouhých 180 MB a tudíž se vejde na jedno CD. Instalační program si následně stáhne všechny potřebné balíčky z internetu. V následující kapitole bude popsána instalace a práce s aplikací aptitude.

2.4. Instalace operačního systému Debian

V první řadě je třeba vytvořit spouštěcí médium, ze kterého se bude zavádět instalační program. Jak již bylo řečeno, síťová instalace má pouhých 180 MB, takže lze použít obyčejné CD, nebo flash disk za předpokladu, že BIOS funkci bootování z USB podporuje. Dále je třeba, aby byla CD/DVD mechanika nastavena v BIOSU jako první médium, ze kterého se má zavádět systém. Pokud se na pevném disku nenachází předchozí instalace nějakého operačního systému, tento krok odpadá.

Nyní už se dostáváme k samotné instalaci operačního systému. Po nabootování z vybraného média se zobrazí krátké menu s možností instalace, nápovědy a položka *Options*, ve které si lze vybrat pokročilejší možnosti instalace. My se budeme zabývat instalací standardní. Dále následuje výběr jazyka instalačního programu, výběr země, ve které se počítač nachází a rozložení klávesnice. Tato tři nastavení bývají stejná u všech operačních systémů.

Při instalaci na systému na počítač s více síťovými kartami se instalátor ptá, kterou z nich má použít při instalaci. Samozřejmě je třeba vybrat tu, do níž je připojen kabel. Instalační program načte informace o nastavení sítě ze serveru DHCP a pokud vše proběhlo úspěšně, objeví se okénko pro zadání názvu počítače. Název může být jakýkoli jednoslovný, ovšem je vhodné vybrat takový, který bude alespoň částečně říkat, co je počítač zač. V tomto případě tedy bude název *proxy*. V následujícím okně zadáváme název

domény, ve které se bude počítač nacházet. Tento server bude přidán do již existující domény *zsrudna.local*.

Nyní přichází důležitá část, a to heslo pro uživatele *root*, což je Linuxový ekvivalent pro Administrátora. Toto heslo by neměl znát nikdo jiný, kromě správce systému, protože root má neomezená práva, tudíž může upravovat veškeré konfigurační soubory, nebo třeba cokoli smazat. Heslo je vhodné zvolit takové, které není ve slovníku a ideálně obsahuje číslice, malá a velká písmena a též může obsahovat speciální znaky.

V dalším kroku vyzívá instalátor k vytvoření uživatele, který ovšem nebude mít administrátorská práva. Obyčejný uživatel není v tomto případě nutný, protože ve valné většině případů bude potřeba použít oprávnění superuživatele root kvůli změnám v konfiguračních souborech.

Instalátor již získal potřebná data o uživatelích, jazyce a síti a přechází k rozdělení pevného disku. Je možné použít automatické rozdělení, ale pro jistotu zde vybereme ruční rozdělení, které dává více možností. První krok je vytvoření nové tabulky oblastí. Vybereme disk, na který bude instalován systém, a vytvoříme novou tabulku oblastí, která bude prázdná. V dalším kroku již vybereme volné místo a vytvoříme novou oblast o námi zadané velikosti. Této oblasti je poté třeba dát přípojný bod. V Linuxu je kořenový adresář značen znakem „ / “. Disk je také možno dělit tak, že například část přiřadíme adresáři */home*, který obsahuje data jednotlivých uživatelů. Tím se dá zamezit zaplnění disku jejich daty. V našem případě však postačí, když necháme volné 4 GB a zbytek připojíme ke kořenovému adresáři a použijeme systém souborů ext3. Na zbylých čtyřech GB místa vytvoříme takzvaný *swap*, což je odkládací prostor. Do swapu si systém ukládá data, která by se nevešla do operační paměti. Vytváření tohoto oddílu není nutné, ale pokud se zaplní operační paměť, systém havaruje. Posledním krokem je zapsání změn na disk, čímž dojde k jeho naformátování a vytvoření systému souborů.

Další částí instalace už je samotné instalování balíčků. Nejdříve je ale třeba vybrat mirror (zrcadlo) s archivem operačního systému Debian. Odtud si poté bude stahovat veškeré aplikace instalační nástroj aptitude. Nyní se dostáváme k výběru skupin balíků, které se mají instalovat. Pro potřeby proxy serveru bude stačit SSH server kvůli vzdálené správě přes protokol SSH, a standardní systémové nástroje. Systém po výběru balíků začne stahovat a instalovat vybraný software, což trvá různě dlouho v závislosti na způsobu

instalace (z DVD, nebo síťová). V případě síťové samozřejmě na rychlosti připojení k internetu.

Po nainstalování vybraných softwarových balíků přichází na řadu konfigurace zavaděče operačního systému. V novějších distribucích se lze téměř výhradně setkat pouze se zavaděčem *GRUB*. Dříve byl používán ještě zavaděč *LiLo* (Linux Loader). Oproti LiLo není nutné GRUB po změně nastavení znovu instalovat. Instalací zavaděče jsme u konce instalace operačního systému a přejdeme k instalaci aplikací, které budou potřeba pro běh proxy serveru a jeho integraci do doménové struktury Active Directory.

2.5. Instalace aplikací

2.5.1. Správce balíků

Jak již bylo řečeno, k instalaci a správě softwarových balíků v operačním systému Debian se používá nástroj aptitude, nebo starší apt-get. Obecně jsou takovéto nástroje nazývány *Správce balíků*. Tento software řeší instalaci a závislosti mezi balíky, odinstalování balíků, ale také update systému, aktualizace aplikací a v neposlední řadě vyhledávání aplikací v repositáři.

Odkazy na repositáře jsou uloženy v konfiguračním souboru *sources.list,* který se nachází v adresáři */etc/apt/*. Pokud se podíváme do tohoto souboru, ve většině případů zde bude několik řádků s odkazi na různá místa. Řádky, které mají na začátku znak mřížky „#“ jsou takzvaně zakomentované a správce balíků s nimi nebude pracovat. Soubor tedy může vypadat takto:

```
Deb http://ftp.cz.debian.org/debian squeeze main
#Deb-src http://ftp.cz.debian.org/debian squeeze main
Deb http://security.debian.org/debian squeeze/updates main
#Deb-src http://security.debian.org/debian squeeze/updates main
```

Řádky, začínající na Deb-src odkazují pouze na zdrojové kódy. Toto je vhodné pro programátory, kteří si nějakou aplikaci chtějí upravit podle sebe. V souboru, zobrazeném výše, jsou tyto zdroje zakomentovány, protože nebudeme pracovat se zdrojovými kódy.

Další zajímavostí jsou poslední dvě slova v řádku. V tomto případě první *squeeze* označuje název používané verze systému a druhé slovo *main* zase hlavní balíčky. K těmto slovům lze ještě přidat třeba *non-free,* nebo *unstable*. Pokud bychom tato dvě slova přidali, správce balíčků bude vyhledávat i mezi balíky, které ještě nejsou otestovány a prohlášeny za stabilní a dále v balíkách, které nejsou volně šiřitelné.

V konfiguračním souboru může také být odkaz na CD, ale pokud je možnost stahovat software z internetu, je vhodné tuto možnost zakomentovat.

2.5.2. Aptitude

Tento správce je založen na textovém rozhraní, na rozdíl například od Yastu v SUSE Linuxu. Obsahuje spoustu různých možností pro práci s balíky, ale my se podíváme pouze na ty nejdůležitější. Pokud má čtenář zájem, může se podívat na manuálové stránky k tomuto softwaru.

První a také nejdůležitější možností aptitude je parametr *install*. Z toho lze odvodit, že tímto příkazem se instalují vybrané softwarové balíky. Ovšem problém je v tom, že uživatel musí vědět přesný název balíku, jinak správce zahlásí chybu, že nemůže příslušný balík nalézt.

Ke hledání balíků a zjištění celých jejich názvů slouží parametr *search*, který prohledá všechny repozitáře podle souboru *sources.list* a vypíše výsledky na obrazovku. Poté již stačí zapsat příkaz: *aptitude install „nazev_baliku“* a správce už se postará o jeho instalaci včetně závislostí.

Samozřejmostí je také možnost smazání již nainstalovaného balíku a to se provádí pomocí parametru *remove,* který odstraní vybraný balík ze systému, nebo také *purge*, který odstraní balík i se všemi datovými soubory k tomuto balíku přidruženými.

Méně často používanými příkazy jsou *update, upgrade* a *dist-upgrade*. Update se používá v případě, že uživatel pozměnil konfigurační soubor s repositáři. Aptitude si po spuštění s tímto parametrem stáhne nový seznam dostupných balíků. Parametr upgrade je

používán pro aktualizace konkrétního balíku a dist-upgrade pro aktualizaci všech nainstalovaných balíků.

2.5.3. Instalace potřebných balíčků

První věc, kterou je vhodné provést, je použití příkazu *aptitude update*, aby si správce balíčků načetl seznam balíků a nehledal v místech, která máme zakomentovaná, popřípadě aby hledal v místech, která jsou nově přidána. Také použijeme dist-upgrade, abychom měli nejnovější verze stabilních balíků.

Teď už se pustíme do potřebných balíčků. První z nich, který bude potřeba stáhnout a nainstalovat je *Kerberos* včetně nutných knihoven. Použijeme balíky krb5-user a libkrb53. Příkaz bude vypadat takto: *aptitude install krb5-user libkrb53*.

Dále potřebujeme balíky proxy serveru, což bude v tomto případě *Squid* a LDAP. Nainstalujeme tedy balíky *squid3* a *ldap-utils* stejným způsobem, jak již bylo zmíněno výše.

Budeme také potřebovat balík Microsoft Keytab Utility, ovšem ten není součástí standardního repositáře, takže si ho stáhneme z internetu. K této utilitě je ještě nutno nainstalovat knihovny *libsasl2-modules-gssapi-mit libsasl2-modules*.

Také bude třeba nainstalovat balíky *Samba, Winbind* a hlavičkové soubory jádra systému. Jako poslední potřebujeme *negotiate wrapper*, který ovšem nelze získat v podobě balíčku, takže si stáhneme zdrojové kódy a přeložíme.

Zdrojové kódy bývají obvykle zabaleny v souborech *.tgz* a *.tar.gz*, který je nejprve třeba dekomprimovat, a to pomocí příkazu *tar -xzf nazev_balicku.tgz*. Za název balíčku můžeme ještě přidat cestu, kam se má obsah rozbalit. Parametr *-x* znamená extrahování, parametr *-z* je zde kvůli dekompresi programem gzip a konečně parametr *-f* udává název souboru. Když máme balík dekomprimovaný do adresáře, musíme vytvořit soubor, podle nějž bude následně program *make*, který vytváří ze zdrojových kódů binární balíčky, překládat zdrojový kód. Tento soubor vytvoříme tak, že se přepneme pomocí příkazu *cd* do adresáře rozbaleného balíku a zadáme příkaz *./configure*, který vygeneruje soubor pro kompilaci. Nyní přijde na řadu příkaz *make* a hned po něm *make install*, který přeložený program nainstaluje do systému.

Takto vypadá postup u konkrétního souboru:

cd /home/michal/Stažené
tar –xvzf negotiate_wrapper-1.0.1.tar.gz
./configure && make && make install

Tímto máme nainstalovány všechny potřebné balíky a nyní se můžeme podívat na konfiguraci jednotlivých aplikací.

2.6. Konfigurace aplikací

Začneme nejprve tím, že klientským stanicím nastavíme pomocí group policy na Windows serveru Integrated Windows Authentication. Díky této funkci bude prohlížeč Internet Explorer předávat zabezpečeně informace o uživateli pomocí protokolu Kerberos, nebo NTLMSSP (NT LAN Manager Security Support Provider) v případě selhání Kerbera. Toto je v podstatě jediné nastavení, které je potřeba udělat na straně Windows serveru, případně na jednotlivých klientských stanicích a dále už se budeme věnovat serveru s operačním systémem GNU/Linux.

2.6.1. Nastavení DNS a NTP u Linuxového serveru

DNS v debianu nastavíme v souboru *resolv.conf*, který se nachází v adresáři */etc/* jako ostatně většina konfiguračních souborů. V tomto souboru by měl být minimálně jeden řádek, v tomto případě bude vypadat takto: *nameserver 192.168.1.3*

Také je třeba editovat soubor */etc/network/interfaces*, kde nastavíme IP adresu tohoto Linuxového serveru, síťovou masku, výchozí bránu, použité DNS servery a také název domény. Výsledný soubor by měl vypadat přibližně takto:

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 192.168.1.5
        netmask 255.255.255.0
        gateway 192.168.1.1
        dns-nameservers 192.168.1.3
        dns-search zrudna.local
```

Nyní nastavíme Linuxový server tak, aby načítal data o aktuálním čase z doménového řadiče. Všechny počítače v doméně by měly mít stejný čas kvůli spolupráci. Pokud se tak nestane, může dojít k problému. Například by tak mohla vzniknout situace, že přijde e-mail několik minut před tím, než byl odeslán. Z toho důvodu se používá protokol NTP, pomocí kterého se dá synchronizovat čas na všech počítačích z jednoho serveru. Editujeme tedy soubor */etc/ntp.conf* a v tomto případě do něho zapíšeme řádek *server server2.zsrudna.local*. Po editaci tohoto souboru je potřeba restartovat službu ntp, což provedeme příkazem *service ntp restart*.

Můžeme vyzkoušet, zda náš server může komunikovat s doménovým řadičem a také jestli vidí do internetu. Toto lze provést příkazem *ping nazev_serveru*.

2.6.2. Konfigurace Kerbera

Protokol Kerberos je potřeba kvůli autentizaci vůči doménovému řadiči. Lze ho nakonfigurovat pro různé verze Windows serveru. V našem případě budeme provádět nastavení pro Windows server 2008 r2.

Konfigurační soubor Kerbera se jmenuje *krb5.conf*. Tento soubor editujeme podle požadavků doménového řadiče a výsledek bude vypadat takto:

```
[libdefaults]
        default_realm = ZSRUDNA.LOCAL

[# The following krb5.conf variables are only for MIT Kerberos.
        dns_lookup_kdc = no
        dns_lookup_realm = no
        ticket_lifetime = 24h
        default_keytab_name = /etc/squid3/PROXY.keytab

    ;Windows 2008
        default_tgs_enctypes = aes256-cts-hmac-sha1-96 rc4-hmac
des-cbc-crc des-cbc-md5
        default_tkt_enctypes = aes256-cts-hmac-sha1-96 rc4-hmac
des-cbc-crc des-cbc-md5
        permitted_enctypes = aes256-cts-hmac-sha1-96 rc4-hmac des-
cbc-crc des-cbc-md5


[realms]

ZSRUDNA.LOCAL  = {
        kdc = server2.zsrudna.local
        admin_server = server2.zsrudna.local
        default_domain = zsrudna.local
}

[domain_realm]

.zsrudna.local = ZSRUDNA.LOCAL

zsrudna.local = ZSRUDNA.LOCAL
```

V prvním oddíle *libdefaults* je nastavena defaultní domény, dále také životnost lístku pro klienta a různé použité enkrypce. V dalším oddíle definujeme server KDC a administrační server. Zde zastává obě tyto funkce jeden počítač.

Nyní máme částečně nakonfigurovaný Kerberos. Vyzkoušíme si tedy přidělení lístku TGT. Budeme potřebovat administrátorský účet, nebo účet, který má povoleno přidávat počítače do domény a spustíme příkaz *kinit administratorske_jmeno* a po výzvě zadáme heslo tohoto účtu. Tímto příkazem získáme nový lístek TGT ze serveru KDC a následně se můžeme na lístek podívat příkazem *klist.* V příkazové řádce by se měl objevit výpis, obsahující datum a čas přidělení lístku, expirace lístku a další informace.

V dalším kroku přidáme proxy server do domény pomocí softwaru *msktutil*. Účet počítače nesmí přesahovat povolenou hranici znaků. Počítač do Active Directory přiřadíme následujícím způsobem:

msktutil -c -b "CN=COMPUTERS" -s HTTP/proxy.zsrudna.local -k
*/etc/squid3/PROXY.keytab --computer-name PROXY-K *
--upn HTTP/proxy.zsrudna.local --server server2.zsrudna.local --verbose --enctypes 28

První dva parametry příkazu říkají, že se má vytvořit záznam do organizační jednotky *COMPUTERS*. Dalšími parametry se určuje metoda zápisu a umístění souboru *keytab*. Tento soubor obsahuje lokální zakódovaný klíč hostitelského počítače a neměl by k němu mít přístup nikdo, kromě superuživatele. Také je zde uvedeno jméno počítače, pod kterým se má vytvořit záznam a také User Principal Names, podle něhož se pak uživatelé mohou přihlašovat například e-mailovou adresou *stovicek@zsrudna.local*. Poslední tři parametry udávají server, na kterém se nachází Active Directory, dále podrobný výpis informací při provádění příkazu. Poslední parametr je určen jen pro Windows servery 2008. Přidává podporu pro kódování AES.

Pokud všechny předchozí kroky dopadly správně, zapsaly se nám do souboru */etc/squid3/PROXY.keytab* klíče. Je třeba, aby byl tento soubor čitelný pro Squid proxy server. Proto upravíme jeho práva příkazem *chgrp proxy /etc/squid3/PROXY.keytab* a *chmod 744 /etc/squid3/PROXY.keytab,* kde prvním příkazem změníme vlastníka daného souboru na *proxy* a druhým příkazem udělíme souboru práva. Vlastník má nyní právo číst, zapisovat i spouštět soubor a skupina s ostatními uživateli pouze číst tento soubor.

Nyní se zahodíme přidělený lístek, přidělený od Kerbara příkazem *kdestroy* a zapíšeme do souboru */etc/default/squid3/* místo, kde se nachází keytab. Toto je nutné kvůli Squidu, aby věděl, kde má hledat. Provedeme to tedy takto:

echo "export KRB5_KTNAME=/etc/squid3/PROXY.keytab" | tee /etc/default/squid3

Program *echo* vypisuje argumenty na standardní výstup a pomocí takzvané roury tento výstup slouží jako vstup do dalšího programu, kterým je *tee*. Ten poté zapíše argument z echa do souboru */etc/default/squid3*. Tím jsme dokončili konfiguraci Kerbera a začněneme s nastavováním Samby kvůli autentizaci NTLM (NT Lan Manager).

2.6.3. Samby a Winbind

Nejprve je vhodné služby Samba a Winbind zastavit, protože budeme dělat zásahy do jejich konfiguračních souborů. Provedeme to příkazy *service smbd stop* a *service winbind stop*.

Nyní editujeme konfigurační soubor Samby */etc/samba/smb.conf,* konkrétně část *[global]*. Soubor tedy bude vypadat podobně, jako tento:

```
#GLOBAL PARAMETERS
[global]
workgroup = ZSRUDNA
realm = ZSRUDNA.LOCAL
preferred master = no
server string = squid proxy server
security = ADS
encrypt passwords = yes
log level = 3
log file = /var/log/samba/%m
max log size = 50
printcap name = cups
printing = cups
winbind enum users = Yes
winbind enum groups = Yes
winbind use default domain = Yes
winbind nested groups = Yes
winbind trusted domains only = Yes
winbind cache time = 3600
winbind separator = +
;template primary group = “Domain Users”
template shell = /bin/bash
```

Prvními parametry je specifikována doména, *preferred master* musí být vypnutý kvůli Windows serveru, který je hlavním serverem, dále je zde zabezpečení Active Directory a zabezpečení hesla. Následuje logování a soubor, do kterého se bude zapisovat, tiskové ovladače. Dále jsou zde nastavení Winbindu. *Enum users* a *enum groups* jsou defaultně nastaveny na hodnotu *Yes*, doporučuji nechat, protože jinak mohou mít některé programy nesprávně. Také je zde použití standardní domény, používání pouze důvěryhodných domén a řádek *winbind nested groups*, který aktivuje podporu pro místní skupiny, nebo také aliasy, které pracují stejně, jako tytéž skupiny ve Windows.

Po této úpravě je možno přidat tento server do naší domény, a to příkazem *net ads join -U uzivatel_s_pravy_administratora.* Nyní znovu spustíme služby Samba a Winbind příkazy *service smbd start* a *service winbind start* a vyzkoušíme, zda vše proběhlo v pořádku pomocí příkazu *wbinfo -t,* který ověří, zda funguje účet, vytvořený po přidání Samba serveru do domény. Také můžeme vyzkoušet ověření pomocí obyčejného uživatele. Takový příkaz bude vypadat takto: *wbinfo -a nas_uzivatel --verbose* , kde verbose vypíše více informací.

Posledním krokem v nastavování Samby a Winbindu je přidání proxy do *Winbindd_priv* skupiny, což provedeme příkazem *gpasswd –a proxy winbindd_priv*. Toto je třeba udělat kvůli umožnění proxy číst z adresáře */var/run/samba/winbindd_privileged*. Program *gpasswd* se používá ke správě souboru */etc/group* a parametr *-a* přidává uživatele.

2.6.4. Squid

Než se pustíme do konfigurace Squidu, musíme vytvořit na Windows serveru doménového uživatele s minimálními právy, kterého budeme používat pro ověřování pomocí protokolu LDAP. Tento uživatel musí mít pevně dané heslo, které bude stále platné a tento uživatel ho také nemůže měnit. Toto heslo následně zapíšeme do souboru /etc/squid3/*ldappass.txt* pomocí příkazu *echo 'heslo‘ > /etc/squid3/ldappass.txt* a nastavíme k tomuto souboru práva tak, aby ho ostatní uživatelé nemohli číst: *chmod o-r /etc/squid3/ldappass.txt* a změníme znovu vlastníka na *proxy*: *chgrp proxy /etc/squid3/ldappass.txt.* Znakem „>“ v programu *echo* došlo k přesměrování standardního výstupu na soubor, uvedený zatím.

Nyní už se dostáváme k samotné konfiguraci Squidu. Editujeme tedy konfigurační soubor */etc/squid3/squid.conf*. Nejprve nakonfigurujeme autentizační mechanismy:

1) Autentizace Negotiate Kerberos a NTLM:

auth_param negotiate program /usr/local/bin/negotiate_wrapper -d --ntlm /usr/bin/ntlm_auth --diagnostics --helper-protocol=squid-2.5-ntlmssp --domain=ZSRUDNA --kerberos /usr/lib/squid3/squid_kerb_auth -d -s GSS_C_NO_NAME
auth_param negotiate children 10
auth_param negotiate keep_alive off

2) Autentizace NTLM:

auth_param ntlm program /usr/bin/ntlm_auth --diagnostics --helper-protocol=squid-2.5-ntlmssp --domain=ZSRUDNA
auth_param ntlm children 10
auth_param ntlm keep_alive off

3) Autentizace přes LDAP:

auth_param basic program /usr/lib/squid3/squid_ldap_auth -R -b "dc=ZSRUDNA,dc=LOCAL" -D drive_vytvoreny_uzivatel@zsrudna.local -W /etc/squid3/ldappass.txt -f sAMAccountName=%s -h server2.zsrudna.local
auth_param basic children 10
auth_param basic realm Internet Proxy
auth_param basic credentialsttl 1 minute

Dále si nadefinujeme Access Control List (ACL) pro výše vypsané autentizační metody: *acl auth proxy_auth REQUIRED* a nastavíme ověřování klientských systémů:

http_access deny !auth a *http_access allow auth*. Toto jsou základní řádky, které jsou nutné pro běh proxy serveru.

Nyní už je možné vytvářet pravidla podle potřeby. My budeme potřebovat pravidlo, které bude blokovat členy skupiny Active Directory, nazváne *G_zaci*. Provedeme to tímto způsobem:

```
        ### definujeme skupinu
        external_acl_type memberof %LOGIN
/usr/lib/squid3/squid_ldap_group -R -K -b "dc=ZSRUDNA,dc=LOCAL" -D
vyse_vytvoreny_uzivatel@zsrudna.local \
        -W /etc/squid3/ldappass.txt -f
"(&(objectclass=person)(sAMAccountName=%v)(memberof=cn=%g,,ou=Zac
i, dc=zsrudna,dc=local))" \
        -h server2.zsrudna.local

        ### Zde vytvoříme pravidlo pro skupinu Zaci
        acl clenove_skupiny_zaci external memberof Zaci
        acl hry dstdomain .zahraj.cz .onlinehry.sk .superhry.cz
        acl auth proxy_auth REQUIRED

        ### Zablokujeme neautorizované klienty
        http_access deny !auth

        ### Členům skupiny Zaci zamítneme přístup
        http_access allow hry !clenove_skupiny_zaci
        http_access allow auth
        http_access deny all
```

Pravidla jsou čtena po rádcích, takže pokud bychom dali na začátek souboru pravidlo *http_access deny all*, tak nikdo nebude mít přístup do vnější sítě. Pomocí Access Control Listů můžeme blokovat, nebo povolovat také soubory díky atribudu *urlpath_regex*, také můžeme mít seznam zakázaných, nebo povolených stránek v externím souboru, nebo třeba povolovat a zakazovat přístup do vnější sítě pro konkrétní IP adresy, nebo rozsah těchto adres.

Shrnutí

Proxy server se podařilo nakonfigurovat a zařadit do běžného provozu v ZŠ Rudná. Je třeba doladit za provozu Access Control Listy a vytvořit firewall pro filtraci i jiných protokolů, než http a ftp.

Autor si z této práce vzal spoustu nových informací ohledně operačního systému GNU/Linux, osvojil si práci v příkazové řádce a v neposlední řadě se dostal k zajímavým článkům, kterým se dále bude věnovat.

Závěr

Jak je vidět, konfigurace funkčního proxy serveru, založeného na platformě GNU/Linux, a jeho přidání do domény, fungující na operačních systémech od společnosti Microsoft, není úplně jednoduchá práce. Je nutné vzít v úvahu, že konfigurace celého proxy serveru zabere mnoho hodin a vyžaduje již hlubší znalost problematiky a také Unixových typů operačních systémů.

Cílem práce bylo nakonfigurovat operační systém GNU/Linux a potřebné aplikace k běhu proxy serveru. Tento krok se povedl, proxy server v této konfiguraci sice spolupracuje s doménovým řadičem, běžícím na operačním systému Windows, ale bylo by třeba vytvořit komplikovanější Access Control Listy, které by například dovolovaly vstup na některé stránky pouze v určitých hodinách kvůli zájmovým kroužkům. Mohou to být například online hry.

Dalším cílem bylo také použít již nakonfigurovaný server v prostředí základní školy. Tento cíl se sice povedlo dosáhnout, ale bude třeba delší monitorování provozu na síti, aby bylo možné nastavit proxy server podle potřeby. Také bude vhodné vytvořit firewall kvůli směrování portů a s tím spojené vzdálené správě serverů.

Nyní již záleží na správci dané sítě, zda je ochoten věnovat mnoho hodin konfiguraci takového proxy, nebo dá přednost koupi o něco rychlejšího počítače a použije jednodušší, ale dražší řešení za pomoci proprietárního softwaru, jakým je třeba ISA server od společnosti Microsoft.

Seznam použitých zdrojů

[1] GARMAN, Jason a Charles PERKINS. Kerberos: definitive guide. Vyd. 1. New York: O'Reilly, 2003, 253 s. ISBN 05-960-0403-6

[2] JACKIEWICZ,Tom a Charles PERKINS. Deploying OpenLDAP. Vyd. 1. New York: Distributed to the Book trade in the United States by Springer-Verlang, c2005, xxxii, 311 p. ISBN 15-905-9413-4

[3] RUSSEL, Charlie a Sharon CRAWFORD. Microsoft Windows Server 2008: velký průvodce administrátora. Vyd. 1. Brno: Computer Press, 2009, 1271 s. Administrace (Computer Press). ISBN 978-80-2115-3.

[4] SHAH, Steve. Administrace systému Linux: překlad čtvrtého vydání. 1. vyd. Praha: Grada, 2007, ISBN 978-80-247-1694-7.

[5] STREBE, Matthew a Charles Perkins. Firewally a proxy-servery. Vyd. 1. Brno: Computer Press, 2003, xxi, 450s. ISBN 80-722-6983-6

Internetové zdroje

[6] CINGROŠ, Milan. Squid. In: Squid [online]. 2007 [cit. 2013-03-13]. Dostupné z: http://www.abclinuxu.cz/software/server/proxy/squid

[7] Samba-opening windows to a wider world [online]. [cit. 2013-03-13]. Dostupné z: http://www.samba.org/

[8] SCHNIKOW, K. T. DHCP: Dynamic Host Control Protocol. *DHCP* [online]. 2006, 15-2-2006 [cit. 2013-03-04]. Dostupné z: http://www.abclinuxu.cz/slovnik/dhcp

[9] Slovníček pojmů: Proxy server. [online]. [cit. 2012-12-29]. Dostupné z: http://hosting.blueboard.cz/slovnicek-pojmu/proxy-server

[10] Základní informace o protokolu DHCP (Dynamic Host Configuration Protocol). Http://support.microsoft.com/?ln=cs [online]. 2011 [cit. 2013-03-13]. Dostupné z: http://support.microsoft.com/kb/169289/cs

Ostatní zdroje

Manuálové stránky použitého softwaru

Seznam obrázků