Nasazení bezdiskových pracovních stanic s OS Linux v síti FAI UTB

Implementation of diskless workstations running OS Linux in FAI UTB network

Petr Kohout

Bakalářská práce
2008

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav aplikované informatiky
akademický rok: 2007/2008

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Petr KOHOUT**
Studijní program:	**B 3902 Inženýrská informatika**
Studijní obor:	**Informační technologie**
Téma práce:	**Nasazení bezdiskových pracovních stanic s OS Linux v síti FAI UTB**

Zásady pro vypracování:

1. **Nabootujte lokalní stanici pomocí PXE.**
2. **Filesystém přimountujte přes NFS.**
3. **Umožněte uživateli při bootu výběr distribuce.**
4. **Uživatele autorizujte přes školní LDAP.**
5. **Berte v uvahu zabezpečení serveru i lokálních stanic.**
6. **Vypracujte podrobný popis postupu přípravy serveru a obrazu pro klienta.**
7. **Popište specifika používání.**

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. **ANVIN, H. Peter. PXELINUX – SYSLINUX for network boot [online]. c1994–2007 [cit. 2008–01–27]. Text v angličtině. Dostupný z WWW: http://syslinux.zytor.com/pxe.php.**
2. **Internet Systems Consortium, Inc.. Dynamic Host Configuration Protocol (DHCP) [online]. c2007 , 2007–12–19 [cit. 2008–01–27]. Text v angličtině. Dostupný z WWW: http://www.isc.org/index.pl?/sw/dhcp/.**
3. **SMITH, Christopher M.. Linux NFS faq [online]. 2008 [cit. 2008–01–27]. Text v angličtině. Dostupný z WWW: http://nfs.sourceforge.net/.**
4. **Cisco Systems, Inc.. Catalyst 2950 Desktop Switch Software Configuration Guide : 12.1(9)EA1 [online]. c1992–2007 , Sat May 05 09:48:18 PDT 2007 [cit. 2008–01–27]. Text v angličtině. Dostupný z WWW: http://www.cisco.com/en/US/docs/switches/lan/catalyst2950/software/release/12.1_9_ea1/co**
5. **Cisco Systems, Inc.. Catalyst 3550 Multilayer Switch Software Configuration Guide : Release 12.1(20)EA2 [online]. c1992–2007 , Sun Jul 01 05:16:48 PDT 2007 [cit. 2008–01–27]. Text v angličtině. Dostupný z WWW: http://www.cisco.com/en/US/docs/switches/lan/catalyst3550/software/release/12.1_20_ea2/c**
6. **Linux Kernel Organization, Inc.. The Linux Kernel Archives [online]. [1994] , 2008–01–24 23:18 UTC [cit. 2008–01–27]. Dostupný z WWW: http://www.kernel.org/.**

Vedoucí bakalářské práce: **Ing. Martin Sysel, Ph.D.**
Ústav aplikované informatiky

Datum zadání bakalářské práce: **20. února 2008**

Termín odevzdání bakalářské práce: **5. května 2008**

Ve Zlíně dne 20. února 2008

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ Fakulta aplikované informatiky L.S.

prof. Ing. Vladimír Vašek, CSc.
děkan

doc. Ing. Ivan Zelinka, Ph.D.
ředitel ústavu

ABSTRAKT

Tato bakalářská práce popisuje zavedení operačního systému Linux na pracovní stanice v počítačových učebnách FAI UTB tak, aby nebylo nijak narušeno současné nastavení těchto stanic.

V teoretické části jsou zmíněny principy, výhody a nevýhody řešení, kdy je celý kořenový souborový systém přístupný pouze na síti.

Praktická část ukazuje nasazení tohoto řešení přímo do některých učeben, včetně konkrétních ukázek nastavení sítě a výstavby linuxového operačního systému.

Klíčová slova: Linux, PXE, DHCP, NFS

ABSTRACT

This bachelor work describes running of operating system Linux on workstations in computer rooms of FAI UTB without any changing of current settings of these workstations.
Theoretical part contains principes, advantages and disadvantages of such solution, where the root filesystem is available only on network.
Practical part shows implementation of this solution directly in some computer rooms and includes network configuration and instructions of building of proper Linux-based operating system.

Keywords: Linux, PXE, DHCP, NFS

Děkuji správci sítě ing. Petru Vojtkovi za možnost nahlédnout do konfigurace současné sítě UTB a možnost podílet se na jejím vývoji a také děkuji vedoucímu bakalářské práce ing. Martinu Syslovi, PhD. za cenné náměty a připomínky.

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

V Uherském Hradišti, 15. května 2008

.....
Podpis diplomanta

Obsah

Část I
ÚVOD

Cílem této práce je nasazení operačního systému Linux na pracovní stanice v počítačových učebnách v budově U5 UTB tak, aby nebyla nijak měněna jejich současná konfigurace a aby byly náklady na výstavbu a údržbu takového systému co nejnižší.

Současný stav pracovních stanic je takový, že na běžných pracovních PC stanicích je nainstalován operační systém Microsoft Windows. Každá stanice má přidělenou vlastní IP adresu a v případě poškození disku nebo operačního systému technik obnoví původní stav pomocí bootovacího CD disku, kdy na pevný disk nahraje obraz již hotového předinstalovaného operačního systému a dále jen minimálně upraví identifikaci počítače v síti.

Vzhledem k nutnosti zachovat stávající řešení správy počítačů s operačním systémem Microsoft Windows a potřebě dostupnosti jiného operačního systému (konkrétně OS Linux) byla zvolena možnost zavedení OS Linux z ethernetové sítě, která je pro každé PC k dispozici. Každé PC bude nastaveno tak, aby primárně bootovalo nikoliv z pevného disku, ale pomocí protokolu PXE ze sítě. PXE dá uživateli možnost si vybrat, zda chce dále pokračovat ve standardním bootovacím procesu (tedy nahrání Microsoft Windows z pevného disku) nebo zda se má nahrát OS Linux ze sítě.

V případě zvolení bootování z OS Linux dojde k tomu, že do paměti počítače se protokolem TFTP nahraje linuxové jádro, které dále pokračuje ve startování operačního systému. Kořenový adresář celého operačního systému se pak během startovacího procesu připojí protokolem NFS. Tím dojde de facto k tomu, že na stanici poběží v paměti OS Linux a celý kořenový souborový systém bude fyzicky uložen na vzdáleném serveru - PC bude mít k tomuto souborovému systému přístup pouze pro čtení.

Část II
TEORETICKÁ ČÁST

1 Stručný popis řešení

1.1 Popis sítě a umístění síťových služeb

Nezbytným předpokladem k provozování jakýchkoliv bezdiskových stanic, které bootují ze sítě, je správné fungování vlastní sítě a nastavení síťových služeb. Síť je navržena tak, že každé učebně je přiřazen jeden adresní prostor, který bude využíván dynamicky (to zajistí síťová služba *DHCP*), všechny tyto sítě pak jsou zakončeny na jednom páteřním prvku (**směrovači**), který zajišťuje tyto služby:

- *DHCP* - směrovač si spravuje přidělování IP adres jednotlivým stanicím ve všech sítích v učebnách
- *routing* - správné směrování paketů mezi jednotlivými učebnami, dalšími lokálními sítěmi, Internetem a sítí s hlavním serverem
- *firewall* - na směrovači jsou základní omezení provozu pro zvýšení bezpečnosti směrovače a jednotlivých sítí

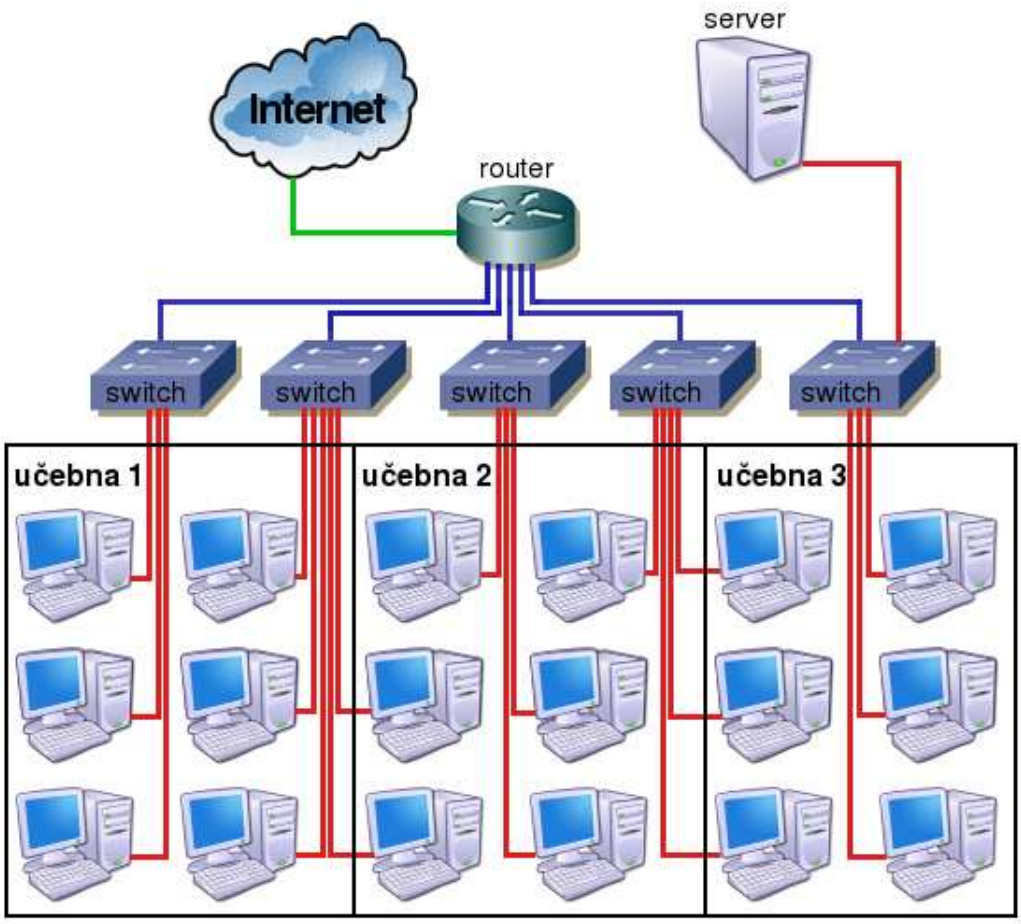
Směrovač bude tedy spojovat fyzickou i logickou strukturu sítě a bude stanicím poskytovat přípojení k Internetu. Kromě toho bude ke směrovači připojen **centrální server**, který bude poskytovat další nezbytné služby:

- *TFTP* - poskytuje stanicím nastavení PXE bootování a obrazy linuxového jádra
- *NFS* - poskytuje celý souborový systém pro OS Linux

1.2 Logická a fyzická topologie sítě

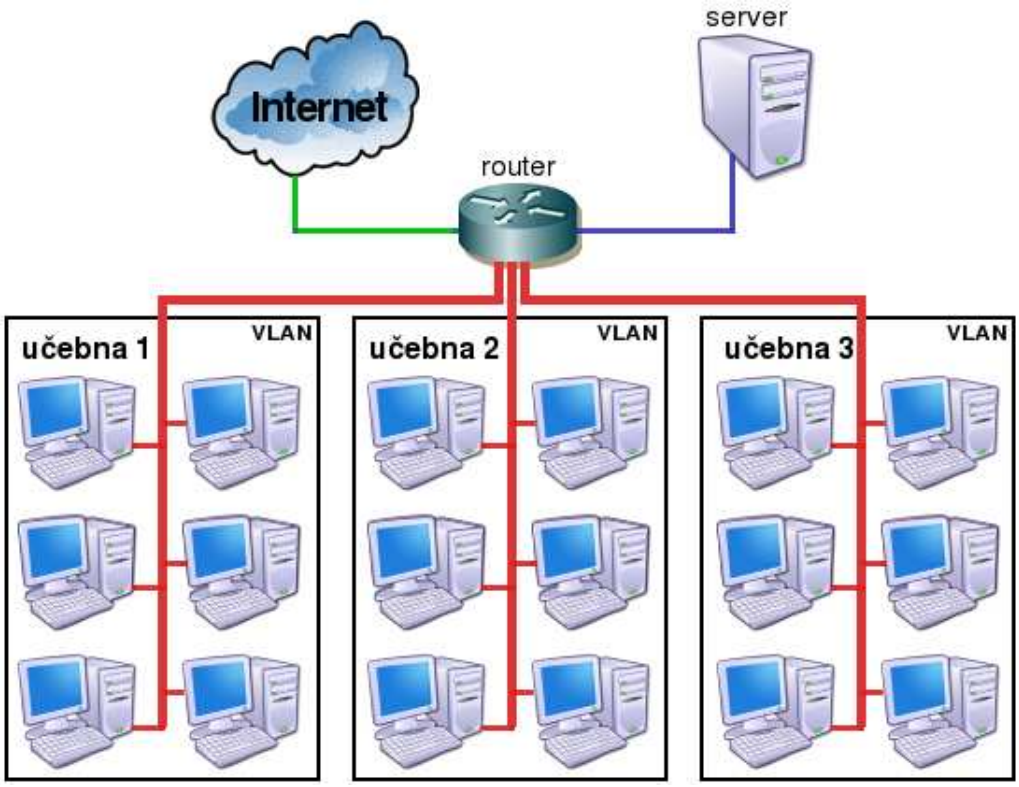
Ze současných síťových prvků na FAI UTB jsou použity prvky **Cisco 2950** jako přístupové přepínače pro učebny a L3 přepínač [1] **Cisco 3550** jako směrovač. Server je počítač s operačním systémem Linux.

[1] přepínač, který umí směrovat pakety mezi různými sítěmi a tedy funguje i na 3. vrstvě modelu OSI



Obrázek 1: Fyzická topologie sítě

Na obrázku je vidět hierarchická struktura sítě. Hlavní páteřní zařízení (směrovač Cisco 3550) je propojeno s přístupovými přepínači (Cisco 2950), do kterých jsou pak dále připojena koncová zařízení (pracovní stanice, servery).



Obrázek 2: Logická topologie sítě

Implementací virtuálních sítí (*VLAN*) jsou vytvořeny logické skupiny zařízení, která k sobě patří, tedy počítače v každé učebně tvoří samostatnou síť, server je také v samostatné síti. Takové rozdělení umožňuje snadnou manipulaci s učebnou jako s jedním objektem, zvýšení bezpečnosti serveru a snížení nechtěného provozu na síti.

1.3 Postup bootování pracovní stanice

1.3.1 Nastavení BIOSu

Nastavení BIOSu se nijak neliší od běžného nastavení pracovní stanice, jediná nutná změna je seznam zařízení, ze kterých se má počítač nabootovat. Současné nastavení umožňuje nabootování pouze z pevného disku (z bezpečnostních důvodů není pro běžné uživatele možné nabootovat pomocí CD-ROM nebo USB flash disku). Aby bylo umožněno bootování ze sítě, musí být tedy jako první zařízení nastaveno právě bootování pomocí PXE (v případě síťové karty přímo na základní desce musí být samozřejmě tato síťová karta v BIOSu zapnuta) a bootování z pevného disku je nastaveno až jako druhá možnost. Stejně jako dosud nebude umožněno bootování z jiného zařízení.

Uživatel samozřejmě nemá do nastavení BIOSu přístup (aby nemohl nastavení měnit), ten je chráněn heslem, které je známé pouze správcům počítačů.

1.3.2 Bootování po síti pomocí PXE

Po startu počítače a ukončení POST [2] přebírá běh počítače PXE (program, který je instalován výrobcem přímo v chipsetu síťové karty):

- pomocí protokolu *DHCP* nejprve zjistí IP adresu a masku sítě, IP adresu bránu, IP adresu *TFTP* serveru a název hlavního zaváděcího souboru
- po nastavení parametrů sítě se protokolem *TFTP* stáhne a zavede hlavní soubor pro PXE
- uživatel u pracovní stanice má nyní na výběr, zda si vybere zavedení OS Linux nebo bude pokračovat v bootování z pevného disku, kde je nainstalován OS Microsoft Windows

Celý tento postup ovšem závisí na tom, zda je funkční síť a všechny nutné síťové služby. V případě, že síť není dostupná vůbec (např. do počítače není zapojen síťový kabel nebo není funkční přístupový přepínač) nebo na síťovém segmentu nefunguje služba *DHCP*, dojde k vypršení časového limitu odeslané DHCP žádosti. Program PXE tedy oznámí BIOSu, že není schopen do počítače zavést operační systém a BIOS pak bootuje dál podle nastaveného pořadí, tedy spustí systém na pevném disku, kde je nainstalovaný systém Microsoft Windows.

K problému může také dojít v případě, že není dostupná služba *TFTP* - PXE si nemůže stáhnout zaváděcí soubor ani jeho nastavení a opět dojde k předání řízení počítače BIOSu.

1.3.3 Zavádění OS Linux

V případě volby OS Linux dojde ke stažení linuxového jádra (opět ze serveru protokolem *TFTP*) a jeho nahrání do paměti. PXE v tento moment končí a předává řízení počítače linuxovému jádru.

Jádro nejprve provádí standardní věci jako testy procesoru, paměti a existence dalšího hardwaru, poté si pomocí protokolu *DHPC* zjistí nastavení sítě (vlastní IP adresu, masku sítě, IP adresu brány) a pomocí protokolu *NFS* si na kořenový adresář souborováho systému připojí příslušný adresář obsahující celý souborvý systém. Hlavní adresáře (např. `/etc`, `/lib`) jsou tedy fyzicky uloženy na serveru, stanicím jsou přístpupné pouze pro čtení protokolem *NFS*.

[2] Power-on Self-test, samotestovací pogram probíhající na počítači při startu před bootováním operačnho systému

Jádro po zavedení a připojení kořenového adresáře pak předá řízení programu **init**, který dále provádí spouštění startovaích skriptů na stanici a umožňuje obsluhu počítače uživateli.

V tomto postupu jsou dva kritické body: stahování linuxového jádra protokolem *TFTP* a připojování souborového systému protokolem *NFS*. V případě chybného stažení linuxového jádra (chybějící soubor na TFTP serveru, pád TFTP serveru, poškození souboru během přenosu po síti atd.) se PXE vrací do hlavního menu, kdy dává uživateli možnost volby operačního systému. V případě, že se linuxové jádro stáhne, přebírá řízení počítače již jádro a program PXE je ukončen - celý bootovací proces je tedy završen a v případě chyby Linuxu se již počítač nemůže vrátit do BIOSu a zavádět jiný operační systém. Stejně tak je to u druhého kritického bodu, tedy připojování souborového systému protokolem *NFS* - v případě nefunčnosti NFS serveru (nebo jakékoliv jiné kritické chyby nutné pro chod OS Linux) linuxové jádro jednoduše zahlásí kritickou chybu **"Kernel panic"** a svůj běh zastaví. V takovém okamžiku již není možné s počítačem cokoliv dělat a počítač se musí restartovat.

1.4 Výhody a nevýhody

Mezi hlavní výhody tohoto řešení patří snadná udržovatelnost celého operačního systému:

- správce systému si spravuje celý souborový systém na serveru, soubory jsou po síti přístupné pouze v režimu *pouze pro čtení*, takže nehrozí poškození souborového systému ze strany uživatele
- souborový systému je snadné zálohovat nebo kopírovat, což umožňuje dělat různé pokusy s instalací nových programů
- na serveru může být samozřejmě více operačních systémů ve více verzích, různé operační systémy, různé linuxové jádra atd.

Snadné je i udržování pracovní stanice. Vzhledem k tomu, že počítač pracuje v bezdiskovém režimu, stačí pouze správně nastavit bootování a o vše ostatní je postaráno, vše již má na starosti server. Správce tedy nemusí obcházet jednotlivé stanice s instalačním CD nebo na ně vůbec něco instalovat.

Využitím protokolu *DHCP* se také sníží náročnost správy nastavení sítě jednotlivých stanic - pracovní stanice si síť nastaví jednoduše samy, IP adresa je jim přidělena dynamicky z příslušného rozsahu.

Celé toto řešení však má i své nevýhody: vyšší zátěž celé sítě a také serveru. To se může projevit především v okamžiku nabootování většího počtu stanic najednou, např. při obnovení funkčnosti elektrické sítě po předešlém výpadku. V takovém případě dochází ke zvýšení počtu požadavků na DHCP server, TFTP server i na NFS server.

Dalším nedostatkem je zvýšení provozu *DHCP paketů* - při každém bootování stanice do OS Linux vysílá stanice 3x požadavek na DHCP:

- PXE (potřebuje nastavit síťovou kartu, aby vůbec stanice mohla vzdálené bootování provádět)
- linuxové jádro před připojením souborového systému (potřebuje znát nastavení sítě, aby mohlo připojit souborový systém protokolem *NFS*)
- DHCP client - software běžící po rozběhnutí OS Linux (udržuje v běžícím systému informace o nastavení sítě, pravidelně sleduje vypršení doby platnosti IP adresy atd.)

Bohužel mezi těmito třemi objekty není možné informace o nastavení sítě předávat, takže dochází k tomu, že DHCP server musí během poměrně krátké doby bootování vyřídit požadavek od jedné stanice několikrát.

Vysoké zátěži je navíc podroben centrální NFS server. Při připojení většího množství pracovních stanic a jejich možným různorodým požadavkům na jednotlivé soubory na nejrůznějších místech souborového systému představuje tento server kritický bod - konkrétně zatížení disku by mohlo být větší než únosná míra. Aby byl přístup protokolem *NFS* maximálně urychlen a tím nedošlo ke zpomalování práce jednotlivých stanic, leží celý poskytovaný souborový systém na *ramdisku* - v paměti serveru je vyčleněno dostatečně velké množství paměti, do které je souborový systém nakopírován. Přístup pracovních stanic k jednotlivým souborům je pak zpracováván pouze v paměti, bez jakékoliv práce s disky na centrálním serveru.

I toto dílčí zrychlení má ovšem své nedostatky, které vyplývají z vlastností ramdisku: po rebootu serveru je paměť prázdná a je třeba zajistit, aby byl vytvořen a naformátován ramdisk a aby byl do něho celý souborový systém nakopírován. To ovšem znamená zdržení. Další nevýhodou ramdisku je požadavek na větší velikost paměti serveru, zvlášť v případě, že je k dispozici nekolik operačních systémů nebo jejich verzí.

Poslední velkou nevýhodou je množství síťových služeb, které musí bezchybně fungovat: DHCP, TFTP, NFS, směrování, přepínání a firewally - úplná nefunkčnost nebo chybné fungování kterékoliv z těchto služeb znamená nefunkčnost celého systému, proto je nutné dbát na správný chod všech zařízení (týka se jak hardwaru, tak softwaru), které jsou do celého systému zapojeny.

Vyjmenované nevýhody tohoto řešení spočívají především v náročnosti na použitý hardware, síťové zařízení a kvalitu sítě vůbec. Lze konstatovat, že všechna zařízení, která jsou použita na síťové služby, jsou natolik kvalitní, že by všechen běžný provoz měla zvládnout bez potíží.

2 Nastavení sítě

2.1 DHCP

DHCP je protokol umožňující konfigurovat (nejen) síť na různých počítačích v síti pouze z jednoho místa. Protokol je typickou ukázkou vztahu klient-server, kdy v síti existuje **DHCP server** a obvykle více **klientů**.

Přístup klientské stanice k síti není nejprve vůbec nijak nakonfigurován, ale v okamžiku, kdy stanice potřebuje na síti jakkoliv komunikovat, musí vědět svoji IP adresu, masku sítě a další podrobnosti. Při použití protokolu *DHCP* vyšle klient žádost, na kterou očekává odpověď. Je-li v síti DHCP server, který požadavek klienta zachytí a zpracuje, pošle odpověď s veškerými informacemi, které klient potřebuje vědět.

V tomto případě potřebuje znát pracovní stanice tyto údaje:

- vlastní IP adresu stanice
- dobu platnosti IP adresy
- masku sítě
- IP adresu brány
- IP adresu TFTP serveru
- název zaváděcího souboru pro PXE na TFTP serveru

IP adresa TFTP serveru, stejně jako název souboru pro PXE jsou ve všech odpovědích shodné, IP adresa stanice je přidělována dynamicky z příslušného rozsahu, stejně tak IP adresa brány a maska sítě. Doba platnosti IP adresy byla stanovena na 30 minut - před uplynutím této doby musí klient požádat o obnovu IP adresy, pokud by se tak nestalo, může být tato adresa poskytnuta jinému klientovi. Tato doba je rozumným kompromisem mezi zbytečným zatěžováním sítě častými DHCP dotazy a odpověďmi a příliš dlouhou dobou blokování adres, které mohou být přiděleny již odpojeným zařízením.

2.1.1 Zabezpečení DHCP

Jsou-li na jednom segmentu sítě dva (nebo více) DHCP servery, může dojít k tomu, že klient obdrží dvě odpovědi na svůj požadavek o konfiguraci pomocí *DHCP*. V takovém případě si klient může vybrat, kterou odpovědí se bude řídit, dle svého vlastního uvážení [1]. Potenciální útočník by tedy mohl snadno do sítě připojit vlastní počítač s vlastním DHCP serverem, čímž by mohl získat naprostou kontrolu nad všemi daty, které se po síti posílají [11].

Aby se takovému útoku zamezilo, je na přístuponých přepínačích implementován *DHCP snooping* [2], technologie, která umožňuje tiše zahazovat (tedy dále neposílat) DHCP odpovědi od zařízení, která nejsou povolena. Konkrétně na každém přepínači musí být označen fyzický ethernetový port, ze kterého může do přepínače přijít DHCP odpověd - v tomto případě je to na každém přístupovém přepínači pouze port, kterým je přepínač propojen se směrovačem. Dojde-li přepínači na nepovoleném portu DHCP odpověď, je úplně ignorována.

2.2 TFTP

Protokol *TFTP* slouží k přenášení souborů mezi počítači bez jakékoliv autentizace pomocí UDP paketů. Na serveru jsou umístěny soubory nutné k úspěšnému zavedení ovládacích souborů pro PXE:

- **pxelinux.0** - hlavní ovládací soubor pro PXE
- **vesamenu.c32** - umožňuje grafické menu v PXE
- **bg.jpg** - pozadí v grafickém menu
- **pxelinux.cfg/default** - nastavení jednotlivých položek v menu

Kromě souborů pro PXE jsou na TFTP serveru přístupné také linuxová jádra a samozřejmě mohou být přístupné i obrazy různých operačních systémů.

2.2.1 Bezpečnostní rizika TFTP

Protokol *TFTP* v sobě nemá žádnou autentizaci, každý, kdo se připojí k serveru, může stahovat libovolný soubor, o jehož existenci ví (nelze listovat soubory nebo adresáře). Rizikem může být úniku dat z TFTP serveru k útočníkovi, ale obsah těchto dat (konfigurace PXE, obraz operačních systému apod.) nejsou příliš citlivá data, takže toto hledisko lze pominout.

Protokolem TFTP lze ovšem nejem číst data ze serveru, ale je jím možné i soubory na serveru vytvářet nebo přepisovat, což už bezpečnostním rizikem rozhodně je. Server je tedy nastaven tak, aby nové soubory nebylo možné vytvářet.

2.3 NFS

NFS je protokol pro sdílení souborů a adresářů mezi jednotlivými počítači na síti, v provedení klient-server. Server poskytuje část své adresářové struktury klientovi, který si může poskytovaná data připojit do lokálního adresáře.

Linuxový server má v adresáři `/mnt/sda7/disks` adresáře s jednotlivými distribucemi, např. `/mnt/sda7/disks/suse-10.3-mini` nebo `/mnt/sda7/disks/debian-4.0`. Celý adresář s distribucemi je pak nasdílen ke čtení počítačům v síti.

Klient, který chce pak používat některý operační systém, si pak přímo na svůj kořenový adresář připojí adresář ze serveru:

```
client# mount -t nfs server:/mnt/sda7/disks/debian-4.0 /
```

Adresář, který je na serveru jako `/mnt/sda7/disks/debian-4.0/etc` je pak na stanici viditelný jako `/etc`.

2.3.1 NFS a bezpečnost

Protokol NFS umožňuje sdílet adresáře i v režimu **read-write**, což je v tomto případě neprosto nežádoucí. Server tedy musí být nastaven tak, aby umožňoval pouze **read-only** přístup ke svým souborům.

Soubory a adresáře jsou přístupné každému, kdo se k NFS serveru připojí, nicméně je možné tento přístup omezit pouze na určitý seznam IP adres.

2.4 Bezpečnost

2.4.1 Zabezpečení směrovače a přepínačů

Směrovač je centrálním komunikačním prvkem mezi pracovními stanicemi a serverem - nebude-li fungovat, nebudou fungovat ani stanice, protože k jakékoliv akci, při které by potřebovaly číst z disku, nyní čtou přes síť ze serveru. V případě nefunkčního směrovače jsou všichni uživatele bezmocní a proto je třeba se jeho zabezpečení dostatečně věnovat.

Hlavní zabezpečení je omezení přístupu k administraci síťových prvků - konfigurovat je lze pouze připojením konzole na sériový port a zadáním hesla (to ovšem vyžaduje fyzický přístup do zamčené serverovny) nebo vzdáleným přístupem po síti, ovšem pouze administrační virtuální síti, kterou nemají stanice v učebnách k dispozici.

U přístupových přepínačů je pak třeba dbát na kontrolu dat, která připojená koncová zařízení do sítě posílají, především je třeba se bránit proti útokům na ARP tabulku pomocí *port-security* [3]. Tato technologie zajistí vypnutí přístupového portu přepínače v případě, že útočník odesílá velké množství rámců s různými MAC adresami (tím může dojít k zaplnění ARP tabulky na přepínači, čímž je degradován na *hub* a útočník se tím pádem dostane k veškerému provozu na síti [11]).

Další důležitou ochranou přístupového bodu je puštění *BPDU guard* [3], cože je funkce, která zabraňuje přijímání BPDU rámců [3] na portech označených jako *port-fast* (předpoklad, že připojené zařízení do tohoto portu přepínače bude již koncová stanice a ne další přepínač). V případě přijetí takového rámce se jedná buď o nechtěný zásah do páteřní infrastruktury (což může vést teoreticky až ke kolapsu celé sítě) nebo o cílený útok (útočník by pak mohl nepozorovaně zachytávat úplně všechen provoz na síti [11]).

2.4.2 Ochrana serveru

Přístup k serveru je možný pouze z klávesnice (opět je nutný fyzický přístup do serverovny) nebo vzdáleně šifrovaným protokolem *SSH*, to vše se znalostí názvu uživatele a hesla.

Na serveru je nainstalován firewall, který zahazuje jakékoliv pakety, které nejsou explicitně povoleny. Mezi povolený provoz patří:

- SSH (port `22/tcp`) z omezeného množství IP adres
- TFTP (port `69/udp`) pouze ze sítí v učebnách
- NFS (port `777/udp`) pouze ze sítí v učebnách

I jednotlivé programy na serveru mají další vlastní omezení přispívající ke zvýšení bezpečnosti, které se mohou uplatnit v případě nefunkčnosti lokálního firewallu:

- TFTP server pracuje pouze v režimu read-only
- NFS server pracuje pouze v režimu read-only
- NFS server poskytuje data pouze klientům ze sítí v učebnách

2.4.3 Kontrola paketů v síti

Kromě lokálního firewallu na serveru existuje ještě firewall na směrovači, který nepouští k serveru jiné než povolené pakety - de facto se jedná o obdobu serverového firewallu.

Tento firewall lze do budoucna využít i ke kontrole nebo omezení dat směrujících z učeben, do učeben nebo mezi jednotlivými učebnami.

[3] součást protokolu STP, který slouží k zabraňování packet stormů ve velkých přepínaných sítích a určování tras na druhé vrstvě OSI

3 Pracovní stanice

3.1 Nastavení BIOSu

V podstatě jediné nastavení v BIOSu pracovních stanic je správné pořadí bootování. Pracovní stanice je nastavena tak, že je povoleno pouze bootování ze sítě a z pevného disku (pro případ, že by z jakéhokoliv důvodu síť nebyla k dispozici). Přístup k tomuto nastavení je chráněn heslem, aby ho mohli měnit pouze správci (především při opravě lokální instalace OS Microsoft Windows na pevném disku, kdy je nutné bootovat z CD).

3.2 PXE menu

Při bootování ze sítě si PXE stáhne konfigurační soubor, ve kterém jsou obsaženy jednotlivé položky menu. Implicitní volba je pokračování v bootování z pevného disku, tedy naběhnutí lokálního operačního systému Microsoft Windows. Ostatní volby jsou předpřipravené verze operačního systému Linux umístěné na centrálním serveru.

Od okamžiku zobrazení menu má pak uživatel 4 sekundy čas na to, aby si vybral jiný než předvolený operační systém (čas se zastaví stisknutím libovolné klávesy). Pak má dalších 30 sekund na to, aby si pomocí šipek vybral jinou položku v menu.

3.3 Linuxové jádro

V případě výběru OS Linux se pomocí protokolu *TFTP* stáhne obraz linuxového jádra. Jádro pak přebírá běh počítače - začne detekcí procesoru, pamětí, hardwaru atd. Důležtou věcí nakonec je možnost připojení linuxového souborového systému přes síť - jádro vyšle DHCP dotaz na nastavení sítě, pak nastaví síťové rozhraní a nakonec připojí kořenový souborový systém a v něm pak standardně spustí program `init`.

3.4 Běh OS Linux

Celý spuštěný operační systém se pak nadále chová jako běžný nabootovaný OS Linux - `init` zajistí spuštění základních startovacích skriptů, spuštění softwarových služeb a umožní uživateli přihlášení a běžnou práci.

Část III
PRAKTICKÁ ČÁST

4 Konfigurace síťových prvků

4.1 Přístupové přepínače Cisco 2950

Všechny zde uvedené konfigurace přepínače Cisco 2950, včetně jejich popisů, jsou k dispozici v manuálnu k tomuto zařízení [3].

4.1.1 Důležité globální nastavení a zabezpečení

Hlavní zabezpečovací globální direktivy zajišťují šifrované uložení hesla [4], omezení přístupu pouze z konzole nebo z omezeného počtu IP adres [5] a použití *BPDU Guard* [3] jako ochranu proti možnému ohrožení nastavení virtuálních sítí.

```
service password-encryption
enable secret 5 $1$5UYo$wO7xFXanBb0e5XRvAfOmQ0
access-list 8 permit 10.0.5.1
aceess-list 8 permit 10.0.5.11
spanning-tree portfast bpduguard default
line con 0
  password 7 1314001719181D
  login
line vty 0 15
  access-class 8 in
  password 7 1314001719181D
  login
```

Kromě těchto obvyklých zabezpečovacích metod je nutné také zapnout *DHCP snooping* [3], tedy zahazování DHCP paketů typu *DHCPOFFER* již na druhé vrstvě OSI, stačí pouze pro ty sítě, které jsou na učebnách a protokol *DHCP* je zde nutné používat.

```
ip dhcp snooping vlan 781 782 783
no ip dhcp snooping vlan option
```

4.1.2 Nastavení přístupových portů

Přístupové porty jsou ty porty na přepínači, do kterých jsou již připojena koncová zařízení (tedy přímo jednotlivé počítače na učebnách). Podle toho, ve které učebně se počítač fyzicky nachází, je port nastaven přímo v dané virtuální síti.

[4] v uvedeném příkladu je zašifrované heslo **qwerty**

[5] seznam je uveden v access-listu č. 8

V nastavení portu lze vidět několik zabezpečení:

- přiřazení portu do konkrétní virtuální sítě
- nastavení *port-security* [3], tedy odražení útoku, kdy je na přepínač posíláno většího množství MAC adres [11]
- označení portu jako *port-fast* [3], což zabraňuje útoku na protokol *STP* v souvislosti s globálním zapnutím *BPDU guard*
- omezení počtu vysílání DHCP žádostí

```
interface FastEthernet0/3
 description Workstations - 305/13
 switchport access vlan 781
 switchport mode access
 switchport port-security
 switchport port-security maximum 2
 switchport port-security aging time 2
 switchport port-security violation restrict
 switchport port-security aging type inactivity
 spanning-tree portfast
 ip dhcp snooping limit rate 16
!
```

4.1.3 Nastavení uplinku

Nastavení portu pro uplink již neobsahuje tolik restrikcí, nicméně vzhledem k používání DHCP snoopingu je nutné na tomto portu povolit DHCP odpovědi (na tomto portu budou přicházet od DHCP serveru). Dalším malým krůčkem k vyšší bezpečnosti je vyjmenování jednotlivých virtuálních sítí, které na uplinku povolujeme. [6]

```
interface FastEthernet0/24
 description Uplink
 switchport trunk allowed vlan 50,781,782,783
 switchport mode trunk
 ip dhcp snooping trust
!
```

4.2 Směrovač Cisco 3550

Páteřní směrovač slouží jako spojovací uzel mezi jednotlivými přístupovými přepínači a jako bod připojení k Internetu, proto funguje ve všech virtuálních sítích (pro učebny i server) jako brána. Všechny zde uvedené konfigurace směrovače Cisco 3550, včetně jejich popisů, jsou k dispozici v manuálu k tomuto zařízení [4].

[6] VLAN č. 50 je administrativní virtuální síť

4.2.1 Zabezpečení přístupu

Zabezpečení přístupu je stejné jako u přístupových přepínačů, BPDU guard ani DHCP Snooping zde nemá smysl, to je ošetřeno již na přístupové vrstvě.

4.2.2 Směrování

Směrovač zajišťuje především směrování paketů mezi jednotlivými sítěmi a zajišťuje připojení sítí k Internetu. Zde je ukázková konfigurace směrování, kdy je vidět adresace jednotlivých virtuálních sítí a nastavení brány pro přístup k Internetu:

```
interface Vlan50
 description Internet
 ip address 10.5.0.3 255.255.255.0
!
interface Vlan780
 description Server
 ip address 10.254.254.1 255.255.255.192
!
interface Vlan781
 description Ucebna 1
 ip address 10.254.254.65 255.255.255.192
!
interface Vlan782
 description Ucebna 2
 ip address 10.254.254.129 255.255.255.192
!
interface Vlan783
 description Ucebna 3
 ip address 10.254.254.193 255.255.255.192
!
ip default-gateway 10.5.0.1
```

Jak je vidět, každá učebna má přidělený rozsah o velikosti 64 IP adres, což umožňuje připojit až 61 koncových zařízení (první adresa je číslo sítě, druhá adresa je IP na virtuálním rozhraní směrovače - IP adresa brány pro připojené počítače, poslední adresa je broadcast).

4.2.3 DHCP Server

Na tomto směrovači je nutné nastavit také správný chod DHCP serveru [4]. V této ukázce je vidět nastavení pro 3 učebny. První 4 adresy z IP rozsahu server nebude přidělovat (mohou sloužit k administrativním účelům nebo jako rezerva). Jednotlivé rozsahy mají nastavenu síť, bránu, IP adresu TFTP serveru, název hlavního souboru pro PXE na TFTP serveru a dobu platnosti IP adresy. [7].

[7] hodnoty *0 0 30* znamenají 0 dní, 0 hodin a 30 minut

```
ip dhcp excluded-address 10.254.254.64 10.254.254.68
ip dhcp excluded-address 10.254.254.128 10.254.254.132
ip dhcp excluded-address 10.254.254.192 10.254.254.196
ip dhcp pool workstations-room-305
 network 10.254.254.64 255.255.255.192
 default-router 10.254.254.65
 bootfile /pxelinux.0
 next-server 10.254.254.2
 lease 0 0 30
!
ip dhcp pool workstations-room-306
 network 10.254.254.128 255.255.255.192
 default-router 10.254.254.65
 bootfile /pxelinux.0
 next-server 10.254.254.2
 lease 0 0 30
!
ip dhcp pool workstations-room-307
 network 10.254.254.192 255.255.255.192
 default-router 10.254.254.65
 bootfile /pxelinux.0
 next-server 10.254.254.2
 lease 0 0 30
!
```

5 Konfigurace PXE

Celá konfigurace PXE je uložena v jednom souboru, který je pro pracovní stanici dostupný protokolem *TFTP*. Konfigurační soubor použitý pro pracovní stanice v učebnách FAI UTB je součástí přílohy této práce, důležité parametry jsou popsány v této kapitole.

5.1 Zabezpečení a timeouty

Základní zabezpečení PXE spočívá v tom, že uživatel může pouze vybírat z těch položek, které mu dá správce TFTP serveru k dispozici, tedy že si nemůže přidat vlastní možnost a ani nesmí upravovat parametry nabízených položek. Toho se docílí několika parametry [5]:

- **noescape 1** - PXE nebude reagovat na tzv. *escape* klávesy, pomocí kterých by mohl uživatel úplně vyskočit z menu
- **prompt 0** - neumožní zadat uživateli vlastní možnost jako alternativu k definovaným položkám v menu
- **allowoptions 0** - znemožní uživateli měnit parametry jednotlivých položek v menu

Timeouty jsou definovány, aby uživatel v daném časovém úseku musel provést nějakou akci, pokud žádnou neprovede, spustí se v menu první položka:

- **timeout 40** - uživatel musí do 4 sekund od zobrazení menu stisknout jakoukoliv klávesu
- **totaltimeout 300** - uživatel má do 30 sekund na to, aby si pročetl položky v menu (eventuelně nápovědy k nim) a některou z nich si vybral

5.2 Nastavení menu

PXE umožňuje jak obyčejné textové rozhraní pro výběr položek v menu, tak grafické rozhraní. Grafické rozhraní umožňuje práci s více barvami, posun menu tak, aby *zapadlo* do obrázku na pozadí a vůbec působí na uživatele více přátelsky. Z toho důvodu bude použito grafické menu [6]:

- **default vesamenu.c32** - PXE si pomocí *TFTP* stáhne tento soubor, který grafické rozhraní umožňuje
- **menu background bg.jpg** - PXE si opět pomocí *TFTP* stáhne soubor s obrázkem (musí mít rozměr 640x480 pixelů, formát může být JPG nebo PNG)

Další konfigurace v zhledu menu, upravují pozici a rozměry menu a nápověd, jsou vidět přímo v souboru na DVD přiloženým k této práci, v souboru tvoří druhý odstavec a jsou uvozeny klíčovým slovem **menu**.

5.3 Položky v menu

Jednotlivé položky v menu jsou uvozeny klíčovým slovem **label** a unikátním identifikátorem položky. Až do další definice **label** se všechna nastavení týkají této položky. Každá položka má navíc tyto parametry [5]:

- **menu label** - slovní název položky, který se zobrazí v menu
- **text help, endtext** - všechny řádky mezi těmito dvěma parametry budou v menu zobrazeny jako nápověda nebo popis k dané položce
- **kernel** - jádro operačního systému, které se má stáhnout pomocí *TFTP* a spustit, v tomto projektu se jedná vždy jen o zkompilované linuxové jádro nebo o soubor `memdisk`, který je součástí balíku `pxelinux` a který stáhne pomocí *TFTP* celý obraz operačního systému, který se má spustit, nahraje ho do paměti a pokusí se ho nabootovat. Jako ukázka je na TFTP serveru uložen obraz starého systému MS-DOS 6.2, který je dnes volně přístupný na Internetu a který se dříve bootoval z diskety. Operační systém je velmi jednoduchý, kromě základních příkazů na práci s adresáři a soubory obsahuje pouze podporu pro CD-ROM.
- **append** - parametry pro jádro operačního systému, v případě linuxového jádra jde přímo o parametry pro jádro, v případě použití `memdisku` obsahuje název souboru přístupného pomocí *TFTP*, který bude nahrán do paměti a nabootován
- **localboot 0** - tento příkaz se použije v případě, že se nebootuje po síti, ale z lokálního disku (číslo udává, který disk se má použít), v tomto případě se parametry `kernel` ani `memdisk` nepoužijí

5.4 Parametry pro linuxové jádro

Linuxovému jádru mohou být předány jakékoliv parametry, které potřebuje, nicméně pro konkrétní nabootování po síti jsou nutné tyto [7]:

- **root=/dev/nfs** - celý kořenový souborový systém bude připojen protokolem *NFS*
- **ip=dhcp** - jádro si musí zjistit IP adresu síťového rozhraní a další nastavení sítě protokolem *DHCP*
- **nfsroot=IP:/path** - cesta k souborovému systému, který je přístupný protokolem *NFS*, `IP` je IP adresa serveru a `/path` je celá cesta k souborovému systému na centrálním serveru, celé přesné nastavení tedy může vypadat např. takto:

```
nfsroot=10.254.254.2:/remote-linux/disks/debian-4.0
```

6 Centrální linuxový server

Operační systém na centrálním serveru je Linux, je lhostejné, o jakou se jedná distribuci, nutné ovšem je, aby server měl nainstalovaný všechen potřebný software:

- TFTP server
- NFS server
- podpora dostatečně velkého ramdisku
- kompilace linuxového jádra

Všechna data potřebná pro běh pracovních stanic jsou umístěna na samostatném diskovém oddílu, který je připojen do `/mnt/sda7`. Zde jsou tyto adresáře:

- **kernel** - obsahuje rozbalená linuxová jádra (např. `kernel/linux-2.6.25`) i již zkompilovaná jádra (např. `kernel/bzImage-2.6.25`)
- **disks** - obsahuje souborové systémy pro pracovní stanice (např. `disks/debian-4.0`)
- **tftp** - obsahuje potřebné soubory pro TFTP

6.1 Ramdisk

Pro zvýšení rychlosti odezvy serveru budou veškerá data potřebná pro pracovní stanice umístěna na ramdisku, tedy v části paměti. Aby bylo něco takového na serveru možné, je nutné mít v jádře podporu pro minimálně jeden dostatečně velký ramdisk. Po startu serveru je ovšem nutné zajistit jeho formátování, připojení do adresáře `/remote-linux` a vytvoření adresářové struktury se soubory. Skript, který toto všechno zajistí, je k vidění na DVD přiloženém k této práci.

6.2 TFTP server

TFTP server je možné spouštět pomocí obecného síťového softwaru pro spouštění služeb `xinetd` nebo jako tzv. *standalone* službu. Pro zvýšení bezpečnosti se bude spouštět *tftpd* s právy uživatele *nobody*, poskytované soubory budou uloženy v adresáři `/remote-linux/tftp`.

6.2.1 Spouštění přes xinetd

V případě, že *tftpd* má být spouštěno pomocí xinetd, stačí upravit soubor `/etc/xinetd.d/tftp`, a to takto:

```
service tftp
{
  socket_type  = dgram
  protocol     = udp
  wait         = yes
  user         = root
  server       = /usr/sbin/in.tftpd
  server_args  = -u nobody -s /remote-linux/tftp
  disable      = no
}
```

Nyní již stačí zajistit spouštění xinetd na serveru standardní cestou.

6.2.2 Spouštění jako standalone server

Při absenci xinetd nebo z jakéhokoliv jiného důvodu je možné spouštět *tftpd* i jako samostatně běžící server. Na to je třeba vytvořit skript, který by pouštěl službu automaticky při startu serveru (bohužel nebývá součástí standardní instalace tftpd). Tento skript se může nacházet mezi ostatními standardním serverovými skripty v `/etc/rc.d`, popř. jinde, záleží na konkrétní distribuci. Ukázkový skript je uložen na DVD, které je přiloženo k této práci.

6.2.3 Parametry pro tftpd

V obou uvedených případech spouštění jsou zadané následující parametry [8]:

- **-u nobody** - zajistí, že server poběží s právy uživatele *nobody* a nikoliv uživatele *root*, který server spouští
- **-s /remote-linux/tftp** - adresář, ve kterém se nachází soubory a adresáře přístupné pomocí *TFTP*

V případě spouštění tftpd jako standalone serveru k těmto dvěma parametrům ještě přibude **-l**, což je instrukce pro tftpd, aby právě běžel ve standalone módu a nikoliv jako jediný proces, který po obsloužení příchozí konexe bude samočinně ukončen.

6.2.4 Obsah adresáře pro TFTP

Obsahem adresáře `/remote-linux/tftp` je všechno, co potřebuje pracovní stanice pro uskutečnění bootování pomocí PXE:

- **bg.jpg** - soubor v rozlišení 800x600 bodů, použije se na pozadí při výběru operačního systému
- **images/bzImage-2.6.25** - zkompilované linuxové jádro
- **images/dosboot.img** - obraz operačního systému MS-DOS (slouží jako ukázka, jak lze nabootovat i jiný než linuxový operační systém)
- **memdisk** - soubor umožňující pomocí PXE stáhnout do pracovní stanice obraz jakéhokoliv systému a následně ho spustit
- **pxelinux.0** - hlavní ovládací soubor pro PXE
- **pxelinux.cfg/default** - nastavení menu pro PXE
- **vesamenu.c32** - podpůrný soubor pro PXE, umožňuje grafické pozadí při výběru operačního systému

Výrobě linuxového jádra je věnována samostatná kapitola, soubor `dosboot.img` je obraz stažený z Internetu. Obrázek na pozadí je jednoduchý grafický výtvor.

6.3 NFS server

6.3.1 Spouštění NFS serveru

NFS server lze spouštět standardní cestou, pouze je třeba zajistit, aby NFS server poslouchal na konkrétním portu [8], abychom mohli snadno kontrolovat příchozí pakety. Toho se dosáhne tím, že se ve startovacím skriptu pro NFS server zajistí, aby se proces `rpc.mountd` spouštěl s parametrem **-d** následovaným číslem portu. Pro tento konkrétní případ byl vybrán port **777**.

6.3.2 Nastavení adresářů přístupných přes NFS

Nastavení NFS serveru se provádí editací souboru `/etc/exports`. Nastavení je v tomto případě poměrně jednoduché:

```
/remote-linux/disks    10.254.254.0/255.255.255.0(ro)
```

Tím bude obsah adresáře `/remote-linux/disks` přístupný pouze pro čtení (díky parametru **ro**) pro jakékoliv zařízení na síti, jehož IP adresa je v rozmezí `10.254.254.0 - 10.254.254.255`. V tomto rozsahu jsou zahrnuty všechny sítě ve všech ukázkových učebnách.

[8] nezadáním parametru pro konkrétní port nastartuje NFS server na náhodném volném portu

6.4 Lokální firewall

Lokální firewall je realizován pomocí **iptables** a je koncipován tak, že všechny pakety jdoucí na server nebo ze serveru musí mít charakter povoleného provozu, jinak jsou zahozeny. Firewall je pouštěn po startu počítače pomocí skriptu umístěného v `/etc/rc.d` nebo podobném (opět záležní na konkrétní distribuci).

Ukázkový startovací skript je možné nalézt na DVD přiloženým k této práci, stejně jako soubor s pravidly. Pravidla je ovšem třeba trochu popsat:

- povoleny jsou pouze takové přijímané pakety, jejichž cílová IP adresa je adresou serveru, zároveň jsou povoleny pouze takové odchozí pakety, jejichž zdrojová IP adresa je IP adresa serveru
- povolen je provoz na loopbacku
- je zakázáno přijímat fragmenty paketů, navíc každá nová příchozí TCP konexe musí začínat korektním paketem typu `SYN`
- je povolen provoz protokolem ICMP, ovšem pouze takový provoz, který vzniknul na serveru
- směrem ze serveru jsou povoleny pouze TCP konexe na cílové porty 20 a 21 (ftp), 22 (ssh), 25 (smtp), 80 (http) a 443 (https)
- směrem ze serveru jsou povoleny pouze UDP konexe na cílové porty 53 (dns) a 123 (ntp)
- přístup na služby LDAP, TFTP, NFS a SSH je povolen pouze z definovaných IP adres[9]

6.5 Kompilace linuxového jádra pro pracovní stanice

6.5.1 Nutné vlastnosti jádra

Na linuxové jádro není kladeno příliš nároků, aby však pracovní stanice fungovala tak, jak má, musí být splněna tato kritéria:

- jádro by mělo být nemodulární (tedy nebude mít žádnou podporu pro moduly), tím se usnadní vazba jádra a jednotlivých distribucí, protože pak je možné dosáhnout stavu, kdy s jedním jádrem lze bez další manipulace s jednotlivými moduly fungovat na různých distribucích, tedy odpadá instalace modulů do všech připravených distribucí
- podpora pro ty síťové karty, které jsou v pracovních stanicích

[9] nutnost protokolu LDAP je diskutován v dalších kapitolách

- podpora pro ramdisk (některé programy vyžadují možnost zápisu na disk, což na pracovní bezdiskové stanici není možné) - postačí podpora 4 ramdisků o velikosti 32 MiB
- podpora protokolu *NFS*
- možnost mít kořenový souborový systém připojený pomocí *NFS*
- žádná podpora pro pevné IDE nebo SATA disky (tím je ochráněn lokální disk v jednotlivých stanicích proti chtěnému nebo nechtěnému čtení a zápisu, nainstalovaný systém Microsoft Windows během práce v OS Linux bude nedotknutelný)

Jádra jsou rozbalena na serveru v adresáři `/mnt/sda7/kernel`, tedy např. v adresáři `/mnt/sda7/kernel/linux-2.6.25`. Konfigurace je standardně v souboru `.config`. Jádro lze jednoduše konfigurovat pomocí sekvence příkazů:

- **cd /mnt/sda7/kernel/linux-2.6.25**
- **make menuconfig** - spustí konfigurační program, který využívá hiearchický systém konfigurace
- **make bzImage** - zkompiluje výsledný obraz jádra
- **cp arch/x86/boot/bzImage ../bzImage-2.6.25** - zkopíruje výsledný obraz jádra, aby nebyl přepsán eventuelní rekompilací a aby byl dostupný pro kopírování na ramdisk do adresáře pro TFTP

Protože v budově FAI UTB je několik počítačových učeben, je zvolené jádro zkompilováno pokud možno co nejvíce obecně - podpora některých konkrétních věcí ustupuje, naopak jádro obsahuje podporu pro obecné drivery. Výjimku tvoří samozřejmě podpora síťových karet. Souboru `.config` je uložen na DVD přiloženým k této práci.

6.5.2 Důležité parametry linuxového jádra

Při výběru konfigurace jádra byly zvoleny následující parametry (vybrány jsou zde jen důležité):

- General setup / Support for pagging of anonymous memory - vypnuto (žádná podpora pro swap; u bezdiskových stanic nemá smysl)
- Enable loadable module support - vypnuto (jádro nebude nijak pracovat s moduly)
- Processor type and features
  - Processor family - zvoleno Pentium-III
  - Generic x86 support - vybráno

  - High Memory Support - zvoleno 4GB (umožňuje využití RAM v pracovní stanici až do 4 GiB)

- Executable file formats - zvolena pouze podpora pro ELF binaries
- Networking / Network options
  - IP: Kernel level autoconfiguration - zapnuto (umožňuje používání protokolů DHCP apod. během bootování)
  - IP: DHCP support - zapnuto
  - The IPv6 protocol - vypnuto (IPv6 se v sítí FAI UTB nepoužívá)
- Device drivers
  - Block devices / RAM block device support - zapnuto (podpora pro ramdisk)
  - Block devices / Default number of RAM disks - nastaveno 4
  - Block devices / Default RAM disk size - nastaveno 32768
  - ATA/ATAPI/MFM/RLL support / Include IDE/ATA-2 DISK support - vypnuto (systém vůbec nebude pracovat s pevnými IDE disky)
  - ATA/ATAPI/MFM/RLL support / Include IDE/ATAPI CDROM support - zapnuto (naopak práce s CD-ROM by měla být umožněna)
  - SCSI device support / SCSI disk support - zapnuto (umožňuje práci s SCSI disky - takové disky v pracovních stanicích sice nejsou, ale předpokládá se práce s USB flash disky, s těmi se pracuje díky emulaci SCSI)
  - SCSI device support / SCSI low-level drivers - vypnuto (podpora pro fyzické SCSI řadiče není potřeba)
  - Serial ATA and Parallel ATA drivers - vypnuto (podpora SATA disků není u bezdiskové stanice třeba)
  - Network device support / Ethernet (10 or 100 Mbit) - zde byla vybrána podpora pro tyto síťové karty: 3c590/3c900, EtherExpressPro/100, PCI NE2000, nForce, RealTek RTL-8129/8130/8139
  - Graphics support
    - /dev/agppart - zapnuta podpora těchto chipsetů: ATI, AMD, Intel, NVIDIA, VIA
    - Direct Rendering manager - zapnuto
  - Sound card support - vypnuto (na pracovních stanicích v učebnách se nebude žádné zvukové zařízení používat).
  - USB Support / USB Mass Storage Support - zapnuto (podpora pro USB disky, včetně obvyklých USB Flash pamětí)

- File systems
  - Second extended fs support - zapnuto
  - Ext3 journalling file system support - zapnuto
  - Kernel automounter support - zapnuto (usnadní uživatelům práci s připojením CD-ROM nebo USB disků)
  - CD-ROM/DVD FIlesystems - zapnuta podpora pro ISO 9660, Microsoft Joliet Extension a UDF
  - DOS/FAT/NT Filesystems - zapnuta podpora pro MSDOS a VFAT
  - Network File Systems - zapnuta podpora pro NFSv3, Root file system on NFS, SMB, CIFS a NCP (připojování svazků přes ssíť typu Novell)

7 Linuxová distribuce pro pracovní stanice

Připravená linuxová distribuce je jedním z hlavních úkolů této práce. Distribuce obsahuje kompletní adresářovou strukturu a soubory nutné pro standardní běh operačního systému, aby bylo možné s ním běžným způsobem pracovat.

K vytvoření distribuce vhodné pro běh v režimu na síti vedou dvě cesty. Distribuce se dá vytvářet přímo soubor po souboru, kdy na začátku je jen prázdný adresář, ve kterém se vytváří adresářová struktura a do ní se postupně nahrávají všechny potřebné soubory, nebo lze na jednu vybranou pracovní stanici nainstalovat OS Linux se všemi potřebnými programy a vybavením, celý nainstalovaný systém se pak zkopíruje na server a dále upraví tak, aby byl připraven na spouštění pomocí protokolu *NFS*.

Jako ukázková distribuce, která bude nainstalována na pracovní stanici a pak zkopírována na server, kde bude upravena, byla vybrána distribuce Debian verze 4.0 Etch. Její finální podobou by měl být plně využitelný pracovní desktop (tedy bude obsahovat webový prohlížeč, kancelářský software apod.). Tento postup není univerzální, při použití jiné distribuce nebo dokonce i jiné verze distribuce Debian může nastat situace, kdy bude potřeba provést další úpravy, znemožnit spouštění některých programů nebo naopak nové programy nainstalovat.

Distribuce, která bude vytvořena od základů jako ukázková minidistribuce, bude mít svůj základ v SuSE 10.3. Slouží zde pouze jako ukázka tvorby velmi malé distribuce, která bude mít pouze základní sadu nástrojů pro práci se soubory a adresáři a samozřejmě pro práci se sítí. Popsaný postup je téměř univerzální a lze ho uplatnit i na jiné distribuce nebo jiné verze.

7.1 Hlavní očekávané problémy

Největší problémy se spuštěním OS Linux po síti budou s programy, které vyžadují nějaký zápis do souborů, a to převážně do adresářové struktury `/var`, do adresáře pro dočasné soubory `/tmp` a do adresáře se soubory pro speciální zařízení `/dev`. Samozřejmě je možné, že různé softwary budou vyžadovat zápis ještě jinam, to se však bude muset řešit individuálně.

Zápis do adresářů `/var` a `/tmp` lze vyřešit poměrně snadno: vytvořením dvou ramdisků a jejich připojením na oba adresáře, navíc v `/var` vytvořit příslušnou adresářovou strukturu. Celý tento postup bude muset proběhnout před startem dalších programů.

Dalším očekávaným problémem bude připojování souborových systémů, protože každé připojení znamená zápis do souboru `/etc/mtab`, který je ale na pracovní stanici k dispozici pouze pro čtení a nikoliv pro zápis. Samotné připojování problém není, příkaz `mount` sice vypíše varování, ale souborový systém přesto připojí. Problém nastává až

v okamžiku spouštění programů, které z tohoto souboru čtou informace o připojených souborových systémech, protože tyto programy pak nemusí fungovat korektně, např. program pro zjištění volného místa na disku `df`. Stejné informace jako `/etc/mtab` ovšem dává soubor `/proc/mounts`, řešením tedy pravděpodobně bude neexistence souboru `/etc/mtab` a místo něho symbolický odkaz na `/proc/mounts`.

7.2 Instalace a konfigurace OS Debian 4.0

7.2.1 Instalace OS na pracovní stanici a jeho kopírování

Instalace OS na pracovní stanici probíhá naprosto stejně jako jakákoliv jiná instalace, výběr instalovaného softwaru je téměř libovolný. Jakmile je systém kompletně nainstalován a nakonfigurován, je systém připraven ke kopírování.

Kopírování běžícího systému ovšem s sebou nese rizika, některé programy mohou mít na disku otevřené soubory, nelze kopírovat existující sockety apod. Zkopírovaný systém by tedy mohl mít se spouštěním některých programů potíže. Systmém lze zkopírovat dvěma způsoby.

Buď lze zastavit všechny možné běžící programy, kromě těch, které jsou nezbytné k vlastnímu kopírování (zastavit lze např. `syslog`, `cron`, `sshd`, `acpid` atd.) a pak vytvořit jediný velký zkompriomvaný soubor:

```
workstation# cd /
workstation# tar -zc -f /debian-4.0.tar.gz bin boot dev etc
             home lib mnt opt root sbin tmp usr var
```

Příkaz `tar` vytvoří soubor, který bude obsahovat všechny nutné soubory a adresáře v distibuci, navíc zachová všchna přístupová práva nastavená na souborech a adresářích, stejně jako údaje o vlastníkovi a skupině. Adresáře jsou vyjmenovány, v seznamu chybí adresáře `/proc` a `/sys`, které obsahují dynamicky generované systémy a jejich přenos tedy nemá žádný smysl. Vzniklý soubor `/debian-4.0.tar.gz` se může pak nakopírovat přímo na server použitím jakéhokoliv síťového protokolu umožňující přenos souborů (*SSH*, *NFS*, *SMB*, *FTP* atd.) nebo na USB flash, CD nebo DVD atd. Soubor se pak na centrálním serveru rozbalí a distribuce se bude opravovat až na serveru:

```
server# cd /mnt/sda7/disks
server# mkdir debian-4.0
server# cd debian-4.0
server# tar -xz -f /debian-4.0.tar.gz
```

Druhý způsob znamená zastavit systém a rebootovat počítač do jiného operačního systému (lze použít např. live linuxovou distribuci spustitelnou z CD nebo se dá nabootovat do jiného operačního systému na disku). Disk nebo část disku obsahující distibuci Debian se pak připojí do nějakého adresáře a zabalí do jednoho souboru:

```
workstation# mkdir /mnt/debian
workstation# mount /dev/hda4 /mnt/debian
workstation# cd /mnt/debian
workstation# tar -zc -f /debian-4.0.tar.gz bin boot dev etc
             home lib mnt opt root sbin tmp usr var
```

Přenos a rozbalování souboru `/debian-4.0.tar.gz` pak probíhá stejným způsobem jako v prvním případě.

7.2.2 Úprava souborů /etc/fstab a /etc/mtab

Základní podmínkou pro úspěšné nabootování OS Linux přes síť je úprava souboru `/etc/fstab`. Ten obsahuje údaje o tom, které souborové systémy se budou připojovat při startu operačního systému, což při prvotní instalaci na disk bude určitě některá část na disku, např. `/dev/sda4`. Tento disk ovšem není na pracovní stanici k dispozici a bootovací proces by se v tomto okamžiku zastavil. Upravený soubor by měl vypadat takto:

```
proc       /proc            proc       defaults     0 0
sysfs      /sys             sysfs      defaults     0 0
usbfs      /proc/bus/usb    usbfs      defaults     0 0
```

Připojení kořenového souborového systému již proběhlo pomocí protokolu *NFS*, takže zbývá připojit virtuální souborové systému nutné pro chod serveru a dalších služeb.

V souboru `/etc/fstab` není definován žádný lokální disk ani jiné lokální zařízení. Oproti běžným systémům zde není ani *swap* [10], jeho použití u bezdiskových stanic nemá žádný smysl.

Soubor `/etc/mtab` by tedy měl být symbolický odkaz na `/proc/mounts`:

```
server# cd /mnt/sda7/disks/debian-4.0/etc
server# rm mtab
server# ln -s ../proc/mounts mtab
```

7.2.3 Obsah ramdisků a jejich připojování

Celkem budou použity 2 ramdisky: **/dev/ram0** bude sloužit pro zápis do `/tmp`, **/dev/ram1** bude sloužit programům, které vyžadují zápis do `/var`. Celý existující adresář `/var` ale zabírá příliš místa (např. `/var/lib` má více než 100 MiB). Ramdisk tedy bude připojen do adresáře `/var/ram` a pokud bude vyžadován zápis do některého z adresářů, bude se postupovat takto:

[10] vyhrazené místo na disku pro případ, že by došlo k zaplnění celé operační paměti a bylo by třeba její část dočasně uvolnit tak, že se uloží na disk a v případě potřeby se opět načte

- obsah všech vybraných adresářů se uloží do balíku `/var/ram.tar`
- tyto vybrané adresáře se odstraní
- místo každého takového adresáře se udělá symbolický odkaz do adresářové struktury ve `/var/ram`
- při startu systému se vyvtoří ramdisk a připojí se do `/var/ram`
- do adresáře `/var/ram` se rozbalí předpřipravený balík `/var/ram.tar`

Vybrané adresáře, které musí být z různých důvodů zapisovatelné, jsou: `/var/lock`, `/var/run`, `/var/tmp`, `/var/log`, `/var/lib/urandom`, `/var/lib/exim4`, `/var/lib/gdm` a `/var/db`.

Adresář `/var/log` nicméně obsahuje staré logové soubory ještě z instalace, které na pracovní stanici nemají žádný smysl, takže před vytvořením balíku `/var/ram.tar` se klidně mohou odstranit.

```
server# cd /mnt/sda7/disks/debian-4.0/var
server# mkdir ram
server# find log/ -type f -exec rm -f {} \;
server# tar -cf ram.tar lock run log tmp lib/urandom lib/exim4
server# for DIR in in db lock run log tmp; do
> rm -rf ${DIR}
> ln -s ram/${DIR} ${DIR}
> done
server# for DIR in lib/urandom lib/exim4 lib/gdm; do
> rm -rf ${DIR}
> ln -s ../ram/${DIR} ${DIR}
> done
```

Připojení ramdisků musí proběhnout před spouštěním dalších programů. Toho se docílí vytvořením speciálního startovacího skriptu, který se bude spouštět jako první. Jeho umístění je `/etc/init.d/preinit`, spouštění se zajistí těmito příkazy:

```
server# cd /mnt/sda7/disks/debian-4.0/etc/rcS.d/
server# ln -s ../init.d/preinit S00preinit
```

Práce programu `preinit`:

- vytvoří souborový systém na `/dev/ram0` a připojí ho do `/tmp`
- vytvoří adresářovou strukturu a nastaví správná přístupová práva do `/tmp`
- vytvoří souborový systém na `/dev/ram1` a připojí ho do `/var/ram`
- rozbalí balík `/var/ram.tar` do adresáře `/var/ram`

Program `preinit` bude provádět ještě další úkony, které jsou popsány v dalších kapitolách.

7.2.4 Drobné úpravy některých startovacích skriptů

Některé startovací skripty během startu počítače hlásí varování, která ale nejsou nijak kritická. Uživatele by to mohlo mást, proto je třeba upravit některé soubory:

- `/etc/init.d/discover` - komentovat řádku 177 (čte soubor `/proc/modules`, ale tento soubor neexistuje, protože podpora modulů není do linuxového jádra přidána)
- `/etc/init.d/mountall.sh` - skript se snaží připojit již připojené systémy (např. `/proc`), a proto vždy skončí s chybou, řešením je dodání příkazu `/bin/true` na řádku 27

7.2.5 Nastavení sítě

Nastavení sítě, které je v OS Linux Debian, bohužel kvůli své robustnosti příliš nevyhovuje jednoduchému požadavku, aby jediné síťové rozhraní bylo ovládáno pomocí programu `dhclient`, který zajišťuje konfiguraci pomocí protokolu *DHCP*. Z toho důvodu je veškeré nastavování sítě vypnuto a program `dhclient` je spouštěný z programu `preinit`. Nepotřebné programy se dají vypnout:

```
server# chroot /mnt/sda7/disks/debian-4.0/
bash# update-rc.d -f ifupdown remove
bash# update-rc.d -f networking remove
bash# update-rc.d -f hostname.sh remove
```

Ještě je třeba odstranit některé soubory z adresáře `/etc/init.d/dbus`, které se také snaží manipulovat se sítí. Soubory se mohou někam zazálohovat pro případ pozdějšího využití nebo úplně smazat:

```
server# cd /mnt/sda7/disks/debian-4.0/etc/dbus/event.d
server# rm 24dhcdbd 25NetworkManager 26NetworkManagerDispatcher
```

Nové úkony v programu `preinit` jsou:

- připojení souborového systému `/proc` (bez něj nemůže program `dhclient` vůbec fungovat)
- spuštění programu `dhclient` (nastavování sítě pomocí *DHCP*)
- zjištění přidělené IP adresy a jejího reverzního záznamu, z něho pak odvodit a nastavit *hostname* [11]

Nastavení programu `dhclient` [9] (soubor `/etc/dhcp3/dhclient.conf`):

```
send dhcp-lease-time 1800;
request subnet-mask, routers;
require subnet-mask, routers;
timeout 24;
select-timeout 5;
backoff-cutoff 16;
initial-interval 8;
script "/bin/true";
```

Dále je třeba změnit obsah konfiguračního souboru pro program `udev`, který se snaží přiřadit názvy síťových interfaců podle jejich MAC adres (údaje v souboru jsou z doby instalace). Ze souboru `/etc/udev/rules.d/z25_persistent-net.rules` je třeba odstranit každou řádku, která začíná nastavením proměnné **SUBSYSTEM**:

```
server# cd /mnt/sda7/disks/debian-4.0/etc/udev/rules.d
server# grep -v ^SUBSYSTEM= z25_persistent-net.rules > tmpfile
server# mv -f tmpfile z25_persistent-net.rules
```

Posledním krokem pro funkční nastavení sítě je korektní nastavení DNS serveru. Pro celou univerzitní síť se používá jeden DNS server, takže jeho IP adresa může být nastavena přímo v souboru `/etc/resolv.conf`:

```
server# echo "nameserver 195.178.88.66" >
        /mnt/sda7/disks/debian-4.0/etc/resolv.conf
```

7.2.6 Práce s CD/DVD mechanikou

CD/DVD mechaniky bohužel nejsou na všech pracovních stanicích FAI UTB stejné, dokonce nejsou ani všude stejně zapojené. Mechaniky typu `IDE` jsou na některých stanicích jako zařízení č. 1, jinde jako č. 2 nebo č. 3. Detekce těchto mechanik je tedy další úkol programu `preinit`, ten musí zjistit, jaké zařízení je typu CD/DVD.

[11] název pracovní stanice

Program `preinit` se nejprve pokusí najít zařízení typu `IDE`. Prochází jednotlivá zařízení v adresáři `/proc/ide` a pro každé z nich zkouší obsah souboru `media`. Je-li jeho obsahem klíčové slovo **cdrom**, jedná se o `IDE` CD/DVD mechaniku. Program tedy vytvoří symbolický odkaz `/var/ram/cd` ukazující přímo na speciální zařízení pro dané `IDE` zařízení v adresáři `/dev`, např. `/dev/hdd`.

Nenalezne-li program `preinit` CD/DVD mechaniku typu `IDE`, pokusí se ještě nalézt zařízení typu `SATA`. K tomu použije příkaz `lsscsi`, který mu dodá všechna dostupná `SCSI` zařízení [12] včetně jejich typu a speciálních souborů.

Není-li žádné takové zařízení nalezeno, symbolický odkaz `/var/ram/cd` nebude vytvořen. Pokud existuje více takových zařízení (na žádné současné pracovní stanici tomu tak není), vytvoří se symbolický odkaz pouze na první nalezené zařízení.

Se symbolickým odkazem `/var/ram/cd` se může pracovat stejně jako se speciálním souborem v adresáři `/dev`, např. připojování souborového systému na CD/DVD může vypadat takto:

```
workstation# mkdir /tmp/cdrom
workstation# mount -t auto /var/ram/cd /tmp/cdrom/
```

7.2.7 Připojování USB flash disků

Velmi populární zařízení na přenos dat mezi počítači jsou v dnešní době USB flash disky. Jedná se o malá zařízení, jejichž rozměry nepřesahují pár centimetrů, do počítače se připojují přes USB rozhraní a fungují tak jako přenosná datová úložiště.

USB flash disk je v systému přístupný díky emulaci v jádru jako běžný SCSI disk, na kterém je jen jeden oddíl [13]. Na tomto oddílu je obvykle souborový systém typu *FAT* (podpora pro tento souborový systém v jádře pro pracovní stanice je).

První připojený USB flash disk bude v systému přístupný jako první SCSI disk, tedy `/dev/sda`, s jediným připojitelným oddílem `/dev/sda1`. Tento oddíl lze snadno připojit do libovolného adresáře:

[12] zařízení typu `SATA` systému jsou přístupné přem emulaci `SCSI`

[13] pro ověření rozdělování těchto zařízení na jednotlivé oddíly bylo otestováno 10 různých USB flash disků nebo MP3 přehrávačů o různých velikostech a od různých výrobců, všechny byly rozděleny stejně

```
workstation# fdisk -l /dev/sda

Disk: /dev/sda: 262 MB, 262144000 bytes
16 heads, 32 sectors/track, 1000 cylinders
Units = cylinders of 512 * 512 = 262144 bytes

   Device Boot      Start         End      Blocks     Id   System
/dev/sda1   *           1         999      255728      6   FAT16

workstation# mkdir /tmp/flash
workstation# mount /dev/sda1 /tmp/flash
```

Další připojený USB flash disk bude mít data uložená na `/dev/sdb1`, třetí připojený disk na `/dev/sdc1` atd.

7.2.8 Používání autofs pro připojování USB disků a CD/DVD

Uvedený postup pro připojování souborových systémů může být pro uživatele příliš náročný a navíc pro spouštění příkazů `mount` (i `umount`) v této podobě vyžaduje přístupová práva uživatele *root*, který ale není běžným uživatelům k dispozici.

Snadné řešení je použití `autofs`, systému, který je součástí linuxového jádra (podpora tohoto systému byla do jádra pro pracovní stanice vložena) a který umí připojit souborový systém v okamžiku, kdy se uživatel nebo spuštěný program pokusí číst nebo zapsat do předem určeného adresáře. Program `autofs` se spouští automaticky při startu systému a jeho hlavní nastavení je v souboru `/etc/auto.master`:

```
/mnt    /etc/auto.mnt   --timeout=60
/flash  /etc/auto.flash --timeout=5
```

Toto nastavení znamená, že program `autofs` bude sledovat přístupy do zatím prázdného adresáře `/mnt` a do adresáře `/flash`, ten je třeba ovšem vytvořit:

```
server# mkdir /mnt/sda7/disks/debian-4.0/flash
```

Jednotlivé podadresáře uvnitř těchto dvou adresářů nejsou nijak vidět, dokud k nim není přistoupeno. Parametr `--timeout` udává počet sekund, kdy je zařízení samočinně odpojeno, pokud k němu nikdo nepřistupuje. U USB flash systémů, kde jsou uživatelé většinou zvyklí odpojovat zařízení kdykoliv, je nastaveno málo sekund, u CD/DVD mechaniky je tento čas delší. Nastavení souboru `/etc/auto.mnt`:

```
cd      -fstype=auto,users,ro    :/var/ram/cd
```

Tím je definován adresář `/mnt/cd`, který zatím neexistuje. Pokud ovšem uživatel provede přístup k tomuto adresáři (např. příkazem `$cd /mnt/cd`), program `autofs` se pokusí připojit zařízení `/dev/cd` do adresáře `/mnt/cd` a v případě úspěchu se příkaz `cd` úspěšně provede.

Parametry jsou na jediné řádce souboru `/etc/auto.mnt` zapsány v tomto pořadí: podadresář, parametry pro příkaz `mount` a zařízení, které se má připojit. V tomto případě jsou parametry pro příkaz `mount` [12] tyto:

- **fstype=auto** - automatické rozpoznávání typu souborového systému
- **user** - kterýkoliv uživatel může zařízení připojit (nemusí mít tedy přístupová prává uživatele *root*)
- **ro** - systém bude připojen v režimu pro *read-only*

Nastavení souboru `/etc/auto.flash` (předpokládá se, že v jedné pracovní stanici budou zapojeny nejvýše 4 USB flash disky současně, pokud by se ukázalo, že tento počet je malý, stačí přidat další řádky):

```
1  -fstype=auto,users,fmask=111,dmask=000,sync  :/dev/sda1
2  -fstype=auto,users,fmask=111,dmask=000,sync  :/dev/sdb1
3  -fstype=auto,users,fmask=111,dmask=000,sync  :/dev/sdc1
4  -fstype=auto,users,fmask=111,dmask=000,sync  :/dev/sdd1
```

Přístup na jednotlivé připojené USB flash disky tedy uživatel získá přístupem do adresářů `/flash/1`, `/flash/2` atd. Oproti parametrům pro příkaz `mount` ze souboru `/etc/auto.mnt` přibyly další dva [12]:

- **fmask=111** - nastavuje přístupová práva na soubory v připojeném souborovém systému, hodnota **111** znamená, že u každého souboru je povolen zápis a čtení, jak pro vlastníka, tak pro skupinu a ostatní uživatele, vlastníkem sice bude vždy uživatel *root*, stejně jako skupina, ale tímto parametrem je zajištěno, že uživatelé mohou s daty na flash disku libovolně nakládat
- **dmask=000** - nastavuje přístupová práva na adresáře v připojeném souborovém systému, hodnota **000** znamená, že vlastník, skupina i ostatní uživatelé mají pro každý adresář všechna práva
- **sync** - všechna čtení i zápisy na USB flash disk budou vykonávána okamžitě

7.2.9 Konfigurace X Window

Systém **X Window** je grafické prostředí v OS Linux. Jeho univerzální konfigurace zřejmě neexistuje, je příliš závislá na používaném hardwaru, od vstupních zařízení (myši, klávesnice) až po výstupní (grafická karta, monitor). Výroba nějaké konfigurace, která by byla funkční v každé pracovní stanici ve všech počítačových učebnách není možná, protože v různých učebnách jsou různé druhy hardwaru.

Linuxové distribuce mají obvykle nějaký nástroj na detekci hardwaru a následné vytváření konfiguračního souboru `/etc/X11/xorg.conf`, v distribuci Debian se tento nástroj spouští tímto příkazem:

```
# dpkg-reconfigure -f passthrough xserver-xorg
```

Parametr **-f passthough** zajistí, že program se nebude ptát uživatele na žádnou otázku (bez udání tohoto parametru si nechává všechny odhadnuté hodnoty potvrdit) a předpokládá, že uživatel odpovídá na každou otázkou souhlasem. Výsledkem tohoto programu je nový soubor `/etc/X11/xorg.conf`.

Adresář `/etc/X11` je ale přístupný pouze pro čtení, takže podobně jako v případě přípravy adresářové struktury `/var` se nyní obsah `/etc/X11` zabalí, odstraní a na jeho místě se vytvoří symbolický odkaz do `/var/ram/X11`:

```
server# cd /mnt/sda7/disks/debian-4.0/etc
server# tar cf X11.tar X11
server# rm -rf X11
server# ln -s ../../var/ram/X11 X11
```

Do programu `preinit` je ještě nutné dodat další úkon: po připojení ramdisku do `/var/ram` zde rozbalit obsah souboru `/etc/X11.tar` a spustit příkaz `dpkg-reconfigure`.

Vytvořený soubor `xorg.conf` ovšem obsahuje pouze minimální konfiguraci, tedy základní nastavení klávesnice (bez podpory češtiny) a rozlišení 800x600. To je třeba ještě upravit, což zajistí opět program `preinit`. Do sekce **Monitor** se musí přidat následující řádky:

```
HorizSync 28-69
VertRefresh  43-75
```

Tyto hodnoty by měly plně postačovat pro běžné rozlišení, které je v současné době na pracovních stanicích 1280x1024 nebo u starších monitorů 1024x768, u stanic s širokoúhlým LCD monitorem pak 1200x800. Tato rozlišení je nutné ještě přidat do souboru `xorg.conf` do sekce **Screen** a všech jejich subsekcí **Display**:

```
Modes "1280x1024" "1200x800" "1024x768"
```

Nastavení české klávesnice pak vyžaduje úpravy v sekci **Keyboard**. Uživatel bude mít na výběr mezi dvěma rozloženími kláves: standardní americké rozložení a české rozložení *qwertz*. Mezi jednotlivými rozloženími se dá přepínat současným stiskem kláves `ALT` a `SHIFT`. Standardní rozložení je české, v případě přepnutí do amerického rozložení se navíc rozsvítí na klávesnici kontrolka `ScrollLock`.

```
Option   "XkbModel"    "pc104"
Option   "XkbLayout"   "cz,us"
Option   "XkbVariant"  "qwertz"
Option   "XkbOptions"  "grp:alt_ahift_toggle,grp_led:scroll"
```

7.2.10 Ověřování uživatelů protokolem LDAP

Pracovní stanice musí být schopná ověřovat uživatele jak z lokálních zdrojů (soubor `/etc/passwd`), tak protokolem *LDAP*. K tomu je třeba modifikovat systémové soubory, které autentizaci uživatelů zajišťují [14].

Soubor `/etc/nsswitch.conf` musí kromě jiných obsahovat i tyto řádky:

```
passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap
```

Soubor /etc/pam.d/common-auth

```
auth      sufficient      pam_ldap.so
auth      required        pam_unix.so nullok_secure use_first_pass
```

Soubor /etc/pam.d/common-account

```
account   required        pam_unix.so
account   sufficient      pam_ldap.so
account   required        pam_unix.so try_first_pass
```

Soubor /etc/pam.d/common-password

```
password  sufficient      pam_ldap.so
password  required        pam_unix.so nullok obscure min=4 max=8 md5
```

Ověřování uživatelů proti univerzitnímu serveru `ldap.utb.cz` bohužel není možné. Datová struktura uživatele pro úspěšné přihlášení do OS Linux musí mít několik povinných položek [14], univerzitní LDAP server ovšem neposkytuje ani jednu:

- `objectClass: account`
- `objectClass: posixAccount`
- `objectClass: shadowAccount`
- `loginShell` (obvyklá hodnota */bin/bash*)
- `uidNumber` (identifikační číslo uživatele, např. *1012*)
- `gidNumber` (identifikační číslo skupiny, např. *1000*)
- `homeDirectory` (cesta k domovskému adresáři)
- `userPassword` (zašifrované heslo uživatele)

V současné době se uvažuje o reinstalaci univerzitního LDAP serveru a o přechodu na jinou než současnou platformu (nyní se používá NDS od firmy Novell). Do té doby server není možné používat.

Prozatím bude tedy spuštěn vlastní LDAP server (na centrálním serveru), který bude mít vlastní databázi vlastních uživatelů. OS Debian tedy bude nakonfigurován tak, aby používal centrální linuxový server jako LDAP server pro autorizaci uživatelů. Opět je tedy nutné změnit obsah některých souborů [14]:

Soubor /etc/ldap/ldap.conf

```
host  10.254.254.2
base  ou=fake,o=utb
```

Soubor /etc/pam_ldap.conf

```
host 10.254.254.2
base ou=fake,o=utb
ldap_version 3
ssl no
pam_password crypt
binddn cn=passwd,ou=fake,o=utb
bindpw qwerty
```

Soubor /etc/libnss-ldap.conf

```
host 10.254.254.2
base ou=fake,o=utb
ldap_version 3
ssl no
pam_password crypt
binddn cn=passwd,ou=fake,o=utb
bindpw qwerty
```

7.2.11 Spuštění a nastavení vlastního LDAP serveru

Na centrálním linuxovém serveru je nainstalován *openLDAP* s vlastní databází, protože získávání autorizačních údajů z univerzitního LDAP serveru ani nějaká synchronizace dat ve větším měřítku není možná. Pro co nejlepší simulaci datové struktury univerzitních LDAP údajů byl zvolen základní kontext `ou=fake,o=utb` (na univerzitním LDAP serveru jsou hodnoty např. `ou=fai-st,o=utb` pro studenty FAI nebo `ou=fame,o=utb` pro zaměstnance FAME).

[14] údaje uvnitř těchto souborů jsou popsány v další kapitole

Hlavní konfigurační soubor pro openLDAP je `/etc/ldap/slapd.conf`. Kromě standardních nastavení je třeba definovat tyto hodnoty [14]:

```
suffix  "ou=fake,o=utb"
rootdn  "cn=root,ou=fake,o=utb"
rootpw  qwerty
index   objectClass     eq
index   uid     eq
```

Definice `suffix` je hlavní kontext, který se bude při autorizaci používat. Řádky `rootdn` a `rootpw` ukazují administrátorský přístup, Řádky `index` definují, podle jakých parametrů je pak možné vyhledávat záznamy v LDAP databázi. Kromě tohoto nastavení je ještě třeba definovat přístupová práva:

```
access to * attrs=userPassword
  by dn="passwd,ou=fake,o=utb" read
  by anonymous auth
  by self read
  by * none
acces to *
  by * read
```

Databáze uživatelů je tedy přístupná všem pro čtení, pouze atribut `userPassword` je přístupný pro uživatele `passwd`. Tento uživatel může číst tento atribut u všech ostatních uživatelů, jinak každý uživatel může číst tento atribut pouze u svého vlastního záznamu, u jiných záznamů nebude zobrazen.

Nezbytným krokem ke správnému chodu LDAP serveru je definice obsahu databáze. Nejprve se definuje kořenový objekt a uživatel `passwd` [15]:

```
dn: ou=fake,o=utb
objectclass: top
objectclass: organizationalUnit
ou: fake

dn: cn=passwd,ou=fake,o=utb
cn: passwd
sn: Passwd
objectclass: top
objectclass: person
objectclass: posixAccount
uid: passwd
```

[15] hodnota userpassword se získá příkazem slappasswd, v tomto případě je použito heslo qwerty

```
userpassword: {SSHA}VJEnCQY9S83bv+pHG5vXXLwbjh2dbjeM
uidnumber: 99
gidnumber: 99
gecos: Passwd reader
loginShell: /bin/false
homeDirectory: /var/lib/nobody
```

Následuje definice uživatelů. Zde je definice jednoho uživatele, definice dalších uživatelů je podobná. Jedním z nejdůležitějších atributů je `uidNumber`, identifikační číslo uživatele, které musí být v rámci celé databáze unikátní:

```
dn: cn=john,ou=fake,o=utb
cn: john
sn: Doe
objectClass: top
objectClass: person
objectClass: posixAccount
objectClass: shadowAccount
uid: john
userPassword:: {SSHA}i3PFTcdbC4GszfOaMqVb2CG9/wh2G2G8
uidNumber: 103
gidNumber: 100
gecos: John Doe
loginShell: /bin/bash
shadowLastChange: 10877
shadowMin: 0
shadowMax: 999999
shadowWarning: 7
shadowInactive: -1
shadowExpire: -1
shadowFlag: 0
homeDirectory: /var/ram/home/john
```

Za zmínku stojí také atribut `homeDirectory`, který ukazuje přímo do `/var/ram`. Domovské adresáře na pracovní stanici se vytváří až při přihlášení uživatele, proto musí být umístěny na ramdisku, kde je možné je vytvářet.

7.2.12 Domovské adresáře pro uživatele

Domovské adresáře pro všechny užvatele univerzitního LDAP serveru jsou k dispozici na školním novellovém serveru `nw-central`. Ten je možné připojit i z OS Linux programem `ncpmount` (podpora síťového souborového systému NCP je zabudovaná v jádře pro pracovní stanice). Tento adresář se musí připojit v okamžiku přihlášení uživatele do systému, navíc

s uživatelským heslem. To zajistí modul `pam_script` [13], jehož používání se zapíná v souboru `/etc/pam.d/gdm` [16], do kterého je nunté přidat tuto řádku

```
session     required     pam_script.so expose=1
```

Modul `pam_script` zajistí při přihlášení uživatele spuštění skriptu uloženého jako `/etc/security/onsessionopen`, parametr `expose=1` navíc skriptu předá v podobně environmentální proměnné `AUTHTOK` uživatelské heslo, které je nutné k autorizaci při připojování novellového svazku. Skript `/etc/security/onsessionopen` pak vypadá takto:

```
#!/bin/sh

SERVER="nw-central"
SERVER_IP="195.178.88.70"

LDAP_HOST=‘grep host /etc/ldap/ldap.conf | awk ’{print $2}’‘
LDAP_DN=‘ldapsearch -x -h ${LDAP_HOST} ’(uid=‘${USER}‘) dn | grep ^dn‘
LDAP_USER=""
LDAP_OU=""
for TOK in ‘echo ${LDAP_DN#* } | sed -e ’s/,/ /g’; do
  case ${TOK} in ou=*) LDAP_OU=${TOK#*=} ;; esac
  if [ -z "${LDAP_CONTEXT}" ]; then
    LDAP_USER=${TOK#*=}
  else
    LDAP_USER="${LDAP_USER}.${TOK#*=}"
  fi
done
if [ -z "${LDAP_USER}" -o -z "${LDAP_OU}" ]; then exit 1; fi

mkdir -p ${HOME} || exit 1
ncpmount -S ${SERVER} -C -U ${LDAP_USER} -u ‘id -u‘ -g ‘id -g‘ \
  -V data/${LDAP_OU}/${USER} -A ${SERVER_IP} -P ’${AUTHTOK}’ ${HOME}
```

Po odlogování je ještě třeba svazek odpojit, to zajistí skript `/etc/security/onsessionclose`:

```
#!/bin/sh
ncpumount ${HOME}
```

Poslední věc nutná pro funkčnost tohoto skriptu je existence adresáře `/var/ram/home` s takovými přístupovými právy, aby zde každý uživatel mohl vytvářet adresáře, to zajistí opět program `preinit`.

[16] používání gdm je vysvětleno v další kapitole

Celý tento princip je ovšem funkční pouze v souladu s autorizací oproti univerzitnímu LDAP serveru, kde je správně nastavený kontext uživatele.

7.2.13 Používání pracovní stanice z pohledu uživatele

Při výběru OS Debian během úvodního menu při startování počítače jsou nejprve vidět výstupy jádra, které detekuje hardware a poté spouští jednotlivé programy. Na konci této sekvence je spouštění programu *gdm*, tedy grafické konzole pro přihlášení uživatele. Zde si uživatel může vybrat jazyk grafického prostředí nebo jeho vzhled.

Po úspěšném ověření a přihlášení má uživatel k dispozici svůj domovský adresář na univerzitním novellovém serveru, webový prohlížeč, kancelářský software, nástroje pro práci na síti, prostě kvalitní linuxovou operační stanici. Může pracovat s CD/DVD mechanikou (včetně vypalování), může pracovat s USB flash diky.

Uživatel má stále jen práva obyčejného uživatele, pokud nezná heslo uživatele *root*. I kdyby přesto jakýmkoliv chtěným nebo nechtěným způsobem získal superuživatelská práva, v žádném případě nemůže dojít k poškození souborového systému nebo celé distribuce, protože všechny soubory jsou poskytovány centrálním serverem, navíc pouze v režimu pro čtení. Pracovní stanice s tímto systémem lze tedy považovat za naprosto bezpečné.

V případě jakéhokoliv problému s operačním systémem jako takovým (zatuhnutí jádra, nefunkční síť apod.) je možné pracovní stanici libovolným způsobem restartovat, není problém za běhu stisknout tlačítko `reset`, k žádnému poškození ničeho nedojde, uživatel pouze ztratí neuložená data, se kterými pracuje.

7.3 Tvorba samostatné linuxové minidistribuce

Malý operační systém se dá tvořit jednoduše kopírováním pouze vybraných potřebných souborů z již bežícího systému [10]. Ten může být nainstalován na pracovní stanici, na serveru, téměř kdekoliv. V tomto případě byl použit operační systém běžící mé vlastní na pracovní linuxové stanici, distribuce SuSE verze 10.3, nicméně tento návod je téměř univerzální a dá se použít na jakoukoliv jinou distribuci.

Cílem této minidistribuce je ukázat, jak se dá vytvořit opravdu malá a jednoduchá linuxová distribuce, která bude snadno rozšiřitelná o další programy. Pro jednoduchost je zde pouze uživatel *root*, není zde žádná těžká autorizace, běží jen nezbytně nutné programy a k dispozici je dostatek programů pro základní práci se sítí.

Jako ukázka programu, který běží na pozadí a vykonává práci, je nainstalován `syslogd`, který zajišťuje zapisování informací o systému nebo od běžících programů do souborů.

Tuto minidistribuci lze pak samozřejmě dál rozšiřovat postupnými instalacemi dalších programů a jejich nastavováním, dalším rozšířením mohou být např. programy `cron`, `postfix`, `named` atd.

7.3.1 Příprava adresářové struktury

Nejprve je třeba vytvořit vlastní adresářovou strukturu:

```
server# mkdir /mnt/sda7/disks/suse-10.3-mini
server# cd /mnt/sda7/disks/suse-10.3-mini
server# mkdir -p bin dev/pts etc lib mnt proc root sbin usr/lib var
server# ln -s var/tmp tmp
```

Oproti standardní distribuci zde chybí některé adresáře:

- `boot` - jádro zde není třeba, je k dispozici na TFTP serveru
- `home` - minidistribuce bude mít pouze jediného uživatele: *root*
- `opt` - nepředpokládá se instalace nějakých náročnách softwarů

Adresář `/tmp` je pouze symbolický odkaz na `/var/tmp`, protože do adresáře `/var` bude připojen ramdisk a bude tedy zapisovatelný.

7.3.2 Výběr souborů pro adresář /bin

Adresář `/bin` obsahuje binární soubory, které umožňují základní práci se systémem, tedy např. programy na kopírování souborů, vytváření adresářů apod. Pro základní ukázku a funkčnost systému byly vybrány tyto soubory (uvedeno včetně umístění na nainstalovaném systému):

```
/sbin/agetty
/sbin/arp
/bin/awk
/bin/basename
/bin/bash
/bin/cat
/bin/chmod
/bin/chown
/bin/cp
/usr/bin/cut
/bin/date
/bin/dd
/bin/df
/sbin/dhclient
/usr/bin/dirname
/usr/bin/du
/bin/false
/sbin/fdisk
/sbin/fsck
/sbin/fsck.ext2
/bin/fuser
/bin/grep
/bin/gzip
/sbin/halt
/usr/bin/head
/usr/bin/host
/bin/hostname
/usr/bin/id
/bin/ip
/usr/bin/killall
/sbin/killall5
/bin/ln
/bin/ls
/usr/bin/lsof
/usr/bin/md5sum
/bin/mkdir
/sbin/mkfs.ext2
/bin/more
/bin/mount
/usr/sbin/mtr
/bin/mv
/bin/netstat
/usr/bin/nslookup
/usr/sbin/ntpdate
/bin/ping
/bin/ps
/sbin/reboot
/bin/rm
/bin/rmdir
/usr/bin/scp
/bin/sed
/bin/sleep
/bin/sort
/bin/stty
/bin/sync
/sbin/syslogd
/usr/bin/tac
/usr/bin/tail
/bin/tar
/usr/sbin/tcpdump
/usr/bin/tee
/usr/bin/telnet
/usr/bin/top
/bin/touch
/usr/sbin/traceroute
/bin/true
/usr/bin/tty
/sbin/tune2fs
/bin/umount
/bin/uname
/usr/bin/uniq
/usr/bin/uptime
/bin/vi
/usr/bin/w
/usr/bin/who
/usr/bin/whoami
/usr/bin/wc
/usr/bin/wget
```

Uvedené soubory lze snadno kopírovat pomocí příkazu `cp`:

```
server# cp /bin/agetty /mnt/sda7/disks/suse-10.3-mini/bin/
```

Kromě těchto binárních souborů se ještě může hodit symbolický odkaz `/bin/sh`:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/bin/
server# ln -s bash sh
```

V seznamu souborů chybí `/bin/login`. Tento soubor je v distribuci pevně svázán s autorizačními moduly `pam`, které ovšem v minidistribuci nejsou potřeba. Místo něho bude mít distribuce vlastní malý soubor, který pouze načte heslo, zjistí jeho MD5 součet a porovná ho se záznamem v `/etc/passwd`. Tento soubor je k dispozici na DVD přiloženým k této práci.

7.3.3 Kopírování knihoven do /lib a /usr/lib

Ke spouštění binárních souborů je ovšem nutné mít knihovny, které tyto soubory využívají. Kromě těchto základních knihoven jsou třeba ještě podpůrné knihovny z balílku `glibc`:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/bin/
server# mkdir -p lib usr/lib
server# for FILE in `ldd * | grep "=>" | awk '{print $3}' | sort | uniq`; do
server>   objcopy -S ${FILE} ..${FILE}
server>   chmod 755 ..${FILE}
server> done
server# for FILE in ld-linux libnss_compat libnss_dns libnss_files; do
server> objcopy -S /lib/${FILE}.so.2 ../lib/${FILE}.so.2
server> chmod 755 ../lib/${FILE}.so.2
server> done
```

Uvedené příkazy zjistí seznam potřebných knihoven na běžícím operačním systému a nakopírují je do příslušných adresářů v minidistribuci. Uvedený seznam vyžadoval pouze existenci knihoven v adresářích `/lib` a `/usr/lib`, které již byly vytvořeny předem. Příkaz `objcopy` vytváří bohužel soubory bez přístupových práv pro spouštění (toto právo je ale nutné při otevírání knihovny binárním souborem), proto se musí toto přístupové právo nastavit příkazem `chmod`.

7.3.4 Kopírování speciálních souborů do /dev

Kopírování souborů je velmi snadné, opět se použije příkaz `cp`:

```
server# cd /dev/
server# cp -dpR console initctl kmem mem null ptmx ram[0123] random
        tty tty[01] urandom zero /mnt/sda7/disks/suse-10.3-mini/bin/
```

Kopírují se jen nejnutnější soubory nutné pro chod systému, nejsou zde tedy zmíněny (např. `hda1` umožňující práci s IDE zařízením nebo `ttyS0` pro práci se sériovým portem).

V této minidistribuci se bude také spouštět program `syslogd`, který ale vytváří speciální soubor `/dev/log`, což v tomto případě ale není možné. Proto se vytvoří pouze symbolický odkaz na soubor `/tmp/dev-log` a program `syslogd` se spustí s příslušným parametrem, aby nevytvářel speciální soubor `/dev/log`, ale aby místo něj vytvořil na ramdisku soubor `/tmp/dev-log`:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/dev/
server# ln -s ../tmp/dev-log log
```

7.3.5 Obsah adresáře /sbin

Adresář `/sbin` obsahuje pouze 3 soubory:

- **init** - základní soubor spouštěný jádrem
- **shutdown** - slouží ke korektnímu vypnutí počítače (i když vypnutí bezdiskové stanice tlačítkem reset nebo vypojením napájení nezpůsobuje vůbec žádné škody)
- **rc** - program spouštěný z programu `init`, stará se o připojování souborových systémů a spouštění základních programů

Programy `init` a `shutdown` se opět zkopírují:

```
server# cp /sbin/{init,shutdown} /mnt/sda7/disks/suse-10.3-mini/bin/
```

S programem `rc` je to složitější, pro účely minidistribuce je nutné ho napsat, aby udělal pouze nutnou základní práci:

- připojí souborové systémy `/proc` a `/dev/pts`
- vytvoří ramdisk a připojí ho do `/var`
- vytvoří adresářovou strukturu v `/var`
- spustí program `dhclient` (obsluhuje DHCP)
- nastaví *hostname* (údaj pro název souboru vezme z DNS)
- nastaví správný čas (název NTP serveru vezme z `/etc/ntpserver`)
- spustí program `syslogd`
- zapíše informace o startu do souboru `/var/log/boot.log`

Celý tento skript je uložen na DVD přiloženým k této práci.

7.3.6 Ramdisk na adresáři /var

Do adresáře `/var` vyžadují zápis některé programy, proto musí být na pracovní stanici vytvořen ramdisk (o to se postará `/sbin/rc`). Do tohoto adresáře vyžadují zápis:

- **syslogd** - zapisuje svůj PID [17]
- **dhclient** - zapisuje informace z DHCP odpovědí do `/var/lib/dhcpd/dhcpd.leases`
- **/sbin/rc** - zapisuje údaje o spuštění systému do `/var/log/boot.date`

Další výhodou je existence adresáře `/var/tmp`, na který se odkazuje přímo `/tmp` - tím tedy adresář `/tmp` funguje pro zápis pro kterýkoliv spuštěný program.

7.3.7 Obsah adresáře /etc

Adresář `/etc` uchovává data o nastavení téměř všeho, z toho důvodu může být jeho popis velmi rozsáhlý. Některé soubory stačí zkopírovat, jiné se musí upravit nebo vůbec vytvořit od začátku.

[17] Process identification - číslo běžícího procesu

Soubor inittab

Soubor `inittab` určuje, co má hlavní startovací soubor `init` v kterém režimu spustit. V jednoduché minidistribuci vypadá takto:

```
id:3:initdefault:
si::sysinit:/sbin/rc
ca::ctrlaltdel:/sbin/shutdown -r now
1:12345:respawn:/bin/agetty 38400 tty1
```

V prvním řádku se definuje tzv. *runlevel*, tedy jakýsí mód běhu (v různých módech mohou být spuštěny různé programy nebo služby), v tomto případě je hodnota **3**. Druhý řádek definuje soubor, který se má spustit při startu systému (tedy `/sbin/rc`). Třetí řádek definuje, co se má dít při stisku kláves CTRL-ALT-DEL (program `/sbin/shutdown`) a poslední řádek zajišťuje spouštění jedné konzole, kde se uživatel může přihlásit do systému.

Soubory pro přihlášení uživatele

Minidistribuce bude mít jediného uživatele, a to *root*, stejně tak bude mít pouze jednu uživatelskou skupinu stejného jména. Protože tato minidistribuce má sloužit především k demonstrativním a výukovým účelům, bude zde jednoduchá autorizace: heslo se snadno zašifruje pomocí příkazu `md5sum` a uloží do souboru `/etc/passwd` (pro ukázku bylo zvoleno heslo `qwerty`):

```
server# cd /mnt/sda7/disks/suse-10.3-mini/etc
server# PASSWD=‘echo qwerty | md5sum | cut -d \  -f 1‘
server# echo "root:${PASSWD}:0:0:root:/root:/bin/bash" > passwd
server# echo "root:x:0:root" > group
```

Soubory pro práci na síti

Některé soubory nutné pro práci se sítí lze zkopírovat:

```
server# cd /etc
server# cp host.conf nsswitch.conf protocols services
        /mnt/sda7/disks/suse-10.3-mini/bin/
```

Další soubory je nutné vytvořit:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/etc
server# echo -e "localhost\t127.0.0.1" > hosts
server# echo "nameserver 195.178.88.66" > resolv.conf
server# echo "tik.cesnet.cz" > ntpserver
```

Program `dhclient` potřebuje konfigurační soubor `dhclient.conf` [9]:

```
send dhcp-lease-time 1800;
request subnet-mask, routers;
require subnet-mask, routers;
timeout 24;
select-timeout 5;
backoff-cutoff 16;
initial-interval 8;
script "/bin/true";
```

Soubory pro práci s knihovnami

Soubor `/etc/ld.so.conf` musí obsahovat seznam adresářů s knihovnami, seznam jednotlivých knihoven pak vytvoří příkaz `ldconfig`. Tento příkaz se ale musí pustit ve správném adresáři, což zajistí parametr `-r`:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/
server# echo -e "/lib\n/usr/lib" > etc/ld.so.conf
server# ldconfig -r .
```

Výsledkem příkazu `ldconfig` je soubor `/etc/ld.so.cache`.

Soubor /etc/mtab

Soubor `/etc/mtab` je používán různými programy ke zjištění již připojených souborových systémů. Celý adresář `/etc` je ovšem k dispozici pouze pro čtení, tudíž připojení nebo odpojení souborových systémů se v něm nemůže projevit. Stejnou informaci o připojených souborových systémech ale dává dynamicky generovaný soubor `/proc/mounts`, takže stačí vyrobit symbolický odkaz:

```
server# cd /mnt/sda7/disks/suse-10.3-mini/etc/
server# ln -s ../proc/mounts mtab
```

Soubor /etc/profile

V souboru `/etc/profile` jsou základní nastavení pro příkazouvou řádku, provádí se ihned po úspěšném přihlášení uživatele, obsah souboru je jednoduchý:

```
export HOME=/root
export PATH=/bin:/sbin
export PS1="[\u@\h \W]# "
export PS2='> '
alias ll='ls -l'
cd $HOME
echo
```

7.3.8 Ostatní adresáře

Ostatní adresáře zůstávají prázdné:

- **/root** - domovský adresář uživatele *root*
- **/mnt** - adresář pro připojení dalších souborových systémů
- **/proc** - do tohoto adresáře se připojí souborový systém `proc`

7.3.9 Program syslogd

Spuštění programu `syslogd` zajišťuje soubor `/sbin/rc` pouštěný při startu počítače. Jeho nastavení je jednoduché, všechny zprávy se budou zapisovat do souboru `/var/log/messages`. Nastavení je uloženo v souboru `/etc/syslog.conf`:

```
*.*   /var/log/messages
```

7.4 Správa a zálohy jednotlivých distribucí

Správa jednotlivých distribucí může probíhat přímo na serveru. Pokud se má instalovat nový software nebo měnit konfigurace, nejsnadnější cesta vede přes program `chroot`. Ten změní kořenový adresář běžícího systému na jiný, zvolený. Např. má-li být do distribuce Debian instalován nový balík, stačí se na serveru pomocí `chroot` přepnout do adresáře s debianovskou distribucí a instalovat:

```
server# chroot /mnt/sda7/disks/debian-4.0
bash# apt-get balik
```

Úplně stejný postup lze použít např. pro testování některých příkazů, upgrady jednotlivých balíků atd. Tyto změny lze provádět buď přímo na serveru v ramdisku, který je přístupný protokolem *NFS* uživatelům na pracovních stanicích, čímž je možné měnit systém na pracovní stanici přímo za běhu, nebo lze změny provádět pouze na pevném disku a jeho obsah až po té znovu nahrát na ramdisk.

Aby však nedocházelo k poškození některé plně funkční distribuce, může být někdy vhodné celou distribuci zazálohovat. Nejsnadnější je udělat jeden soubor, který obsahuje celou distribuci, ten se pak dá dnadno zálohovat na CD/DVD, USB flash, pásku nebo poslat přes síť na jiný server.

```
server# cd /mnt/sda7/disks
server# tar -c -f debian-4.0.tar debian-4.0/
server# tar -c -f suse-10.3-mini.tar suse-10.3-mini/
```

7.4.1 Hesla lokálních uživatelů

Pro dosažení maximální jednoduchosti minidistribuce a alespoň nějakého stupně zabezpečení je heslo pro uživatele *root* zapsáno ve tvaru MD5 součtu přímo v souboru `/etc/passwd`. Upravený soubor `/bin/login` pak vytvoří ze zadaného hesla také MD5 součet a porovná ho s tímto záznamem. Aby nebylo příliš pracné heslo měnit, byl do minidistribuce přidán ještě soubor `/bin/passwd`, který je k vidění na DVD přiloženém k této práci.

Program funguje jednoduše: načte dvakrát heslo (bez zobrazování znaků na obrazovce), následně porovná délku hesla s nastaveným minimem a zjistí, zda obě zadaná hesla jsou totožná. Pokud vše proběhlo v pořádku, vytvoří opět MD5 součet hesla a přepíše soubor `/etc/passwd`.

Tento postup ovšem není možné provádět na pracovní stanici, protože do souboru `/etc/passwd` zde není možné zapsat. Tento postup lze tedy uplatnit pouze na serveru, kde ovšem hrozí přepsání souboru `/etc/passwd` přímo v systému celého serveru. Z toho důvodu je třeba nejprve změnit kořenový adresář a teprve poté měnit heslo:

```
server# chroot /mnt/sda7/disks/suse-10.3-mini
bash# passwd
Changing password for user root
New password: noveheslo
New password again: noveheslo
Password changed
```

Naprosto stejný postup pro změnu hesla platí i pro lokální uživatele (jak *root*, tak i ostatních) ve všech ostatních distribucích na serveru, změnu lze provádět pouze v distribuci na serveru a pouze s přepnutým kořenovým adresářem, aby nedošlo ke změně hesla lokálního uživatele serveru:

```
server# chroot /mnt/sda7/disks/debian-4.0
bash# passwd nobody
```

8 Ekonomické aspekty

8.1 Instalace nové pracovní stanice

Současný systém instalace pracovních stanic představuje pro administrátory velkou časovou zátěž. Instalace nové pracovní stanice probíhá tak, že administrátor má připravený obraz disku s operačním systémem Microsoft Windows, který nahraje na lokální disk dané stanice. Po úspěšném provedení takové akce ještě následuje nastavení sítě na stanici (na každé stanici se zadává přímo IP adresa a ostatní parametry). Pokud se zřizuje nová počítačová učebna, je třeba tento postup opakovat pro každý počítač zvlášť a navíc při nastavování jednotlivých stanic se nesmí udělat chyba. Zařizuje-li se např. počítačová učebna s 20 novými pracovními stanicemi, může jít o několikahodinovou nebo dokonce i o několikadenní práci.

Již zavedení služby *DHCP* do sítě představuje usnadnění této práce, administrátorovi by již mělo stačit přednastavit si obraz operačního systému tak, aby nastavení sítě bral protokolem *DHCP* a pak ho nebude muset nastavovat ručně. K instalaci nové stanice by tedy stačilo nahrát obraz operačního systému na pevný disk a korektně nastavit BIOS. Tím se samozřejmě sníží jak časová náročnost instalace, tak chybovost administrátora, navíc tuto práci zvládne i méně kvalifikovaný pracovník. Při vyšším počtu instalací se dá navíc pracovat s několika počítači najednou: během kopírování dat na disk se již může pracovník zabývat další stanicí, jediným předpokladem pro takovou práci je větší počet instalačních médií.

Oproti tomu je používání OS přes síť velmi snadná záležitost, stačí prostě nabootovat po síti již existující distribuci, která je přístupná po síti - žádné kopírování dat na disk, žádné nastavování sítě. Jediná práce na pracovní stanici v tomto případě je nastavení BIOSu, aby umožňoval bootování po síti.

8.2 Instalace nového softwaru na pracovní stanici

Instalace nového softwaru, úprava konfigurace již nainstalovaného softwaru, upgrade softwaru nebo vůbec zásahy do operačních systémů na stanici představují obrovský problém: buď se může provádět na jednom počítači za druhém nebo se může opravit předpřipravený obraz operačního systému (tj. stejný postup jako v případě instalace nové pracovní stanice). Obě možnosti znamenají opět vysokou zátěž pro administrátora.

Úprava OS Linux žádnou takovou zátěž nepředstavuje, protože stačí upravit distribuci na centrálním serveru. Některé zásahy jako jsou updaty některých nenáročných uživatelských programů se dokonce dají dělat za běhu. Velké updaty náročných softwarů jako např. kancelářského softwaru nebo webového prohlížeče se mohou udělat večer nebo ráno, kdy jsou učebny nevyužité. Administrátor pouze potřebuje mít přístup k distribuci na centrálním serveru.

Ještě snadnější je upgrade nebo oprava linuxového jádra. To stačí zkompilovat a dát na FTP server. Při novém nabootování pracovní stanice se již nahraje nové jádro. Jediná práce pro administrátora znamená nahrání souboru na FTP server.

8.3 Opravy operačního systému na pracovní stanici

V současnosti občas dochází k poškození operačního systému Microsoft Windows samotnými uživateli. Uživatelé mohou zaplnit disk, nainstalovat virus nebo poškodit nainstalovaný software. OS je sice částečně zabezpečen, aby k některým věcem měl přístup pouze administrátor a nikoliv běžný uživatel, přesto však k poškození dochází. Nepoužitelná stanice se opravuje tak, že se jednoduše nainstaluje znovu. To samozřejmě znamená, že administrátor musí přijít a opakovat časově náročný postup na instalaci stanice.

S OS Linux, který funguje *přes síť*, nic takového nehrozí. Celý souborový systém je přístupný v režimu **pouze pro čtení**, což je zajištěno přímo na serveru, takže uživatel na pracovní stanici ho nemůže nijak poškodit nebo upravit. Jádro navíc nemá podporu pevných disků, takže spolehlivě je ochráněn i lokální OS Microsoft Windows. Celý problém s chtěným nebo nechtěným poškozením instalovaných programů nebo jádra nebo se zaplněným diskem je tedy celkově eliminován.

8.4 Odstranění pevného disku ze stanic

Z předchozích odstavců jasně vyplývá, že práce s jednotlivými pracovními stanicemi je mnohem méně výhodná než práce s jedním serverem. Bohužel v současné době není na FAI UTB možné bootovat operační systém Microsoft Windows vzdáleně, tedy stejným nebo podobným způsobem jako OS Linux. Pokud by se našel způsob, jak OS Microsoft Windows bootovat bez použití pevného disku, daly by se tyto pevné disky ze stanic úplně odstranit. Takovým krokem by došlo k výraznému ušetření peněžních nákladů:

- snížení nákladů na pořízení pracovní stanice (cena pevného disku je v současné době kolem 1 000 Kč, takže vybavení jedné počítačové učebny o 20 počítačích za cenu 8 000 Kč znamená ušetření **12.5%**)
- snížení nákladů na lidskou práci při instalaci nových stanic (nyní zabere instalace jednoho počítače zhruba 30 minut, takže učebna o 20 stanicích znamená 10 pracovních hodin, v případě bezdiskových stanic by šlo pouze o nastavení BIOSu, asi 2 minuty na jednu stanici, což je celkem 40 minut - ušetření **96%**)
- snížení nákladů na lidskou práci v případě uprgadu software nebo instalaci nového software (nyní je třeba nový nebo upgradovaný software otestovat a připravit a pak nainstalovat tolikrát, kolik je počítačů v učebnách, v případě bezdiskových stanic by bylo třeba provést instalaci nebo upgrade softwaru pouze jednou - ušetření času potřebného na instalaci se zde **limitně blíží ke 100%**)

- eliminace nákladů na lidskou práci v případě poškození operačního systému nebo softwaru uživateli - ušetření **100%**

Odstraněním pevných disků dojde také ke snížení poruchovosti pracovních stanic - pevný disk je jedna z nejporuchovějších částí a porucha pevného disku a následná reklamace nebo koupě nového disku znamená:

- úplnou nemožnost používání pracovní stanice po celou dobu od poruchy do montáže nového nebo opraveného disku
- použití lidských zdrojů na demontáž a montáž pevného disku a další čas věnovaný reklamaci nebo nákupu nového disku
- v případě prošlé záruky peněžní náklady na nákup nového disku

8.5 Zkvalitnění výuky OS Linux

Nemalým přínosem pro FAI UTB je i zkvalitnění výuky. V některých předmětech se sice vyučuje OS Linux, ale studenti k tomuto operačnímu systému nemají na učebnách přístup - pokud chtějí získat praktické dovednosti, musí si ho instalovat sami na svých počítačích, což některé studenty velmi odrazuje.

Možnost nabootovat již existující OS Linux bez jakéhokoliv zásahu do pracovní stanice, nemožnost *rozbití* nainstalovaného systému a snadný přístup určitě znamená zvýšení znalostí studentů o tomto operačním systému. Navíc vyučující nemusí ukazovat práci s OS Linux jen teoreticky (na tabuli nebo na svém počítači), do OS Linux mohou nabootovat všichni na učebně.

Část IV
ZÁVĚR

Tato práce obsahuje především návod na výrobu prostředí, ve kterém je možné pracovat s bezdiskovými stanicemi, na kterých běží plnohodnotný operační systém Linux, tedy nejde pouze o terminál nebo jinou podobu vzdálené pracovní plochy, kdy všechny spouštěné programy obsluhuje centrální server, ale je plně využita kapacita hardwaru pracovní stanice. Centrální server pouze poskytuje souborový systém pro pracovní stanici pomocí protokolu *NFS* a slouží tedy spíše jako náhražka lokálního disku.

Takovéto řešení ovšem nespočívá pouze v úpravě linuxové distribuce a její zpřístupnění na síti, vyžaduje součinnost několika na sobě nezávislých hardwarových zařízení i softwarových služeb, bezchybné nastavení sítě a zabezpečení, která znemožňují uživatelům chtěná i nechtěná poškození jednotlivých komponent.

Pro zprovoznění celého systému je třeba začít nastavením sítě, tedy správným návrhem rozložení sítí a jejich adresace a volbou správných hardwarových síťových zařízeních, tedy směrovačů a přepínačů. Tato zařízení na FAI UTB již jsou a plně vyhovují nárokům současným a pravděpodobně i budoucím - navíc v případě zjištěného přetížení jednotlivých zařízení je možný jejich upgrade s minimálními změnami v konfiguraci.

Dalším krokem je zprovoznění, nastavení a zabezpečení centrálního serveru. Zde je větší pravděpodobnost, že v budoucnu může dojít k přetěžování serveru, což se dá ovšem řešit zvýšením počtu serverů nebo posílením hardwaru serveru.

Posledním krokem je příprava linuxové distribuce, která bude umět pracovat v síťovém režimu. Ta může být připravena buď z existujícího nainstalovaného systému nebo může být vytvořena *ručně*. Distribuce jsou snadno udržovatelné a spravovatelné.

Na tuto práci je možné navázat, a to v oblasti vzdáleného bootování jiných operačních systémů než je Linux. V současnosti se na pracovních stanicích používá operační systém Microsoft Windows, který je ovšem umístěn na každé stanici na lokálním pevném disku a správa většího počtu takových pracovních stanic je pro správce spíše noční můrou. Zavedením podobného řešení jako u OS Linux, tedy možnost bootování Microsoft Windows po síti by se velmi zjednodušila práce pro správce a navíc by se z pracovních stanic dal odstranit pevný disk, to by znamenalo přínos především v ušetření peněžních nákladů na pracovní stanice a také snížení poruchovosti pracovních stanic.

V případě úspěšného nasazení tohoto řešení v síti FAI UTB může dojít ke zkvalitnění výuky OS Linux, neboť tento systém by byl velmi snadno dostupný a spustitelný.

Conclusion

This work contains mainly instructions for building an environment with the possibility of working with diskless stations with OS Linux with its full capabilities, i.e. not just terminal or remote desktop, where all processes run on central server, but in this case, all capabilities of all hardware included in the workstation are fully utilized. Central server is present just for providing the filesystem for workstations over the *NFS* protocol, that means it is here instead the local hard drive.

Such solution lies not in modification of a linux distribution and sharing it over the network, the solution requires cooperation of some independent hardware equipment and software services, error-free network configuration and security arrangements which avoid users from doing wanted and unwanted damage to any component.

To launch all the system it is neccessary to start with the network configuration: correct network design and addressing and choosing of good hardware equipment (switches and routers). Current equipment in FAI UTB fully corresponds with actual requirements and probably with future ones too. In case of rising of some overload, there is the possibility to easily upgrade to *higher* devices with a very few changes to running configuration.

Next step is running, configuring and securing the central server. This is the most loaded device, so there can be some overload sooner as overload in network equipment. Simple solution is running more central servers together or upgrade proper part of hardware.

Last step is preparation of linux distribution, which should be completely prepared for running in network environment. It can be prepared from a running distribution or can be made *by hand*. Distributions are easy maitainable and manageable.

It is possible to extend this work, primarily in running other operating systems than Linux in the same environment. Currently the most used operating system on workstations is Microsoft Windows, which is located on every workstation on its harddrive - maintaining of high count of such workstation could be a real nightmare for administrator. If there was such a solution for this operationg system allowing diskless running of the operating system, the administrator would have much less work and the workstation could be really diskless. That would save financial expenses for workstations and the workstations would be less faulty.

In case of successfull launching of this solution in the FAI UTB network, the quality of education of OS Linux can be increased, because this operating system would be very simple acciessible and executable.

Část V
SEZNAM POUŽITÉ LITERATURY

[1] DROMS, Ralph. *RFC 2131 (rfc2131) - Dynamic Host Configuration Protocol* [online]. Bucknell University, March 1997 [cit. 2008-05-01]. Text v angličtině. Dostupný z WWW: <http://www.faqs.org/rfcs/rfc2131.html>.

[2] *DHCP snooping* [online]. Wikimedia Foundation, Inc., 2008, last modified on 9 April 2008, at 03:02 [cit. 2008-04-15]. Text v angličtině. Dostupný z WWW: <http://en.wikipedia.org/wiki/DHCP_snooping>.

[3] Cisco Systems, Inc.. *Catalyst 2950 Desktop Switch Software Configuration Guide : 12.1(9)EA1* [online]. c1992-2007, Sat May 05 09:48:18 PDT 2007 [cit. 2008-01-27]. Text v angličtině. Dostupný z WWW: <http://www.cisco.com/en/US/docs/switches/lan/catalyst2950/software/release/12.1_9_ea1/configuration/guide/scg.html>.

[4] Cisco Systems, Inc.. *Catalyst 3550 Multilayer Switch Software Configuration Guide : Release 12.1(20)EA2* [online]. c1992-2007, Sun Jul 01 05:16:48 PDT 2007 [cit. 2008-01-27]. Dostupný z WWW: <http://www.cisco.com/en/US/docs/switches/lan/catalyst3550/software/release/12.1_20_ea2/configuration/guide/3550scg.html>.

[5] ANVIN, H. Peter. *SYSLINUX : Syslinux wiki* [online]. 2008, last modified 23:51, 25 April 2008 [cit. 2008-05-25]. Text v angličtině. Dostupný z WWW: <http://syslinux.zytor.com/wiki/index.php/SYSLINUX>.

[6] ANVIN, H. Peter. PXELINUX : Syslinux wiki [online]. 2008, last modified 09:37, 28 April 2008 [cit. 2005-05-01]. Text v angličtině. Dostupný z WWW: <http://syslinux.zytor.com/wiki/index.php/PXELINUX>.

[7] KUHLMANN, Gero, MAREŠ, Martin, SCHOTTELIUS, Nico. *Linux-2.6.17/Documentation/nfsroot.txt* [online]. [2007] [cit. 2008-04-28]. Text v angličtině. Dostupný z WWW: <http://www.gelato.unsw.edu.au/lxr/source/Documentation/nfsroot.txt>.

[8] ANVIN, H. Peter. *Tftpd(8) - Linux man page* [online]. [2007] [cit. 2008-04-10]. Text v angličtině. Dostupný z WWW: <http://linux.die.net/man/8/tftpd>.

[9] LEMON, Tom. *Dhclient(8) - Linux man page* [online]. Internet Systems Consortium,Inc., [2007] [cit. 2008-04-15]. Text v angličtině. Dostupný z WWW: <http://linux.die.net/man/8/dhclient>.

[10] FAWCETT, Tom. *The Linux Bootdisk HOWTO* [online]. v4.5. c1995-2002, January 2002 [cit. 2008-04-02]. Text v angličtině. Dostupný z WWW: <http://tldp.org/HOWTO/Bootdisk-HOWTO/>.

[11] DUBEC, Michal. *LAN bezpečnost a ověřování identity* [online]. [2007], Last-Modified: Thu, 16 Aug 2007 11:03:05 GMT [cit. 2008-02-05]. Dostupný z WWW: <http://www.alefnula.cz/downloads/KC/KC-identita.pdf>.

[12] *Mount(8) - mount file system : Linux man page* [online]. [2008] [cit. 2008-05-03]. Dostupný z WWW: <http://linux.die.net/man/8/mount>.

[13] BON, Stef. *HOWTO execute scripts at begin and end of a usersession using PAM with examples* [online]. [1008] [cit. 2008-05-14]. Text v angličtině. Dostupný z WWW: <http://linux.bononline.nl/linux/pamscript/01/build.html>.

[14] BIONDO, Giuseppe, VAN MEER, Roel. *LDAP Implementation HOWTO* [online]. v0.5. 2000-2001 [cit. 2005-05-14]. Text v angličtině. Dostupný z WWW: <http://www.faqs.org/docs/Linux-HOWTO/LDAP-Implementation-HOWTO.html>.

[15] ANVIN, H. Peter. *PXELINUX - SYSLINUX for network boot* [online]. c1994-2007 [cit. 2008-01-27]. Text v angličtině. Dostupný z WWW: <http://syslinux.zytor.com/pxe.php>.

[16] Internet Systems Consortium, Inc.. *Dynamic Host Configuration Protocol (DHCP)* [online]. c2007 , 2007-12-19 [cit. 2008-01-27]. Text v angličtině. Dostupný z WWW: <http://www.isc.org/index.pl?/sw/dhcp/>.

[17] SMITH, Christopher M.. *Linux NFS faq* [online]. 2008 [cit. 2008-01-27]. Text v angličtině. Dostupný z WWW: <http://nfs.sourceforge.net/>.

[18] Linux Kernel Organization, Inc.. *The Linux Kernel Archives* [online]. [1994], 2008-01-24 23:18 UTC [cit. 2008-01-27]. Dostupný z WWW: <http://www.kernel.org/>.

Část VI
SEZNAM OBRÁZKŮ

Část VII
SEZNAM POUŽITÝCH ZKRATEK

BIOS Basic Input/Output System - program umístěný v chipsetu základní desky počítače, má na starosti inicializaci a test jednotlivých zařízení jako jsou procesor, paměť, síťová karta atd. Spuštěním počítače se spustí právě tento program.

IP Internet Protocol - základní soubor pravidel pro komunikaci počítačů po síti Internet

PXE Preboot Execution Environment - prostředí umožňující zavést operační systém do počítače pomocí sítě, nezávisle na existenci nebo nastavení pevných disků

OS Operační systém - sada programů umožňující uživateli plnohodnotnou práci s počítačem, má na starosti obsluhu hardwarových zařízení na základě pokynů uživatele nebo běžících programů

DHCP Dynamic Host Configuration Protocol - protokol, který umožňuje počítačům v síti získat nastavení vlastního síťového rozhraní na základě údajů, které dostane od serveru

TFTP Trivial File Transfer Protocol - protokol, který umožňuje jednoduchý přenos souborů po síti mezi klientem a serverem (oběma směry), jednoduchost spočívá v používaní malých UDP paketů pro přenos a v neexistenci jakékoliv autorizace

NFS Network File System - protokol umožňující sdílení souborů a adresářů na bázi server-klient, klientská stanice pak k souborům na NFS serveru přistupuje přes síť, jakoby přistupovala k souborům na lokálním disku

USB Universal Serial Bus - typ datové směrnice v počítači, umožňuje připojování nejrůznějších druhů hardwarových zařízení včetně jejich napájení, na jedné sběrnici může být až několik desítek takových zařízení

LDAP Lightweight Directory Access Protocol - protokol pro centralizovanou správu uživatelů na bázi server-klient, na serveru je databáze s uživateli a jejich vlastnostmi, klient se pak dotazuje serveru a tím provádí autorizaci uživatelů, protokol má navíc možnost hierarchické struktury dat

IDE Integrated Drive Electronics - starší technologie umožňující připojení datových hardwarových zařízení (např. disků, CD-ROM) na základní desku počítače

SATA Serial Advanced Technology Attachment - novější technologie umožňující sériové připojení datových hardwarových zařízení (např. disků, CD-ROM) na základní desku počtače