

UNIVERZITA PARDUBICE  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2009

Jan Vlastník

UNIVERZITA PARDUBICE  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

## **Správa virtuálních serverů**

Jan Vlastník

**Bakalářská práce**

**2009**

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Katedra informačních technologií  
Akademický rok: 2008/2009

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jan VLASTNÍK**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Správa virtuálních serverů**

### Z á s a d y p r o v y p r a c o v á n í :

Teoretická část: \* Vytvořte přehled virtualizačních technologií i s konkrétními zstupci v oblasti SW (komerční i nekomerční). \* Rozdělte tyto technologie podle vhodnosti nasazení v konkrétních podmínkách. \* Stanovte rizika spojená s těmito technologiemi a jak se s nimi vypořádat. \* Porovnejte jednotlivé technologie s klasickým přístupem (co služba, to fyzický server). Praktická část: \* Vyberte vhodné virtualizační řešení pro provozování serverů se zaměřením na vývoj webových aplikací. \* Vybrané řešení zprovozněte a uveďte problémy, které bylo potřeba vyřešit navíc oproti klasickému přístupu, a také které problémy zase odpadly. \* Vyřešte též otázku zálohování virtuálních serverů a uveďte rozdíly oproti klasickému přístupu. \* Zhodnoťte přínosy použití virtualizace na tomto konkrétním příkladu využití.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

**Literatura:** \* **Virtualization.** 2008 [cit. 2008-09-21].  
URL: <http://en.wikipedia.org/wiki/Virtualization> \* **Xen.** 2008 [cit. 2008-09-21]. URL: <http://en.wikipedia.org/wiki/Xen> \* **VMware server.** 2008 [cit. 2008-09-21]. URL: [http://en.wikipedia.org/wiki/VMware\\_Server](http://en.wikipedia.org/wiki/VMware_Server) \* **Linux V-Server.** 2008 [cit. 2008-09-21]. URL: <http://en.wikipedia.org/wiki/Linux-VServer> \* **QEMU.** 2008 [cit. 2008-09-21]. URL: <http://en.wikipedia.org/wiki/QEMU> \* **VirtualBox.** 2008 [cit. 2008-09-21]. URL: <http://en.wikipedia.org/wiki/VirtualBox>

Vedoucí bakalářské práce:

**Mgr. Tomáš Hudec**

Katedra informačních technologií

Datum zadání bakalářské práce: **15. ledna 2009**

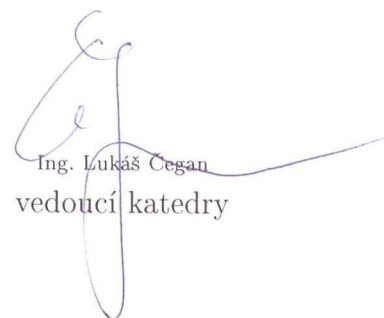
Termín odevzdání bakalářské práce: **15. května 2009**



doc. Ing. Simeon Karamazov, Dr.  
děkan



L.S.



Ing. Lukáš Čegan  
vedoucí katedry

V Pardubicích dne 31. března 2009

## Prohlášení

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 10.08.2009

Jan Vlastník

## Poděkování

Chtěl bych poděkovat panu Mgr. Tomáši Hudcovi za odborné vedení a pomoc při tvorbě této bakalářské práce.

Dále bych rád poděkovat společnosti Ebrána, s. r. o. za projevenou důvěru a maximální spolupráci v rámci realizace praktické části této práce a v neposlední řadě jejich správci IT panu Ladislavu Novákovi za jeho nadšení a chuť pronikat do nových technologií.

Speciální poděkování věnuji mé ženě za podporu a trpělivost s jakou tolerovala můj dvouletý výzkum a za její pomoc v rámci finálních korektur této práce.

## **ANOTACE**

Cílem práce je popis současné situace v oblasti virtualizačních technologií. V praktické části se zabývá optimalizací architektury firemních serverů pro menší firmu se zaměřením na vývoj webových produktů, za využití virtualizačního nástroje VMware Server. Součástí optimalizace je i vyřešení problematiky zálohování a případné obnovy dat.

## **KLÍČOVÁ SLOVA**

virtualizace, virtuální stroj, server, zálohování, prevence výpadků

## **TITLE**

Virtual server administration

## **ANNOTATION**

The aim of this work is to describe current situation in the field of virtualization technologies. Practical part deals with server architecture optimization for a small-size company focused on web products development; VMware Server virtualization tool was used. Data backup and recovery issues are also included as an integral part of the optimization process.

## **KEYWORDS**

virtualization, virtual machine, server, backup, failure prevention

# Obsah

<b>Úvod</b>	<b>8</b>
<b>1 Teoretická část</b>	<b>9</b>
1.1 Virtualizace a její užití . . . . .	9
1.1.1 Co je to virtualizace . . . . .	9
1.1.2 Proč virtualizovat . . . . .	9
1.1.3 Možné varianty virtualizace . . . . .	10
1.2 Přehled virtualizačních nástrojů . . . . .	12
1.2.1 Xen . . . . .	12
1.2.2 Qemu . . . . .	12
1.2.3 VirtualBox . . . . .	13
1.2.4 Linux-VServer . . . . .	13
1.2.5 VMware . . . . .	14
1.2.6 Rekapitulace . . . . .	15
<b>2 Praktická část</b>	<b>17</b>
2.1 Analýza stávajícího řešení . . . . .	17
2.1.1 Sumarizace potřeb firmy . . . . .	17
2.1.2 Navržení optimálního řešení . . . . .	19
2.2 Instalace a konfigurace . . . . .	22
2.2.1 Hostitelský operační systém . . . . .	22
2.2.2 Virtuální operační systém . . . . .	23
2.3 Testování a optimalizace . . . . .	24
2.3.1 Ladění před uvedením do ostrého provozu . . . . .	24
2.3.2 Ladění v ostrém provozu . . . . .	25
2.4 Systém zálohování firemních dat . . . . .	27
2.4.1 Hlavní požadavky . . . . .	27
2.4.2 Nástroje . . . . .	28
2.4.3 Navržený systém zálohování . . . . .	29
<b>Závěr</b>	<b>30</b>
<b>Seznam užitých zkratk a pojmů</b>	<b>31</b>
<b>Literatura</b>	<b>34</b>



# Úvod

„V poslední době se v odborné i laické informatické literatuře stále častěji vyskytuje slovo virtualizace, zpravidla v souvislosti s procesory, případně s úložným prostorem. Až příliš často je virtualizace, zejména v polo-odborné literatuře, prezentována jako nový koncept, který pomůže vyřešit když ne všechny, tak velkou část praktických problémů, které jsou spojeny s nasazením počítačů.“ [1]

Tato práce si klade za cíl popsat současnou situaci v oblasti virtualizačních technologií a dostupnosti virtuálních serverů pro menší firmy, které nemají dostatek prostředků na pořízení profesionálního řešení poskytovaného specializovanou firmou.

V praktické části pak představím řešení virtualizace, které jsem realizoval ve společnosti Ebrána, s. r. o. Součástí tohoto řešení je vstupní analýza, postup virtualizace, preventivní opatření pro zabezpečení bezproblémového chodu a výčet možných komplikací doplněný o doporučení jak dané problémy řešit.

Vzhledem k tomu, že stejně jako v případě fyzických serverů, tak i u virtuálních systémů je nejakutnějším problémem ztráta dat, věnuji se v této práci i problematice zálohování a minimalizaci doby případných výpadků.

# Kapitola 1

## Teoretická část

### 1.1 Virtualizace a její užití

#### 1.1.1 Co je to virtualizace

„Původní význam termínu virtualizace, pocházející z 60. let 20. století, je vytváření virtuálních strojů za pomoci kombinace hardwaru a softwaru. Virtualizace platformy je prováděna na dané hardwarové platformě pomocí softwaru hostitele (řídící program), který vytváří simulované prostředí počítače (virtuální stroj) pro hostovaný software. Software hosta, což často bývá celý operační systém, běží, jako by byl nainstalován na samostatné hardwarové platformě.“ [18]

Virtualizaci nerepresentují pouze virtuální stroje. Dalšími příklady mohou být virtuální paměť, mechanika, peníze, realita a dokonce i virtuální svět. Všechny tyto pojmy spojuje potřeba simulovat reálný subjekt abstraktní formou. Abstraktní forma na první pohled nerozeznatelná od reálné předlohy je v dnešní době omezených zdrojů nezbytnou součástí informačních technologií.

#### 1.1.2 Proč virtualizovat

**Efektivita** Hlavním důvodem proč virtualizovat je efektivnější využívání fyzických zdrojů. Nejlepším příkladem bude přirovnání nevirtuálního serveru vybaveného několika procesory a dostatečným množstvím operační paměti k jaderné elektrárně, pak ekvivalentem odběratele elektrického proudu bude provozovaný operační systém. Pro úplné upotřebení vyprodukované elektrické energie, musí být k dispozici adekvátní množství odběratelů, jinak dochází k plýtvání.

**Centralizace a vzdálená správa** Spočívá v možnosti spravovat více serverů z jednoho místa. Většina nástrojů umožňujících virtualizaci poskytuje administrační roz-

hraní, které obvykle podporuje funkce spuštění, vypnutí, restart či instalaci nového serveru, bez nutnosti fyzického přístupu do místnosti se servery. Nespornou výhodou je možnost vzdálené úpravy virtuálního HW. Například lze virtuálnímu PC aktuálně alokujícímu 2 GB paměti virtuální RAM s dlouhodobým požadavkem 250 MB RAM snížit alokovanou paměť na 512 MB a získané zdroje pak přerozdělit mezi potřebnější.

**Úspora místa a energií** Bude-li si chtít společnost pořídit 20 fyzických serverů, musí zabezpečit dostatečně klimatizovanou a prostornou místnost se stabilním zdrojem elektrického proudu (dostatečný počet záložních zdrojů atp.). Oproti tomu virtualizace nabízí provozování 20 virtuálních OS na jednom fyzickém serveru, což by znamenalo značné snížení dlouhodobých nákladů na energie a upgrade HW prostředků.<sup>1</sup>

**Přenositelnost** Představme si, že společnost zabývající se vývojem SW má k dispozici tři různě výkoné fyzické servery. První je schopen obsloužit 1000, druhý 10 000 a třetí 20 000 klientů. Vydá-li takováto společnost například bezpečnostní záplatu, její klienti si ji začnou okamžitě stahovat. Virtualizace nám umožňuje začít nabízet daný produkt na nejslabším serveru a migrovat celý virtuální OS na výkonější server v okamžiku, kdy by hrozilo úplné vytížení, tzn. v okamžiku, kdy by server obsluhoval 900 zákazníků, aniž by došlo k přerušení obsluhy právě obsluhovaných.

**Zpřístupnění zdrojů** Typickým příkladem je pronájem virtuálních serverů v rámci hostingových služeb. Díky virtuálním OS můžete dnes získat vlastní webhostingový server za minimální cenu a s takovým pak nakládat dle libosti, aniž byste ovlivnili nebo snad i poškodili hostitelský OS či jiné hostované virtuální OS.

### 1.1.3 Možné varianty virtualizace

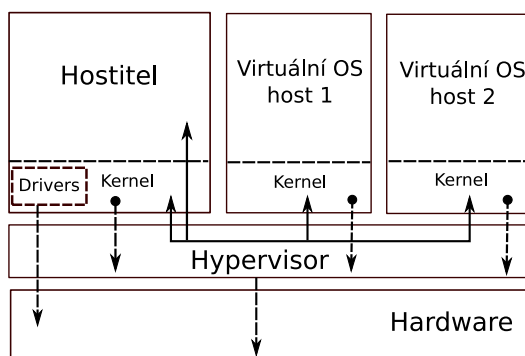
**Virtualizace na úrovni jádra** „Je společně s paravirtualizací považována za neefektivnější typ virtualizace. Virtualizovaná prostředí běží nad společným jádrem, které má přímý přístup k fyzickému hardwaru, tudíž se nevytváří virtuální zařízení ani speciální přístupové API. Režijní ztráty jsou způsobeny pouze oddělením procesů, diskových prostorů a síťového provozu serverů.“ [13]

**Emulace** Je založena na dynamickém překladu instrukcí pro emulovaný HW, a proto se jedná o nejpomalejší způsob virtualizace. Hlavní využití nachází v případě kdy nemáme k dispozici požadovaný HW.

---

<sup>1</sup>zvýšíme-li výkon jednoho fyzického PC, posílíme tím několik virtuálních OS

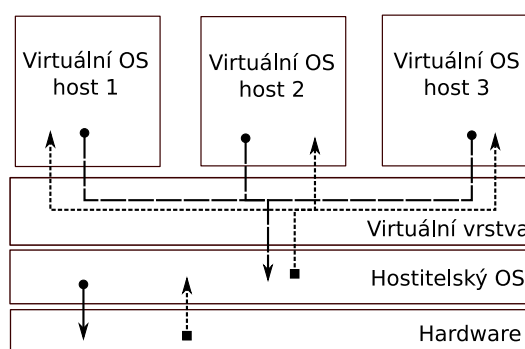
**Paravirtualizace** Nevytváří se kompletní virtuální hardware, nástroje pouze umožňují komunikovat upravenému jádru virtualizovaného operačního systému s fyzickým hardwarem přes speciální API upraveného jádra hostitelského systému. Systémová volání probíhají přes hypervisor (odtud název hypervisorů používaný pro paravirtualizační nástroje). Nejznámějším zástupcem určeným pro Linux je Xen. Dalšími jsou VMware ESX Server nebo Win4Lin 9x. Systémy na bázi paravirtualizace dosahují vysoké efektivity. [13]



Obrázek 1.1: Grafické znázornění metody paravirtualizace.

**Úplná virtualizace** „Pro plnou virtualizaci je potřeba podpora jak ze strany procesoru, tak ze strany základní desky a BIOSu. Nevýhodou plné virtualizace narušil od paravirtualizace je nedokonalé využití prostředků hostitelského systému. Paměť totiž nelze dynamicky rozdělovat mezi hostovanými systémy a tak dochází k plýtvání.

Naopak nespornou výhodou takového řešení je, že jsme schopni spustit systém bez jakékoliv změny. Toto oceníme zejména v momentě kdy chceme virtualizovat systém s uzavřenými zdrojovými kody, jako například OS rodiny Windows.“ [17]



Obrázek 1.2: Grafické znázornění metody simulace a úplné virtualizace.

## 1.2 Přehled virtualizačních nástrojů

„Virtualizace je proces, kdy v jednom operačním systému spustíte pomocí virtualizačního systému jiný OS.“ [4]

V této kapitole stručně popíše několik nejrozšířenějších nástrojů umožňujících virtualizaci a speciálně provozování virtuálních OS. Vybrané nástroje jsou typickými reprezentanty jednotlivých typů virtualizace popsanych v sekci 1.1.3 na straně 10.

### 1.2.1 Xen

Vývoj projektu Xen byl zahájen na univerzitě Cambridge a již v roce 2003 byla vydána jeho první verze. Aktuálně je udržován a rozvíjen jako open-source a licencován pod GNU General Public License (GPL2). [2]

Xen umožňuje jak paravirtualizaci, tak i úplnou virtualizaci. V případě plné virtualizace však vyžaduje specifické hardwarové prostředky, např. CPU s podporou plné virtualizace (Intel ji nazývá Vanderpool, AMD tuto technologii označuje názvem Pacifica [4]).

Aby bylo možné Xen využívat, je potřeba upravit jádro hostitelského OS, proto je provozován pouze na OS s modifikovatelným jádrem, jako je např. Linux.

Z výše uvedeného tedy vyplývá, že Xen není nejvhodnějším kandidátem pro provozování virtuálních windowsových serverů (pokud nemáme k dispozici speciální HW, abychom využili podpory Xen pro úplnou virtualizaci). [3, 5]

### 1.2.2 Qemu

Emulátor procesoru QEMU vytvořil Fabrice Bellard jako open-source a je nyní veden pod licencí GNU Lesser General Public License (GNU LGPL). [8]

QEMU nepodporuje úplnou virtualizaci. Místo ní však nabízí tzv. uživatelskou virtualizaci a jistou formu úplné virtualizace.

Uživatelská virtualizace spočívá ve vytvoření virtuálního prostředí vhodného pro instalaci a následný provoz konkrétní aplikace, která by jinak nemohla být spuštěna v daném operačním systému z důvodu nekompatibility HW.

Druhou variantou je tzv. úplná virtualizace (HW prostředky virtuálnímu OS poskytuje formou dynamického překladu instrukcí, tzn. HW není skutečný, ale pouze emulovaný). V tomto případě nám nástroj QEMU nasimuluje takové prostředí jaké si vyžaduje virtuální OS, který se v něm chystáme provozovat a to bez ohledu na architekturu hostitelského systému. Virtuální OS pak bude v rámci toho prostředí existovat v jakési izolaci vůči svému okolí.

Pro zprovoznění emulátoru QEMU stačí pouze nainstalovat software bez nutnosti úpravy jádra fyzického OS, jak tomu bylo například u virtualizace pomocí Xen.

Výkon virtuálního OS provozovaného pomocí QEMU odpovídá 10 % až 20 % běžného výkonu PC, na kterém je virtualizace provozována. Výkon QEMU lze zvýšit pomocí akcelerátoru kQEMU. [9]

QEMU je vhodná volba pro testování OS a aplikací v nich, případně provozování samotných aplikací určených pro jiný než aktuálně nainstalovaný OS.

### 1.2.3 VirtualBox

Multiplatformní projekt VirtualBox je aktuálně vyvíjen společností Innotek. Jeho distribuce je jak komerční, tak i nekomerční<sup>2</sup>. Komerční verze je oproti volné variantě rozšířena o podporu virtuálního USB rozhraní, vzdáleného běhu virtuálních strojů a sdílených složek umožňujících výměnu dat mezi virtuálním a hostitelským OS. [10]

Jádrem virtualizace je QEMU, tzn., že pro instalaci je potřeba pouze zkompileovat zdrojový kód aplikace. Navíc je nutné do jádra zavést modul vboxdrv, který zajistí nízkoúrovňovou komunikaci s fyzickým hardware. [11]

### 1.2.4 Linux-VServer

„Linux-VServer je virtualizací na úrovni jádra OS. V souvislosti s Linux-VServerem se servery nazývají bezpečnostní kontexty (security contexts), zkráceně kontexty, v kterých je spuštěn a provozován virtuální server (VPS). Každý běžící VPS má svoji vlastní správu nezávislou na ostatních. Systém využívá a rozšiřuje již existující vlastnosti a služby jádra. Jsou to zejména Linux capabilities, omezení výpočetních zdrojů (resource limits), atributy souborů (file attributes) a chroot.

Autoři jádro rozšířili o podporu kontextů, v kterých jsou procesy spuštěny, izolovány a zneviditelněny pro jiné kontexty. Pro usnadnění administrace existuje hostitelský kontext (Host context), v němž běží vlastní systém, a pro přehled procesů všech VPS je vytvořen Spectator kontext. Oba tyto kontexty jsou v současnosti implementovány jako jeden administrátorský kontext nazývaný hostitelský.

Jádro obsahuje rozšíření pro virtualizaci síťového rozhraní. Linux-VServer nepoužívá virtuální zařízení, které by zvýšilo zátěž systému, ale speciální techniky umožňující přidělovat IP adresy procesům.

---

<sup>2</sup>Nekomerční větev je šířena pod licencí GNU GPL.

Pro oddělení diskových oblastí projekt používá upravené původní rozhraní chroot. Pro každý VPS se při spuštění uchovává nejvyšší možný adresář (kořenový adresář serveru), čímž je zabráněno úniku z privátní diskové oblasti“. [14]

Hlavní nevýhodou tohoto typu virtualizace je možnost provozovat pouze operační systémy stejného typu jako upravené jádro. [13]

### 1.2.5 VMware

Americká společnost VMware, Inc., je v současné době asi největším hráčem na trhu v oblasti virtualizace.

VMware nazývá fyzický počítač hostitelským počítačem. Virtuální operační systémy dělí se o prostředky hostitelského počítače pak označuje jako hosty. Tato terminologie se týká obou řad produktů (tj. serverových i desktopových řešení).

VMware, podobně jako emulátor software, poskytuje hostům kompletní sadu virtualizovaného hardwaru počínaje procesorem, paměťmi, přes grafické adaptéry až po síťové karty a ovladače disků.

„Díky tomu jsou virtuální stroje VMware snadno přenositelné mezi různými hostitelskými počítači, protože každý vypadá pro hosta velmi podobně. V praxi to znamená, že administrátor systému může pozastavit operaci prováděnou virtuálním strojem, zkopírovat ho na jiný, např. výkonnější hostitelský počítač, a pokračovat tam, kde byl přerušen. Popřípadě, pro nejvyšší řadu produktu, je k dispozici funkce nazývaná VMotion umožňující přenášet pracující hostující stroj mezi počítači se shodnou konfigurací, ale různým fyzickým hardwarem, sdílejícími společně SAN.“ [7]

**VMware Player** – přehrávač virtuálních OS vytvořených produkty VMware. Nepodporuje možnost instalace nového hosta a má omezené možnosti konfigurace již hotového a jím přehrávaného virtuálního OS.

**VMware Workstation** – vydala společnost VMware jako svůj první produkt v roce 1999. Tento software umožňuje uživateli používat různé instance operačních systémů architektury x86 na jednom PC. [7]

**VMware Ace** – oproti VMware Workstation umožňuje superuživateli omezit práva běžných uživatelů. Tento produkt byl v kombinaci s VMware Player pilotní virtualizací v rámci PC učeben na Fakultě elektrotechniky a informatiky Univerzity Pardubice.

**VMware Server 1** – tento nástroj umožňuje vytvářet virtuální OS. Hlavní výhodou v rámci provozování serverů je oproti VMware Workstation běh těchto serverů

jako služby v rámci hostitelského OS. Konfigurace virtuálních serverů se provádí pomocí speciálního produktu VMware Console. Pomocí tohoto nástroje se můžeme připojit k libovolnému hostitelskému OS s podporou virtualizace na bázi VMware Server 1. Po přihlášení pak můžeme spouštět, restartovat nebo vypínat již existující virtuální hosty. Rovněž nám umožňuje servery instalovat, konfigurovat, přidávat virtuální HW prostředky, defragmentovat virtuální disky, vytvářet snapshoty atp.

**VMware Server 2** – oproti VMware Server 1 již nepodporuje VMware Console a jeho správa je možná pouze pomocí internetového prohlížeče. Díky tomu je potřeba na hostitelský systém nainstalovat v případě linuxového OS Apache. Další změnou je vyjmutí fyzické partition z nabídky dostupného HW, aktuálně nabízí pouze možnost připojit virtuální disk SCSI. VMware Server 2 rovněž neumožňuje defragmentaci virtuálních disků.

### 1.2.6 Rekapitulace

V závěru této kapitoly se pokusím o stručný souhrn a srovnání jednotlivých produktů. Aby toto bylo možné, je potřeba dané nástroje nejdříve rozdělit do několika skupin.

První skupinu tvoří Xen a VMware Server první a druhé generace. Tyto dva produkty jsou vhodnými kandidáty pro provozování profesionálních serverů na jednom fyzickém PC metodou úplné virtualizace. VMware oproti Xenu je multiplatformní. Hlavní výhodu produktů od společnosti VMware vidím v tom, že tato firma se zabývá pouze problematikou virtualizace, což je nejpatrnější na minimální chybovosti jejich produktů a velkým náskokem před konkurencí. Xen však oproti VMware nabízí paravirtualizaci.

Do druhé skupiny bych zařadil QEMU, VirtualBox a VMware Workstation. Tyto produkty jsou vhodné pro instalaci a spuštění libovolného OS, se kterým pak může uživatel pracovat, aniž by ovlivnil hostitelský systém.

Poslední skupina obsahuje Linux-VServer. Jedná se o virtualizaci na úrovni sdílení linuxového jádra mezi hostitelským OS a hostovanými systémy, a proto tento typ virtualizace nelze aplikovat, pokud bychom potřebovali v rámci virtualizace provozovat jiný než linuxový OS.



Produkt	Typ virtualizace	Výkon
Xen	Paravirtualizace, úplná virtualizace	paravirtualizace velmi rychlá, úplná virtualizace středně
VMware Server	Úplná virtualizace	střední/rychlý
VirtualBox	Úplná virtualizace	rychlý/velmi rychlý (komerční verze)
Linux-VServer	Na úrovni jádra	rovnocenný s běžným OS
QEMU	Emulace	pomalý, střední s podporou kQEMU

Tabulka 1.1: *Sumarizace – typ virtualizace a výkon. [19]*

# Kapitola 2

## Praktická část

Tématem této práce je využití virtualizace v rámci provozu serveru se zaměřením na vývoj webových aplikací ve společnosti Ebrána, s. r. o. Praktická část je rozdělena do čtyř základních celků – analýza, instalace, testování a ladění a optimalizace metodiky zálohování.

### 2.1 Analýza stávajícího řešení

Společnost Ebrána, s. r. o., se zaměřením na vývoj webových aplikací (e-shop, CMS, IS pro realitní společnosti atp.), do které jsem v roce 2006 nastoupil jako programátor vývojář, měla k dispozici server s fyzickým OS Linux RH 7, který poskytoval služby WWW server, MySQL server, SAMBA server, datový a FTP server.

Obsluhu serveru zabezpečovala externí firma, a proto vedení společnosti uvítalo mou nabídku optimalizace stávajícího řešení.

#### 2.1.1 Sumarizace potřeb firmy

**Seznam základních požadavků:**

- možnost bezpečné aktualizace sw prostředků,
- minimální doba výpadku,
- zálohování dat a jejich obnova,
- minimální finanční náklady potřebné na realizaci,
- zabezpečení poskytování dosavadních služeb.

### **Popis služeb původního serveru**

Původní server poskytoval služby všem firemním oddělením. Mezi tyto patřily administrativní, obchodní, vývojové, výrobní a grafické oddělení. Server tedy fungoval jako testovací stroj a zároveň jako stabilní systém pro správu firemní agendy. Dalším problémem byly občasné výpadky způsobené přetížením serveru zakončené pádem nebo zamrznutím systému.

Východiskem z této situace byl buď nákup dalších fyzických serverů nebo virtualizace.

V následujícím textu shrnu základní požadavky, které jsou kladeny na výsledné řešení. Každý z požadavků se pokusím rozebrat a nastínit možné řešení s případným využitím virtualizace.

### **Požadavek – aktualizace SW prostředků**

Upgrade OS, databáze MySQL, PHP a dalších služeb může ovlivnit fungování firemních produktů. Ideální postup spočívá v realizaci dané změny na izolované kopii OS a následném testování.

Virtuální OS stačí jednoduše zduplikovat, provést na kopii upgrade, ladění, testování a po dokončení aktualizace zbývá jenom vypnutí původního virtuálního OS a nastavení původní IP adresy na jeho kopii.

### **Požadavek – minimální doba výpadku**

Nákup druhého fyzického serveru by byl v tomto případě řešením pouze pro administrativní a obchodní oddělení, ale neřešil by potřebu fyzického přístupu k serverům za účelem odstranění případných problémů.

Virtualizace oproti tomu nabízí jak oddělené a na sobě nezávislé OS, tak i vzdálenou správu virtuálních serverů. Díky vzdálené správě již není třeba restartovat celý server tzv. hard resetem, ale pouze softwarovým restartem problémového virtuálního hosta s využitím protokolu ssh a terminálu.

V případě, kdy by došlo k trvalému výpadku hardwarové komponenty, virtuální OS lze přenést na jakékoliv dostupné PC s podporou dané virtualizace. V případě fyzického serveru je třeba buď zajistit identickou HW komponentu (délka výpadku se odvíjí od doby dodání HW komponenty) nebo reinstalací celého OS a následnou konfigurací systému (délka výpadku se odvíjí od složitosti konfigurace využívaných aplikací).

Pro přenos virtuálního systému na jiné fyzické PC stačí pouze vytvořit virtuální vrstvu a připojit disk s přenášeným virtuálním OS.

Porovnáme-li výhodnost jednotlivých variant s ohledem na čas potřebný k odstranění případného výpadku systému, zvítězí virtuální OS, kde se doba výpadku pohybuje v řádu minut až hodin oproti fyzickému OS s možnou délkou výpadku den i více.

#### **Požadavek – zálohování dat**

Tento požadavek není stěžejní v otázce rozhodování mezi fyzickým a virtuálním OS. Virtualizace nám však nabízí oproti fyzickému OS možnost zálohování stávajícího stavu systému včetně RAM, což umožňuje vytváření bodů obnovy, ke kterým se můžeme kdykoliv vrátit.

#### **Požadavek – minimální finanční náklady**

Protože hostitelský OS i hostované systémy budou spadat pod licenci GNU GPL, náklady na realizaci se budou odvíjet primárně od ceny HW.

Náklady za HW se rovnají (v případě využívání virtualizačního nástroje VMware Server) pouze ceně RAM, resp. velikost potřebné RAM se odvíjí od počtu provozovaných virtuálních OS a jejich očekávanému vytížení.

Samozřejmě, pokud bychom chtěli co nejvíce optimalizovat HW pro provozování virtuálních OS, mohli bychom kromě nákupu RAM zvažovat i víceprocesorové základní desky, procesory s podporou virtualizace atp.

### **2.1.2 Navržení optimálního řešení**

Z rozboru požadavků na výsledné řešení jasně vyplynula výhodnost virtualizace. S ohledem na závěry teoretické části jsem zvolil pro realizaci virtuálního prostředí nástroj VMware Server, tzn. úplná virtualizace. (V době zahájení experimentu byl dostupný pouze VMware Server 1, v současné době se již přešlo na novější VMware Server 2.)

#### **Dostupné hardwarové prostředky**

**server10** – Intel® Pentium® 4 CPU 3 GHz, 1,5 GB RAM, 3× 300GB SATA disk,

**server20** – Intel® Core™ i7 CPU 920 @ 2,67 GHz, 7,7 GB RAM, 3× 300GB SATA disk,

**server15** – běžné PC.

#### **Hostitelský operační systém**

Jedná se o jeden z nejdůležitějších prvků, protože na hostitelském OS velkou měrou závisí výkon virtuálních serverů na něm hostovaných.

Paravirtualizace může být oproti úplné virtualizaci výhodnější díky systému přidělování zdrojů. Abychom tento rozdíl minimalizovali, musíme za hostitelský OS zvolit naprosto stabilní systém, který bude mít pro zabezpečení svého provozu minimální nároky na HW, tzn., sám bude provozovat nepatrné množství služeb. Hlavní jeho funkcí bude poskytování HW prostředků a řízení přístupu k nim.

Všechny tyto požadavky splňuje OS Linux. Po základní instalaci jádra<sup>1</sup> bude nabízet pouze servery SSH a NFS. Vhodné je instalovat i základní nástroje pro sledování síťového provozu, které ale nepoběží jako démoni. Z hlediska bezpečnosti bude tento OS dostupný pouze z vnitřní sítě a pouze pro uživatele s oprávněním superuživatele.

Pro tento účel jsem zvolil operační systém Debian. Kdybych tento problém řešil dnes, tzn. v roce 2009, zkusil bych porovnat výkon mnou zvoleného systému Debian s nově dostupným systémem Ubuntu<sup>2</sup> s úpravou jádra pro provozování virtuálních serverů.

### Virtuální servery

Základem budou dva virtuální servery. Nároky virtuálních OS na HW nesmí přesahovat možnosti daného HW (viz strana 19).

Fyzické servery budou obsahovat tzv. aktivní a neaktivní virtuální hosty. Neaktivní znamená vypnuté. Tyto neaktivní systémy budou aktivovány pouze v případě výpadku HW prostředků druhého fyzického serveru, aby nedošlo k přerušení chodu firmy. Vzhledem k odlišnému HW je potřeba optimalizovat HW nároky neaktivních kopií, aby systém po aktivaci nepožadoval např. více RAM, než kolik mu jí fyzický HW může nabídnout.

**server10** Hlavním virtuálním OS na tomto serveru bude server SAMBA (samba-server). Hlavní funkcí tohoto hosta bude provozování síťové domény Virtualbrana a poskytování dokumentů jednotlivým zaměstnancům firmy s ohledem na jejich pracovní zařazení, tzn., systém by měl řešit oprávnění v rámci přístupu k jednotlivým dokumentům. Neaktivním virtuálním OS bude www-server.

- Procesor – 1× 2,928 GHz,
- operační paměť – 512 MB RAM,
- místo na disku – do 500 MB.

---

<sup>1</sup>Linux nám umožňuje instalovat samotné jádro systému se základními nástroji, aniž bychom museli instalovat cokoli navíc, jak tomu je u OS typu Windows.

<sup>2</sup>Linuxový OS, vychází Debianu.

Datová úložiště pro samba-server se budou nacházet přímo na počítači server10 (v hostitelském systému) a budou připojena pomocí protokolu NFS do adresáře /mnt (ve virtualizovaném systému).

**server20** Hlavním virtuálním OS na tomto serveru bude server WWW (www-server). Mezi základní služby bude patřit web (Apache), databáze MySQL, PHP. Neaktivním virtuálním OS bude samba-server.

- Procesor – 2× 5,197 GHz,
- operační paměť – 3 600 MB RAM,
- místo na disku – do 500 MB.

HW nároky aktuálně odpovídají očekávanému vytížení virtuálních hostů.

Datová úložiště pro www-server se budou nacházet na počítači server20 a budou připojena jako virtuální disky do /var/www a /var/lib/mysql.

### Koncepce provozu

Původní myšlenka kalkulovala se čtyřmi virtuálními OS (SAMBA, FTP, MySQL a WWW server). Na jednom PC by byly vždy aktivní dva a dvě neaktivní kopie aktivních virtuálních OS z druhého PC. Aktivní data by byla umístěna na PC s aktivním virtuálním FTP a samba-serverem. Data by se v pevně stanovených časových intervalech synchronizovala pomocí Rsync na druhé PC, které by tak fungovalo jako neaktivní záloha. Jelikož by se data nacházela na jiném serveru, než na kterém by byly aktivní servery WWW a MySQL, připojila by se pomocí protokolu NFS.

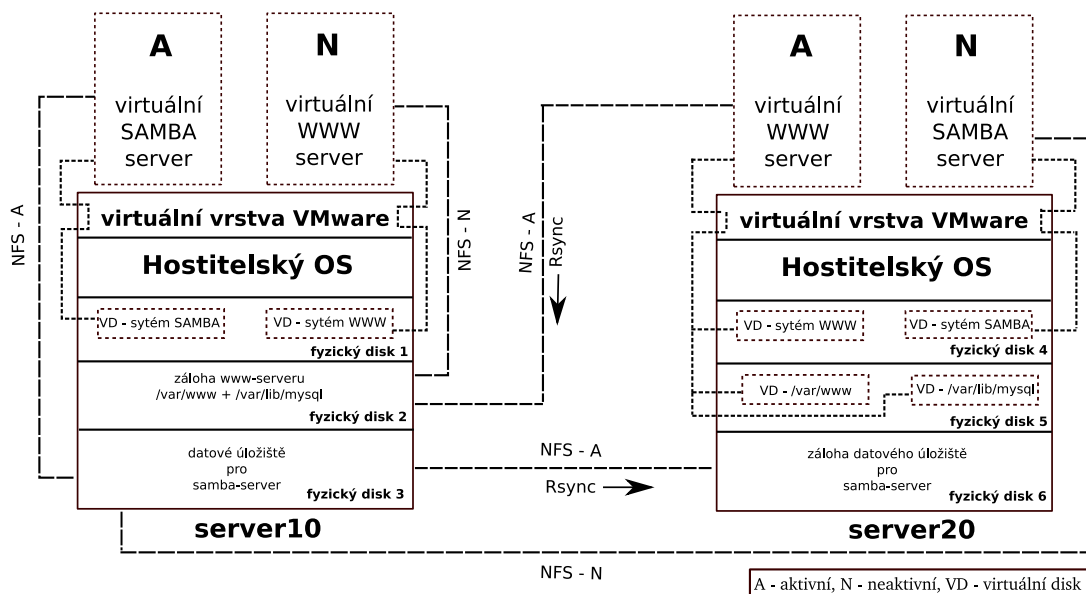
Toto řešení si kladlo za cíl minimalizovat riziko výpadku a ztráty dat. Pokud by totiž došlo k výpadku například www-serveru na dobu delší než 5 minut, byla by aktivována jeho kopie na druhém serveru a správci sítě byl zaslán varovný email informující o výpadku a potřebě aktivního zásahu.

Stejně tak, když by úplně zkolabovalo jedno z těchto dvou PC, došlo by na druhém funkčním PC k připojení datového úložiště a aktivaci neaktivních virtuálních OS.

Z výsledků dlouhodobého testování vyplynulo, že nepotřebujeme samostatný FTP server (stávající FTP protokol byl nahrazen VPN protokolem) a MySQL server se spojil s www-serverem. Tzn., ve výsledku zbyly pouze dva aktivní virtuální OS (www-server a samba-server) běžící na jednom fyzickém PC, využívající lokální disky a zalohující data na druhé fyzické PC.<sup>3</sup>

---

<sup>3</sup>Problematiku zálohování je řešena v sekci 2.4 na straně 27.



Obrázek 2.1: Grafické znázornění navrženého řešení.

Obrázek 2.1 znázorňuje finální koncepci fungování virtuálních serverů. Z HW prostředí jsou zobrazeny pouze datová úložiště (fyzický disk 1 až 6).

Označení *NFS - A* znamená aktivní připojení datového úložiště pomocí NFS. Popis *NFS - N* označuje záložní připojení, které je aktuálně neaktivní. Aktivuje se až v okamžiku výpadku.

Šipka u popisku *Rsync* vyznačuje směr toku dat. Vždy se jedná o jednosměrný systém zálohování. *Rsync* směřující ze server20 zálohuje data uložená na virtuálních discích (virtuální disky nejsou připojeny k serveru20, proto zabezpečuje zálohování virtuální host). *Rsync* ze server10 na server20 pak aktivuje přímo server10.

Přerušovaná čára od virtuálních disků přes zónu virtuální vrstvy oznamuje, že jsou disky připojeny prostřednictvím virtuální vrstvy.

## 2.2 Instalace a konfigurace

### 2.2.1 Hostitelský operační systém

Instalace základního jádra systému byla provedena metodou *net-install* (instalace ze sítě). Jedinými doplňky navíc byly instalace *SSH*, *nfs-server*, *kernel-headers* pro dané aktuální jádro fyzického OS, *make* a kompilátor *g++*.

Instalace virtualizačního nástroje vyžaduje pouze spuštění připraveného scriptu, který zabezpečí konfiguraci a následnou kompilaci včetně instalace.

Ze strany uživatele bylo potřeba potvrdit, případně upřesnit otázky týkající se umístění souborů s nápovědou, umístění virtuálních systémů, konfigurace virtuálních a nevirtuálních sítí a schválit licenční podmínky.

Celá instalace byla realizována prostřednictvím terminálu bez nutnosti instalovat jakoukoliv formu grafického prostředí<sup>4</sup>.

### 2.2.2 Virtuální operační systém

Instalaci virtuálních OS lze provádět vzdáleně. Pro instalaci bylo potřeba nainstalovat tzv. Console pro vzdálenou správu VMware Server 1<sup>5</sup>.

Pro přihlášení ke vzdálenému serveru Console vyžadovala adresu IP hostitelského OS, login a heslo superuživatele.

Nástroj VMware Console obsahuje základní menu a dva sloupce koncipované v poměru 30:70. Menší sloupec je určen pro zobrazení seznamu existujících virtuálních hostů. Větší sloupec pak dle vybraných záložek zobrazuje virtuální HW nebo grafický výstup<sup>6</sup> vzdáleného virtuálního OS.

Instalace nového hosta se zahájí pomocí položky hlavního menu přidat nový virtuální OS. Nejprve je potřeba vybrat z nabídky typ systému. Nabídka neobsahuje všechny OS, proto je potřeba zvolit nejbližší možný<sup>7</sup>.

Následuje přidání disků. Výběr umožňuje volbu mezi existující fyzickou partition nebo virtuálním SCSI diskem. Pro jádro OS je vhodný 4GB virtuální SCSI disk s parametrem postupné alokace místa. Postupná alokace funguje na principu nastavení maximální velikosti disku a blokové velikosti. Pokud velikost systému dosáhne velikosti zvoleného bloku, např. 2 GB, virtualizace si na fyzickém disku alokuje další blok, v tomto případě 2 GB prostoru. Výhodou postupné alokace je úspora místa na fyzickém disku aniž bychom omezovali virtuální disk. Například lze umístit 150GB virtuální disk na 20GB fyzický, pokud tedy virtuální disk neobsahuje více než 20 GB dat.

V případě virtuálního www-serveru byly vybrány virtuální SCSI disky s parametrem úplné alokace místa na fyzickém disku.

Následně je potřeba specifikovat instalační médium. Lze si vybrat mezi fyzickou a virtuální mechanikou. Pro vzdálenou instalaci hosta je nejvhodnější virtuální mechanika.

Samotná instalace virtuálního systému probíhá identicky s instalací nevirtuálního OS a byla provedena dle návrhu v sekci 2.1.2 na straně 20.

---

<sup>4</sup>Grafické prostředí na serveru je zbytečné – vytěžuje HW prostředky a zabírá místo na fyzickém disku.

<sup>5</sup>VMware Console podporuje instalaci v OS typu Linux i Windows.

<sup>6</sup>Stejný výstup, jako v případě připojení monitoru k danému serveru.

<sup>7</sup>Nejbližším systému Debian z nabízených byl linuxový OS s jádrem Ubuntu.



## 2.3 Testování a optimalizace

Testování a optimalizaci rozdělím do dvou částí. První částí bude ladění před uvedením řešení do ostrého provozu a druhou pak testování ostrým provozem.

Vzhledem k tématu této práce nebudu popisovat konfiguraci a testování aplikací, které přímo nesouvisí s problematikou virtualizace.

### 2.3.1 Ladění před uvedením do ostrého provozu

Základem této fáze ladění a testování byla konfigurace samba-serveru, řešení problematiky připojení vzdálených disků pomocí NFS protokolu a ladění automatických procedur pro minimalizaci případných výpadků.

#### Největší slabiny virtuálních OS

**Nestabilní hostitelský OS** Dojde-li k pádu či zamrznutí hostitelského OS, vytrácí se hlavní výhoda virtualizace. Tou je vzdálený přístup a správa virtuálních hostů. Aby toto nebylo ohroženo, je třeba zajistit, aby hostitel neprovozoval žádné nebezpečné nebo hardwarově náročné služby. Tento bod byl vyřešen již při instalaci hostitelského systému, protože bylo instalováno pouze jádro s nejnужnějšími nástroji.

**Nedostatek místa na fyzickém disku** Tento problém může nastat pouze v případě, kdy využíváme virtuální disky s postupnou alokací paměti. Pokud máme k dispozici fyzický disk o velikosti 20 GB a na něm provozujeme 50GB virtuální disk s postupnou alokací paměti, při požadavku o rozšíření nad 20 GB dojde k pádu virtuálního hosta.

Preventivním opatřením může být script aktivovaný v pravidelných časových intervalech pomocí démona cron. Script má za úkol sledovat velikost místa na fyzických discích a pokud se volné místo na disku sníží pod určitou hranici, obdrží správce systému upozornění prostřednictvím e-mailu.

**Chybně nakonfigurovaný virtuální HW** Přidělením více RAM virtuálním OS, než kolik má k dispozici fyzický server a následnou aktivací hostů může nastat situace, kdy hostované systémy plně vytíží dostupnou fyzickou RAM a hostitelský systém je nucen využívat paměť typu swap<sup>8</sup>, což se ve výsledku projeví rapidním snížením výkonu

---

<sup>8</sup>„Swapovací oddíl (případně swapovací soubor), úložiště dat používané pro realizaci virtuální paměti.“ [20]

všech hostů a samotného hostitelského systému. S ohledem na reakční dobu by se taková situace dala klasifikovat jako dočasný výpadek.

**Nedostatečná záloha** Problém nedostatečného zálohování konfiguračních souborů virtuálních disků je zaznamenán v okamžiku, kdy se tyto soubory poškodí. Virtuální disky jsou klasické soubory, kde na každý takový případ jeden konfigurační obsahující informace o rozložení virtuálního souboru. Bez konfiguračního souboru tedy není virtuální systém schopen takový disk správně načíst a data na takovém disku pak mohou být nenávratně ztracena.

**Výpadek** Nejnáchylnějšími k přetížení či pádu jsou virtuální OS. Důvodem je fakt, že provozují mnohem více aplikací než hostitelský systém. Preventivním opatřením pro tento bod je neaktivní kopie aktivního virtuálního hosta. Fyzický server pak stejně jako v případě sledování místa na discích zkouší metodou ping, zda aktivní virtuální host odpovídá. Pokud tomu tak není, dojde po 5 minutách k pokusu o restart daného hosta a pokud ani toto po dalších 5 minutách nepomůže, dojde k pokusu o vypnutí problémového virtuálního OS a aktivaci jeho doposud neaktivní, avšak permanentně připravené kopie. Správci pak je zaslána e-mailová zpráva o výskytu problému, který nelze vyřešit automaticky.

### 2.3.2 Ladění v ostrém provozu

Předchozí kapitola byla zaměřena na obecně známé problémy spojené s virtuálními OS. V následující kapitole se zaměřím na problémy, které jsme zaznamenali až po uvedení celého systému do ostrého provozu.

V rámci popisu problému jsem zavedl strukturu, kdy daný problém nejdříve pojmenuji, následně popíši, nastíním možné varianty řešení a doporučím variantu, která se mi nejvíce osvědčila.

#### Problém – pomalý přenos dat

Data byla uložena v rámci virtuálního disku na server10. Virtuální OS byl umístěn na server20.<sup>9</sup> Datové úložiště pak bylo nabízeno fyzickému server20 prostřednictvím NFS.

**Popis problému 1** *Rychlost přenosu dat mezi virtuálním OS a fyzickým úložištěm připojeným pomocí NFS dosahovala nejvýše 1/5 maximální reálné<sup>10</sup> přenosové rychlosti.*

---

<sup>9</sup>Systémy server10 a server20 jsou popsány v sekci 2.1.2 na straně 19.

<sup>10</sup>Reálná rychlost odpovídá rychlosti přenosu dat mezi dvěma fyzickými servery.

**Varianty řešení:**

1. Přesun dat z virtuálního disku na fyzickou partition.
2. Přesun datového úložiště z fyzického server10 na server20, čímž jsme vyřadili NFS.

**Doporučené řešení problému 1** *Výsledkem pokusů bylo zjištění, že rychlost není ovlivněna využíváním virtuálních SCSI disků, ale NFS protokolem. Problém byl vyřešen upuštěním od využíváním NFS.*

**Problém – prázdný virtuální disk s postupnou alokací zabírá fyzické místo**

Problém se projevoval nejčastěji po operaci dočasné zálohy většího množství dat na virtuálním disku s aktivním systémem postupné alokace fyzické paměti.

**Popis problému 2** *Virtuální disk obsahuje 50 GB dat a na fyzickém disku alokuje 50 GB dat. Maximální možná velikost virtuálního disku činí 400 GB. Na virtuální disk dočasně zazálohujeme 300 GB a poté tato dočasná data z disku odstraníme. Výsledkem bude virtuální disk obsahující 50 GB dat, avšak zbytečně alokující 350 GB na fyzickém disku.*

**Varianty řešení:**

1. Vytvoření nového virtuálního disku, překopírování dat a odstranění původního virtuálního disku.
2. VMware Server 1 oproti jeho nástupci podporoval funkci defragmentace disku, která zabezpečila opětovné zmenšení disku.
3. Rozšíření kapacit fyzických disků a úplná alokace místa ihned po vytvoření virtuálního disku.

**Doporučené řešení problému 2** *První a druhá varianta vyžaduje permanentní kontrolu systému ze strany správce serveru. Pokud by totiž virtuální OS využíval služeb více virtuálních disků, jejichž maximální velikost by přesahovala kapacity fyzických disků, mohlo by dojít k zhroucení systému. Třetí varianta je nejbezpečnější cestou a od VMware Serveru 2 nejspíše jedinou vhodnou volbou, protože VMware Server 2 již nepodporuje připojení fyzické partition, ale pouze virtuálních disků.*

**Problém – systémový čas hosta se rozcházel hostitelovým**

**Popis problému 3** *Informační systém provozovaný v rámci virtuálního hosta začal k zrealizovaným transakcím vkládat chybný časový údaj. Příčinou byl chybný čas, kterým disponoval daný virtuální OS.*

**Varianty řešení:**

1. Pravidelná aktualizace času pomocí cronu.
2. Instalace VMware Tools s následným zapnutím podpory pro synchronizaci času s fyzickým OS v rámci konfigurace virtuálního hosta.

**Doporučené řešení problému 3** *Dokud jsme neobjevili druhé řešení, museli jsme se spokojit s aktualizací času v dvouminutových intervalech – viz první varianta.*

## 2.4 Systém zálohování firemních dat

Společnost Ebrana, s. r. o., se zaměřuje na vývoj softwarových produktů, a proto musí zálohovat nejenom administrativní data (faktury, smlouvy a podklady k zakázkám), ale i vyvíjený software (PHP skripty a databáze).

Celkem je zálohováno cca 300 GB malých souborů většinou nepřesahujících velikost 500 KB. Aby bylo možno efektivně zálohovat takové množství dat, je potřeba využít zálohování s podporou komprese zálohovaných souborů.

### 2.4.1 Hlavní požadavky

Mezi hlavní požadavky kladených na systém zálohování patří:

1. úplná záloha operačního systému,
2. minimální/maximální doba zálohování 24 h/1 měsíc,
3. minimální doba obnovy v případě poškození disku s aktivními daty,
4. úplná automatizace s permanentním informováním správce systému o průběhu záloh.

## 2.4.2 Nástroje

### Snapshot

- implicitní zálohovací mechanismus produktu VMware Server,
- záloha je obrazem systému v okamžiku zahájení zálohy,
- umožňuje zálohování systému za běhu,
- snapshot lze chápat ve smyslu bodu obnovy.

**Implementace** Tento nástroj se využívá hlavně v okamžiku testování či upgrade systému. Pokud dojde k poškození systému, ať už z důvodu zavirování či chybné instalace, vyvolání snapshotu umožní návrat do stavu, kdy byl ještě systém plně stabilní.

### Bacula

Profesionální zálohovací nástroj vedený pod GNU GPL licenci. Je tvořen skupinou programů umožňujících zálohovat a obnovovat data v rámci celé sítě. Nejvýstižnějším přirovnáním k fungování je metoda klient/server.

### Základní stavební prvky

- File Daemon – Je démon (služba) běžící na klientském počítači, který je zálohován. Rovněž bývá zmiňován jako File services, někdy jako Client service, nebo FD. [15]
- Storage Daemon – bývá někdy označován zkratkou SD je program, který zapisuje data a jejich atributy na ukládací svazek (obyčejně páska, nebo disk). [15]
- Director – Hlavní serverový démon Baculy, který plánuje a řídí všechny operace Baculy. Director bývá často uváděn jako zkratka DIR. [15]
- Console – Programové rozhraní od Directoru umožňující správci či uživateli ovládat Baculu. [15]

### Implementace

1. Instalace Storage démonů na PC vyhrazená pro ukládání zálohovaných dat.
2. Instalace File démonů na PC obsahující data určená k zálohování.
3. Instalace Director a Console na jedno vybrané PC. Toto pak bude centrálním mozkiem celého zálohovacího systému.

4. Konfigurace Directoru (zde se nadefinuje, na jakých PC se nachází File démony a Storage démony včetně adresářů určených pro uchování zálohovaných dat a adresáři obsahujících data určená k zálohování).
5. Buď pomocí Console nebo pomocí konfiguračního scriptu se nastaví, v jakých intervalech se mají zálohovat data z jednotlivých PC na vybrané zálohovací servery včetně specifikace typu zálohování, úrovně komprimace dat a délky existence záloh.

### **Rsync**

„Počítačový program původně pro unixové systémy, který synchronizuje soubory a adresáře mezi různými místy za použití co nejmenšího přenosu dat. Tam, kde je to možné, přenáší pouze rozdíly mezi soubory.“ [16]

**Implementace** V 24hodinových intervalech synchronizuje aktivní data se zálohovanými.

### **2.4.3 Navržený systém zálohování**

Aby bylo možné uchovávat měsíc (původní požadavek byl 2 měsíce) staré zálohy, je potřeba zálohovaná data komprimovat. Systém Bacula podporuje diferencíální zálohování včetně komprimace dat. Nevýhodou jsou vyšší časové nároky na obnovu. Abychom tento problém eliminovali, využíváme k zálohování kromě systému Bacula i Rsync.

Data zálohovaná pomocí metody Rsync využíváme v případech, kdy zaznamenáme ztrátu do 24 hodin od poslední zálohy. Tento typ zálohy je přístupný všem a v případě rozsáhlejší ztráty dat nám umožňuje disk se zálohou kamkoliv připojit jako celek, bez nutnosti prodlužování výpadku kopírováním nebo dekomprimací.

Data zálohovaná systémem Bacula lze aktuálně získat pouze s přispěním správce sítě. Ten pomocí Console zajistí obnovu vyžádaných souborů z požadovaného období. Tato operace je časově náročnější už kvůli nutné intervenci správce sítě.

Celý systém zálohování a obnovy byl otestován a následně i optimalizován na základě úplného vymazání fyzického i virtuálních OS a disků s daty.

# Závěr

Cílem práce bylo popsat současnou situaci v oblasti virtualizačních technologií. V praktické části pak navrhnout a realizovat optimalizaci architektury firemních serverů pro menší firmu se zaměřením na vývoj webových produktů, za využití vhodných virtualizačních nástrojů. Součástí optimalizace bylo i vyřešení problematiky zálohování a případné obnovy dat. Tento cíl se podařilo úspěšně splnit.

Po sumarizaci potřeb firmy jsem navrhl optimální řešení. Vzhledem k požadavkům na fungování a možnostem společnosti Ebrána, s. r. o., jsem jako virtualizační nástroj zvolil VMware Server. Toto řešení se po téměř dvouletém provozu ukázalo jako vhodné a splnilo všechna očekávání.

Problémy, které musely být řešeny, bych rozdělil do dvou částí. Problémy týkající se samotné virtualizace a problémy, které s virtualizací nesouvisí, avšak jejich řešení nám může vhodně aplikovaná virtualizace značně usnadnit.

První skupinu problémů jsem popsal v sekci 2.3.2 *Ladění v ostrém provozu* na straně 25. Do druhé skupiny pak náleží situace nenávratného poškození partition tabulek na systému server20, poškození konfiguračních souborů virtuálních disků, výpadek zálohování pro nedostatek místa na záložních discích. Po vyřešení každého z problémů vždy přibylo nové preventivní opatření, případně optimalizace stávající architektury.

Příkladem může být dvojitý systém zálohování<sup>11</sup> nebo koncepce aktivních a neaktivních virtuálních serverů (včetně systému automatické aktivace) rozmístěných na oddělených fyzických strojích<sup>12</sup>.

Uvedený postup se ukázal jako velmi výhodný a lze ho doporučit u společností podobného charakteru.

---

<sup>11</sup>Systém je popsán v sekci 2.4 *Systém zálohování firemních dat* na straně 27.

<sup>12</sup>Metodika je popsána v sekci 2.1.2 *Koncepce provozu* na straně 21.

## Seznam použitých zkratek

**API** – application programming interface

**CMS** – content management system

**CPU** – central processing unit

**FD** – file daemon

**HW** – hardware

**IS** – informační systém

**NFS** – network file system

**OS** – operační systém

**PC** – personal computer

**RAM** – random-access memory

**SAN** – storage area network

**SD** – storage daemon.

**SSH** – secure shell

**SW** – software

**VPN** – virtual private network

**VPS** – virtual private server

## Seznam pojmů

**chroot** – Unixový příkaz, který slouží ke změně kořenového adresáře pro daný proces a jeho potomky.

**cron** – Softwarový démon, který v operačních systémech na bázi Unixu automatizovaně spouští v určitý čas nějaký příkaz.

**daemon nebo démon** – Program běžící na serveru a čekající na požadavky, které obsluhuje. Démon běží neustále.

**desktop** – Klasický stolní počítač.



**emulátor** – Druh software umožňující běh počítačových programů na jiné platformě (architektuře, operačním systému), než pro kterou byly původně vytvořeny a kterou samy od sebe podporují.

**hypervisor** – V případě virtuálního nástroje Xen zabezpečuje hypervisor obhospodáření paměti a obsluhu I/O operací, které zprostředkovává hostům.

**kernel-headers** – Jedná se o zdrojové soubory jádra operačního systému Linux.

**open-source** – Počítačový software s otevřeným zdrojovým kódem.

**protokol NFS** – Internetový protokol pro vzdálený přístup k souborům přes počítačovou síť.

**protokol SSH** – Protokol pro bezpečný vzdálený přístup.

**protokol VPN** – Umožňuje šifrované propojení několika počítačů na různých místech Internetu do jediné virtuální počítačové sítě.

**superuživatel** – Též superuser či root je v operačních systémech rodiny Unix uživatel s nejvyšším oprávněním.

**webhosting** – Webhosting je pronájem prostoru pro webové stránky na cizím serveru.

# Literatura

- [1] L. Matyska. Virtualizace výpočetního prostředí. *Zpravodaj ÚVT MU*. 2006, roč. XVII, č. 2, s. 9-11. [cit.2009-06-07]. ISSN 1212-0901.  
Dostupný též z WWW: <<http://www.ics.muni.cz/zpravodaj/articles/540.html>>
- [2] *Xen* [online]. 2008 [cit.2008-09-21].  
Dostupný z WWW: <<http://en.wikipedia.org/wiki/Xen>>
- [3] Rami Rosen. *Introduction to the Xen Virtual Machine* [online]. 2005, September [cit.2009-06-06].  
Dostupný z WWW: <<http://www.linuxjournal.com/article/8540>>
- [4] Miroslav Suchý. *Úvod do virtualizace pomocí XENu* [online]. 2007, March [cit.2009-06-06].  
Dostupný z WWW: <<http://www.root.cz/clanky/uvod-do-virtualizace-pomoci-xenu>>
- [5] Kamil Dedecius. *(Para)virtualizace pro každého – Xen* [online]. 2006, August [cit.2009-06-06].  
Dostupný z WWW:  
<<http://www.linuxexpres.cz/praxe/para-virtualizace-pro-kazdeho-xen>>
- [6] *VMware.Server* [online]. 2008 [cit.2008-09-21].  
Dostupný z WWW: <[http://en.wikipedia.org/wiki/VMware\\_Server](http://en.wikipedia.org/wiki/VMware_Server)>
- [7] *VMware* [online]. 2009, July [cit.2009-08-02].  
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/VMware>>
- [8] *QEMU* [online]. 2008 [cit.2008-09-21].  
Dostupný z WWW: <<http://en.wikipedia.org/wiki/QEMU?>>
- [9] Petr Krčmář. *QEMU - qemulátor* [online]. 2005, April [cit.2009-06-06].  
Dostupný z WWW: <<http://www.root.cz/clanky/qemu-qemulator>>
- [10] *VirtualBox* [online]. 2008 [cit.2008-09-21].  
Dostupný z WWW: <<http://en.wikipedia.org/wiki/VirtualBox>>

- [11] Petr Krčmář. *VirtualBox: čtvrté kolo u virtualizačního vozu* [online]. 2007, April [cit.2009-06-06]. Dostupný z WWW: <<http://www.root.cz/clanky/virtualbox-ctvrte-kolo-u-virtualizacniho-vozu>>
- [12] *Linux-VServer* [online]. 2008 [cit.2008-09-21]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/Linux-VServer>>
- [13] Jaroslav Tomeček. *Virtualizace na úrovni jádra operačního systému* [online]. 2007, July [cit.2009-06-07]. Dostupný z WWW: <<http://www.abclinuxu.cz/clanky/system/virtualizace-na-urovni-jadra-operacniho-systemu>>
- [14] Jaroslav Tomeček. *Linux-VServer - instalace a spuštění* [online]. 2007, July [cit.2009-06-07]. Dostupný z WWW: <<http://www.abclinuxu.cz/clanky/system/linux-vserver-instalace-a-spusteni>>
- [15] *Terminologie* [online]. 2008, December [cit.2009-08-01]. Dostupný z WWW: <<http://wiki.bacula.org/doku.php?id=terminologie>>
- [16] *Rsync* [online]. 2008, October [cit.2009-08-01]. Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Rsync>>
- [17] Antonín Vítěček. *Virtualizace* [online]. 2008 [cit.2009-06-07]. Dostupný z WWW: <<http://www.fi.muni.cz/~kas/p090/referaty/2008-jaro/ct/virtualizace.html>>
- [18] *Virtualizace* [online]. 2009, July [cit.2009-08-02]. Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Virtualizace>>
- [19] *TechComparison* [online]. 2008 [cit.2009-06-07]. Dostupný z WWW: <<http://virt.kernelnewbies.org/TechComparison>>
- [20] *Swap* [online]. 2008, October [cit.2009-06-07]. Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Swap>>