

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačního inženýrství**



**Bakalářská práce**

**Konfigurace síťových a serverových služeb na  
embedded zařízení**

**Cyril Fessler**

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Cyril Fessler

Informatika

Název práce

**Konfigurace síťových a serverových služeb na embedded zařízení**

Název anglicky

**Configuration of network and server-side services on embedded device**

---

### Cíle práce

Hlavním cílem práce je konfigurace síťových a serverových služeb na malém embedded zařízení s operačním systémem GNU/Linux. Vedlejším cílem práce je tento server optimalizovat pro co nejvyšší možný výkon se zachováním nejvyšší možné bezpečnosti.

### Metodika

Metodika této bakalářské práce je založena na studiu dostupné literatury, dokumentace a internetových zdrojů, jejich vytřídění a zpracování a na navrhnutí a realizaci výstupu – tj. bezpečného serveru pro malé firmy a domácnosti.

V teoretické části budou zahrnuty a popsány všechny použité technologie a serverové aplikace (FTP server, SAMBA server, Web server, SFTP a SSH server, RAID pole, P2P klient, služby síťové infrastruktury a další).

V praktické části budou tyto technologie a aplikace konfigurovány na skutečném embedded zařízení.

Pro vedlejší cíl bude zařízení testováno s různými konfiguracemi aplikací a tyto testy budou následně vyhodnoceny.

## Doporučený rozsah práce

30-40 stran

## Klíčová slova

linux, ARM, server, bezpečnost, soukromí

---

## Doporučené zdroje informací

BRANDEJS, M. *UNIX – LINUX: praktický průvodce*. Praha: Grada, 1996. ISBN 80-7169-170-4. COLLIER-BROWN, D. – ECKSTEIN, R. – TS, J. *Samba Linux jako server v sítích s Windows*. Brno: CP Books, 2005. ISBN 80-251-0649-7.

HEIN, T R. – NEMETH, E. – SNYDER, G. *Linux : kompletní příručka administrátora*. Brno: ComputerPress, 2004. ISBN 80-7226-919-4.

JELÍNEK, L. *Jádro systému Linux : kompletní průvodce programátora*. Brno: ComputerPress, 2008. ISBN 978-80-251-2084-2.

*Linux : dokumentační projekt : nejpodrobnější dokumentace operačního systému Linux v češtině*. Praha:ComputerPress, 1998. ISBN 80-7226-114-2.

MATHEW, N. – STONES, R. *Linux : začínáme programovat*. Praha: ComputerPress, 2000. ISBN 80-7226-307-2.

RANDUS, J A. – SATRAPA, P. *Linux : internet server*. Praha: Neokortex, 1996. ISBN 80-902230-0-1.

---

## Předběžný termín obhajoby

2015/16 LS – PEF

## Vedoucí práce

Ing. Marek Pícka, Ph.D.

## Garantující pracoviště

Katedra informačního inženýrství

---

Elektronicky schváleno dne 20. 2. 2016

**Ing. Mar n Pelikán, Ph.D.**

Vedoucí katedry

---

Elektronicky schváleno dne 20. 2. 2016

**Ing. Mar n Pelikán, Ph.D.**

Děkan

V Praze dne 13. 03. 2016

---

### **Čestné prohlášení**

Prohlašuji, že svou bakalářskou práci "Konfigurace síťových a serverových služeb na embedded zařízení" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 14. 3. 2016 \_\_\_\_\_

## **Poděkování**

Rád bych touto cestou poděkoval vedoucímu práce Ing. Marku Píckovi, Ph.D. za pomoc a poskytování rad v průběhu tvorby této práce a Janě Andrejskové za korekci textu.

# Konfigurace síťových a serverových služeb na embedded zařízení

## Souhrn

Tato bakalářská práce je reakcí na přetrvávající snahu internetových útočníků získávat data z takzvaných *cloudových* úložišť. V poslední době přichází více a více zpráv o průnicích do těchto systémů, a tak se tato tematika postupně dostává do vědomí běžných uživatelů, kterým začíná být zřejmé, že data v tomto úložišti jsou ve skutečnosti data na neurčitém a neznámém serveru, který může být ať už záměrně nebo nezáměrně provozovatelem poskytnut třetí, nežádoucí, osobě. V této práci je nalezena alternativní možnost – použití vlastního serveru, nad kterým má majitel plnou kontrolu, poskytuje stejné nebo lepší služby a je dokonce cenově dostupnější. Serverový software je v této práci nakonfigurován a otestován zátěžovými testy, které jsou poté vyhodnoceny.

**Klíčová slova:** Linux, ARM, server, bezpečnost, soukromí, zabezpečení

## Configuration of network and server-side services on embedded device

## Summary

This bachelor thesis is a reaction to the persistent efforts of Internet attackers to obtain data from the so-called *cloud*. More and more reports about cracking to these infrastructures are coming recently, so this information is getting into the consciousness of ordinary users, which becomes obvious that the data in this *cloud* are actually data on indeterminate and unknown server, which may be either intentionally or unintentionally provided to third, undesirable person. In this work is found an alternative option – using own server, over which the owner has full control and which provides the same or better services and is even more affordable. The server software in this work is configured and stress-tested. These test are then evaluated.

**Keywords:** Linux, ARM, server, security, privacy, protection

# Obsah

<b>1 Úvod.....</b>	<b>1</b>
<b>2 Cíl práce.....</b>	<b>2</b>
<b>3 Metodika .....</b>	<b>2</b>
<b>4 Teoretická východiska .....</b>	<b>3</b>
4.1 Rozdíl mezi architekturami x86 a ARM .....	3
4.1.1 CISC.....	3
4.1.2 RISC.....	3
4.1.3 Rozdíl v energetické účinnosti.....	3
4.2 ARM počítače .....	4
4.2.1 RaspberryPi.....	4
4.2.2 Orange Pi .....	5
4.2.3 BananaPi .....	5
1.1.1. Odroid C1 .....	7
4.3 Speciální distribuce pro desky .....	8
4.3.1 Raspbian.....	8
4.3.2 Armbian .....	9
4.4 Aplikace na straně serveru .....	9
4.4.1 FTP server.....	9
4.4.2 SSH server .....	10
4.4.3 Samba.....	11
4.4.4 LAMP .....	11
4.4.5 VLC server.....	13
4.4.6 P2P .....	14
4.5 RAID .....	14
4.5.1 Co je to RAID .....	14
4.5.2 Běžné úrovně RAID [48] [50] .....	15
4.6 Vstupně-výstupní plánovač.....	17
4.6.1 CFQ.....	17
4.6.2 NOOP.....	17
4.6.3 DEADLINE .....	17
4.7 Souborové systémy .....	18
4.7.1 Ext2 – Second extendedfilesystem .....	18
4.7.2 Ext4 - Fourthextendedfilesystem .....	18
4.7.3 F2FS - Flash-FriendlyFileSystem.....	18
4.7.4 NTFS.....	18

<b>5 Vlastní práce .....</b>	<b>19</b>
5.1 Výběr vhodného jednodeskového počítače pro serverové aplikace .....	19
5.1.1 Porovnání vlastností a vhodnost pro nasazení .....	20
5.1.2 Finální výběr .....	22
5.2 Příprava a instalace operačního systému.....	22
5.3 Konfigurace serverových aplikací.....	24
5.3.1 Konfigurace základního systému.....	24
5.3.2 Konfigurace SSH serveru .....	27
5.3.3 Konfigurace pro FTPd serveru .....	28
5.3.4 Konfigurace RAID pole.....	29
5.3.5 Konfigurace webového serveru – Lighttpd, PHP, SQLite a Wordpress ..	32
5.3.6 Konfigurace Samba serveru.....	33
5.3.7 Konfigurace P2P klienta .....	35
5.3.8 Konfigurace VLC .....	35
5.4 Srovnání s dalšími možnostmi pořízení serveru .....	37
<b>6 Závěr.....</b>	<b>38</b>
<b>7 Bibliografie .....</b>	<b>39</b>
<b>8 Přílohy .....</b>	<b>45</b>



## **Seznam obrázků**

Obrázek 1 - Práce FTP serveru .....	18
Obrázek 2 - Zjednodušený diagram práce webového serveru s dynamickou webovou stránkou.....	21
Obrázek 3 - RAID pole z 8 úložišť .....	39
Obrázek 4 - Test závislosti souborového systému a plánovače na výkon RAID pole .....	40
Obrázek 5 - Hello world, Wordpress .....	41
Obrázek 6 - Přihlašovací výzva operačního systému Microsoft Windows .....	42
Obrázek 7 - VLC, viditelné heslo a port .....	43
Obrázek 8 - VLC, neviditelné heslo a port .....	43
Obrázek 9 - Vzdálené ovládání VLC pomocí aplikace VLC Remote .....	44

## **Seznam tabulek**

Tabulka 1 - Protokoly, které může SSH nahradit pro lepší zabezpečení.....	18
---	----

# 1 Úvod

V poslední době se zvýšila informovanost běžných uživatelů internetu o nedostatečném zabezpečení internetových (takzvaných *cloudových*) úložišť a služeb a o velmi častých útocích na tuto infrastrukturu [1][2]. Právě proto jednotliví uživatelé i firmy vyhledávají alternativy k tomuto typu infrastruktury.

Náklady na alternativní možnosti mohou být pro menší firmy či domácnosti nižší, než náklady na pronájem úložišť na cizí infrastrukturu.

Uživatelé se chtějí chránit jak proti útočníkům, kteří by jejich data chtěli zneužít pro vlastní potřeby nebo s nimi uživatele vydírat, tak také proti stále častěji odhalovanému naslouchání provozu některými státními organizacemi [3] [4] [5] [6].

Kvůli špatným bezpečnostním praktikám některých dodavatelů jsou data uživatelů v nebezpečí a uživatelé tak hledají jiné způsoby, jak uchránit svá data před potencionálními útoky a zneužitím.

Tato práce je věnována nalezení a zprovoznění vhodné alternativy k používání služeb na cizí infrastrukturu.

## **2 Cíl práce**

Cílem této práce je konfigurace serveru na jednodeskovém počítači s operačním systémem Linux.

K dosažení tohoto cíle je zapotřebí najít vhodné zařízení, které lze využít jako server, a nalézt serverový software, který bude s tímto zařízením efektivně pracovat a bude splňovat bezpečnostní zásady. Tento software poté bude na zařízení instalován, konfigurován a otestován.

Tím dojde k vytvoření takového balíku serverového softwaru, který nabídne nejvyšší možnou bezpečnost s ohledem na jeho výkonovou náročnost.

Výstupem této práce jsou konfigurační soubory k použitému softwaru, výsledky výkonových testů a porovnání vlastností jednotlivých jednodeskových počítačů.

## **3 Metodika**

V první části se práce věnuje představení jednodeskových počítačů dostupných na českém trhu, možností konfigurací úložišť a softwaru, který je použitý v druhé části této práce.

Mezi tento software patří především serverové programy jako je FTP server, webový server, systém řízení báze dat, PHP, SSH/SFTP server, multimediální server, Samba a P2P klient.

Druhá část práce se nejprve věnuje výběru nejvhodnějšího jednodeskového počítače podle požadovaných kritérií, jeho přípravě a instalaci základního systému. Poté je nainstalován serverový software, který je představen v první části práce, a je nakonfigurováno diskové pole pomocí technologií dostupných v jádře Linux.

Relevantní služby jednodeskového počítače po konfiguraci podstoupí výkonové a zatěžkávací testy, které jsou vyhodnoceny.

## 4 Teoretická východiska

### 4.1 Rozdíl mezi architekturami x86 a ARM

Architektura x86 používá CISC (Complex instruction set computing), zatímco architektura ARM používá RISC (Reduced Instruction Set Computer).

#### 4.1.1 CISC

Tuto instrukční sadu používají například procesory od společností Intel a AMD. V této sadě je uloženo velké množství komplexních (*velkých*) instrukcí. Tím pádem je instrukční sada pomalejší při zpracování jedné instrukce, právě proto, že je instrukce komplexní. Na druhou stranu potřebuje celkově méně těchto instrukcí zpracovat pro komplexní úlohy.

Toto tvrzení se mimo jiné opírá o filozofii, že hardware je vždy rychlejší než software.  
[7] [8]

#### 4.1.2 RISC

V této sadě je uloženo menší množství malých a primitivnějších instrukcí. Ty jsou zpracovány rychleji, ale na komplexní úlohy je jich zapotřebí více.

Toto tvrzení se mimo jiné opírá o filozofii, že komplexní instrukce obsažené v CISC nejsou mnohdy využívány.[7] [8]

#### 4.1.3 Rozdíl v energetické účinnosti

Často jsou ARM procesory označovány za energeticky účinnější, zato ale méně výkonné. Naopak architektura x86 je označována za výkonnější, ale za méně energeticky účinnou.

Vývoj obou architektur ale stále probíhá, a zatímco se x86 procesory stávají energeticky velmi úsporné a efektivní (Intel Atom), ARM procesory zvyšují výkon (Cortex A-15 a novější).

Moderní úsporné procesory architektury x86 (například Intel Atom) jsou energeticky efektivnější (tedy na splnění úlohy spotřebují méně energie) než procesory architektury ARM zaměřené na výkon (Cortex A-15). ARM procesory zaměřené na úsporu (Cortex A9) jsou ve většině případů energeticky efektivnější než moderní úsporné procesory architektury x86.[9] [10]

## 4.2 ARM počítače

### 4.2.1 RaspberryPi

RaspberryPi byl první miniaturní počítač s ARM SoC „o velikosti kreditní karty“, který se výrazně proslavil a započal vlnu vývoje podobných desek a klonů této desky s drobnými úpravami (dále zmíněné Orange Pi, BananaPi a další). Tvůrcem této desky je RaspberryPiFoundation, která desku vytvořila pro podporu výuky počítačových věd nejen ve vývojových zemích. [11] [12]

Deska se nakonec uchytila nejen jako počítač pro výuku počítačových věd v úplně celém světě, ale stala se jednoduchým a levným řešením pro mnoho projektů, počínaje využitím v chytrých domácnostech či snímání zeměkoule z orbity.[13] Speciální verze RaspberryPi se dokonce dostala na vesmírnou stanici.[14]

Dne 18. 2. 2015 bylo vydáno prohlášení na blogu RaspberryPiFoundation, že bylo vyrobeno 5 miliónů kusů desky. [15] Díky tomu bylo mimo jiné možno podporovat výuku zdarma či podpořit projekty jako například Kodi, PyPy, Libav, Pixman, Wayland/Weston, Squeak, Scratch, Webkit nebo KiCad.

Dne 29. 2. 2016 bylo oznámeno prodání celkem 8 miliónů kusů desky, což z RaspberryPi udělalo nejprodávanější britský počítač v historii. [16]

#### 4.2.1.1 RaspberryPi B+

Deska je osazena SoCintegrovaným obvodem Broadcom BCM2835, obsahujícím 700 MHz ARM1176JZF-S procesor, VideoCore IV GPU a 512 MB RAM.[17]

System je zaváděn z MicroSDHC karty. Na desce jsou k dispozici 4 USB 2.0 výstupy, které jsou ale společně s výstupem ethernetu svedeny na jednu USB 2.0 sběrnici. Tím je součet rychlostí na všech USB portech a ethernet portu limitována rychlostí jednoho USB 2.0.

RaspberryPi také podporuje akcelerované dekódování videa, a to ve formátech H.264/MPEG-4 AVC a při dokoupení licence také formáty MPEG-2 a VC-1[18]. Tyto formáty mají placenou licenci, a aby se cena desky nezvyšovala všem, dokoupit si je mohou jen zájemci.

Deska podporuje připojení kamery přes CSI. Monitor lze připojit přes běžný HDMI port. O zvukový výstup se stará analogová 3.5mm zdířka nebo HDMI. Desku lze napájet pomocí microUSB portu nebo pomocí GPIO výstupů.

Mezi podporované systémy se řadí především ty s jádrem GNU/Linux. Distribucí pro RaspberryPi je velká řada. Doporučená distribuce je Raspbian, vycházející z Debianu. Mezi zajímavé distribuce se řadí například Kali Linux nebo OpenWRT.

Při použití přídatné desky Gertboard je možné použít obrovské množství GPIO výstupů.

#### **4.2.2 Orange Pi**

V době výběru bylo dostupných několik modelů Orange Pi desek: modely Orange Pi 2, One a PC. Prodejcem desek je ShenzhenXunlong Software (CO.,Limited). Všechny tři modely jsou vybaveny čtyřjádrovým procesorem H3 Cortex-A7. Všechny desky podporují dekódování moderních video formátů (například h265/HEVC ve 4k rozlišení). Desky jsou osazeny 10/100M ethernetem, 40 GPI výstupy kompatibilních s RaspberryPi B+. Pro připojení k displeji lze použít klasický HDMI konektor, případně DVI-D <-> HDMI redukcí. Jako systémové úložiště lze použít TF nebo MMC kartu.[19]

K napájení slouží takzvaný barelový konektor, který poskytuje oproti microUSB možnost průtoku vyššího proudu než 1.8 A.[20]

Všechny desky jsou akcelerovány pomocí GPU Mali400, podporující OpenGL ES 2.0 a běžících na 600 MHz. Všechny desky jsou osazeny USB OTG konektorem a také umožňují připojení kamery přes vstupní video konektor.

Mezi podporované systémy se řadí Android, Debian, Raspbian a komunitní Armbian (poslední dva jsou založeny na distribuci Debian).

Model One je osazen 512 MB DDR3 pamětí a pro připojení příslušenství slouží 1 USB port. U modelu PC je kapacita RAM navýšena na 1024 MB DDR3, počet USB konektorů je navýšen na 3 a na desce je IR přijímač. U modelu Orange Pi 2 jsou připraveny dokonce 4 USB porty. [19]

#### **4.2.3 BananaPi**

Mezi dostupné desky v době výběru od společnosti ShenzhenLeMaker Technology (Co., Ltd.) patřily modely M1, R1 (v některých zdrojích uváděn jako Routerboard) a M2.

Všechny desky BananaPi jsou dodávány s 1 GB DDR3 pamětí. Tedy dostatek pro běh odlehčených serverových a desktopových aplikací.

Desky po hardwarové stránce podporují dekodování videa. Nicméně pouze s původním, již v době vydání zastaralým, linuxovým jádrem verze 3.4. Pro desky lze použít i aktuální verzi linuxového jádra, ale některé funkcionality zde chybí. Například právě dekodování videa. [21]

Displej lze připojit pomocí klasického HDMI konektoru, který je velmi pevně přichycen k desce u všech modelů.

Distribuce systému jsou pro desky BananaPi především komunitní. A to z toho důvodu, že oficiální distribuce systémů je velmi často problematická a ne příliš aktualizovaná. Úspěšná komunitní distribuce systému je Armbian.[21] Mezi další systémy patří například Debian, Android a v případě modelu R1 také port OpenWRT.

K napájení slouží micro-USB port. Napájení tímto způsobem přináší určité problémy: Především se jedná o vytrácení příkonu přímo-úměrně k délce kabelu. Dalším problémem je limit proudu u micro-USB konektoru 1.8A[20], což nemusí být vždy dostatek. Za zmínku také stojí obtížnost sehnat zdroj, který by poskytoval stabilní přívod el. energie, nekolísal a především poskytoval takové parametry, jako uvádí na štítku.

Všechny modely podporují vstup kamery, GPIO částečně kompatibilní s RaspberryPi, zvukové výstupy přes HDMI nebo analogový 3.5mm jack a IR přijímač.[22][23]

#### 4.2.3.1 BananaPi M1

Tento model je osazen procesorem ARM Cortex-A7 a grafickým čipem Mali400 podporujícím OpenGL ES 2.0.

USB porty jsou u BananaPi M1 řešeny jinak než u ostatních desek – každý USB port je zde připojen přímo do procesoru a nedělí se tak o jednu USB2.0 sběrnici. Na desce jsou 2 klasické USB porty a jeden USB OTG port, ze kterého je možné pomocí redukce vytvořit klasický USB port. Celkem jsou zde tedy 3 USB porty, každý o teoretické rychlosti specifikace USB 2.0.

Na desce je umístěn také SATA konektor plně podporující standard verze 2.0. Tento konektor není sveden na USB sběrnici a poskytuje skutečně rychlý zápis na disk. Pro

napájení 2.5“ disků slouží přímo konektor na desce, není tedy potřeba napájet disk vlastním zdrojem. Především při použití rotujícího pevného disku je zapotřebí kvalitní zdroj (minimálně skutečných 5V/2A bez kolísání při zatížení).

#### 4.2.3.2 BananaPi R1

Model je také osazen procesorem ARM Cortex-A7 a i ostatní součásti jsou zde stejné jako u modelu M1. Navíc je zde 300 Mbit/s WiFi-n se dvěma anténami a další čtyři ethernet porty. Deska je rozměrově větší a umožňuje přímé přichycení 2.5“ disku na desku.

Parametry bohužel neodpovídají výkonu v reálném prostředí: Antény dodané s deskou mají minimální výkon, wifi při větší zátěži vypadává a je problém připojit již jen několik zařízení naráz. Sekvenční přenos dat po wifi přes nešifrované FTP je maximálně rychlostí 40 Mbit/s. 5 ethernetových portů je svedeno do jednoho přepínače a teoretická rychlost pro všechny porty dohromady je tedy jen 1 Gbit/s.

#### 4.2.3.3 BananaPi M2

Tato deska je osazena čtyřjádrovým Cortex-A7 ARM procesorem, PowerVRSGX544MP2 GPU podporující OpenGL ES 2.0, jedním gigabitovým ethernetovým portem a čtyřmi USB 2.0 porty. [24]

Jako jediný model BananaPi nemá SATA rozhraní, neumožňující přímé připojení disku nativně přes SATA rozhraní a je nutné použít USB redukci, což přenos dat velmi zpomalí. Nástupce řady, model M3, nebyl ještě v době výběru dostupný. I přesto, že je na modelu M3 výstup SATA, je pouze přemostěn na USB sběrnici. Rozhraní tak dosahuje velmi malých rychlostí.[25]

### 1.1.1. Odroid C1

Odroid C1 je produktem společnosti Hardkernel (Co., Ltd.). Rozměrově je menší než ostatní desky, ale díky dobrému návrhu není množství použitých konektorů omezeno.

Deska je osazena čtyřmi USB 2.0 konektory a jedním USB OTG konektorem. USB konektory jsou bohužel pouze rozbočeny rozbočovačem z jedné USB sběrnice. Tím pádem je maximální propustnost rovna specifikaci jednoho USB 2.0 portu, přičemž reálná přenosová rychlost nepřesahuje sekvenční přenos dat 30 MB/s. Osazení všech USB portů na delší dobu nebo osazení portů dalšími rozbočovači vedlo k selhání celého systému, USB



subsystému nebo odpojování zařízení. Také docházelo k přehřívání některých součástek a později celé desky.[26] [27]

Deska je osazena velmi výkonným čtyřjádrovým procesorem. Tento výkonný procesor se na rychlosti a odezvě celé desky podepisuje velmi silně. Výkon je velmi znatelný například při prohlížení webových stránek, kde ostatní desky mají obrovské problémy s plynulostí.

Osazená paměť 1 GB (2x512 MB) je zcela dostatečná pro běh serverových aplikací ale i aplikací běžného desktopu s odlehčeným prostředím jako LXDE nebo XFCE . Připravené distribuce i hardware (GPU) podporují akceleraci moderních video formátů, což je užitečné v případě volby použití zařízení jako multimediální centrum s obrazovkou.

Na desce je i Gigabitový ethernet konektor, který byl ale v době výběru softwarem podporován pouze v rychlostech 10/100 Mbit. Dále zde najdeme Infračervený přijímač pro dálkové ovládání a GPIO piny téměř plně kompatibilní s RaspberryPi.[28]

Pro připojení k displeji je možné použít HDMI-micro. Tento konektor sice zabírá minimum prostoru a umožňuje zapojení do obrazovky podporující HDMI nebo pomocí redukce DVI-D, ale konektor není příliš pevný, kabel není pevně přichycen a při pohybu kabelu obraz vypadává. Konektor také není na desce příliš dobře připevněn a je zapotřebí ne příliš běžná HDMI-micro redukce na straně desky.

Podporované operační systémy výrobcem jsou Ubuntu (verze 14.04) a Android (KitKat). Systém může být uložený na SD kartě nebo na eMMC paměti – obojí je dodáváno zvlášť.

### **4.3 Speciální distribuce pro desky**

Mnoho distribucí je upraveno a odladěno pro potřeby deskových počítačů s méně výkonnými ARM procesory. Pro potřeby této práce jsou důležité distribuce Raspbian a Armbian.

#### **4.3.1 Raspbian**

Distribuce odvozená z distribuce Debian a upravená pro RaspberryPi. Pro první modely poskytoval balíčky kompilované speciálně pro její procesor a na rozdíl od distribuce Debian obsahoval balíčky s hardwarovou podporou pro pohyblivou řádovou čárku (armhf). Díky tomu poskytoval Raspbian vyšší výkon aplikací. Tato distribuce funguje také na kloních RaspberryPi, jako je třeba BananaPi.[29]

### 4.3.2 Armbian

Armbian je distribuce odvozená z distribuce Debian (Wheezy či Jessie) nebo z distribuce Ubuntu (Trusty/14.04). K dispozici jsou obrazy disků pro více než 27 deskových počítačů, mimo jiné pro klony RaspberryPi (jako například Orange Pi či BananaPi). [30]

Mimo to ale Armbian nabízí veřejně skripty, ze kterých obrazy sestavuje. Tyto skripty jsou uživatelsky přívětivé, ale na pozadí velmi komplexní. Umožňují zvolit desku, požadované jádro (většinou „legacy“ – původní jádro vydané společně s deskou výrobcem, „mainline“ – poslední oficiální stabilní jádro Linuxu z webu [www.kernel.org](http://www.kernel.org) nebo „development“ – poslední vývojová verze s nejnovějšími změnami.), požadovanou distribuci (většinou již zmíněný Debian nebo Ubuntu) a volbu podpory X serveru. Po zadání těchto preferencí skript stáhne zdrojové balíčky, aplikuje úpravy (*patche*) pro konkrétní desku, zkompiluje balíčky a vytvoří obraz disku (karty), který jen stačí nahrát na paměťovou kartu. Tento způsob umožňuje vytvořit obraz disku podle svých potřeb a se zcela aktuálními balíčky.

## 4.4 Aplikace na straně serveru

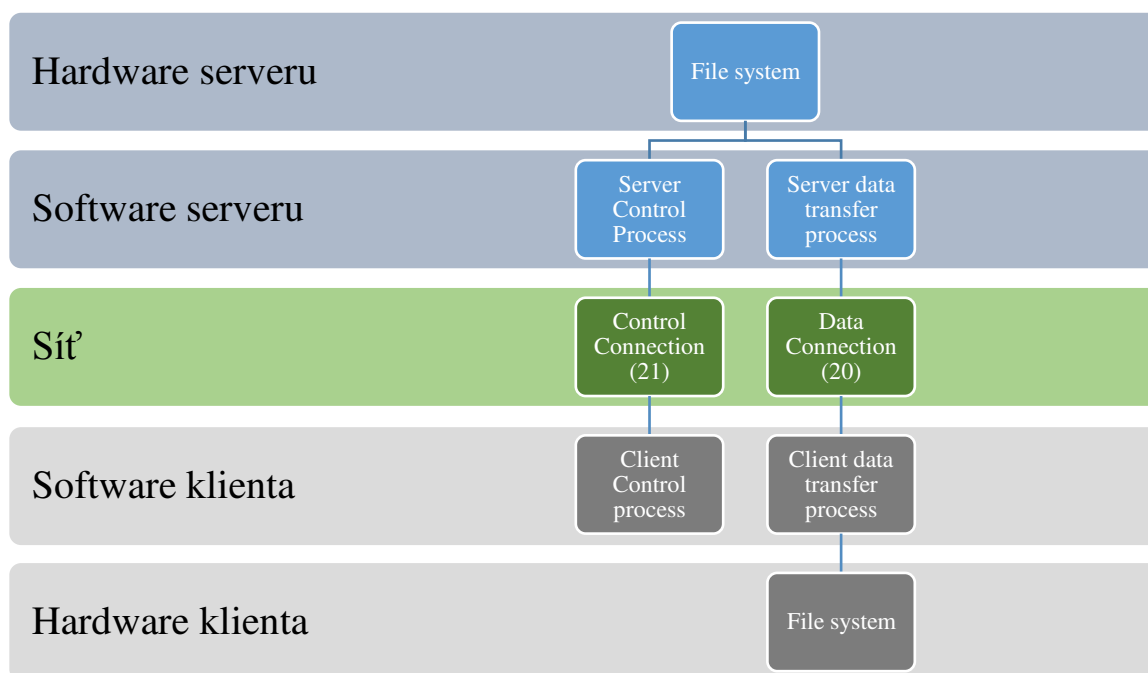
### 4.4.1 FTP server

FTP – File Transfer Protocol – je velmi starý (původní specifikace vznikla v roce 1971), nikoli zastaralý, protokol, používaný k přenosu souborů (mezi klientem a serverem) a operace s těmito soubory.

Tento protokol byl vyvinut pro zjednodušení a podpoření použití vzdálených počítačů, respektive dat na vzdálených počítačích, a to především dalšími programy.

Na straně klienta se nemusí řešit souborový systém disku, ze které jsou data na serveru fyzicky čtena. To má na starosti server. Ten také podle přihlášeného uživatele řídí přístup jen k povoleným souborům a adresářům a omezuje operace s nimi.

Komunikace probíhá ve dvou spojeních. V jednom probíhá komunikace pomocí sady předdefinovaných příkazů (*řídící spojení*) a v druhém probíhá tok samotných dat (*datové spojení*). Ve výchozím nastavení běží řídicí spojení na portu 21 a datové na portu 20.[31]  
[32] [33]



Obrázek 1 - Práce FTP serveru

#### 4.4.2 SSH server

SSH – Secure Shell – je protokol pro bezpečné vzdálené přihlášení a ostatní síťové služby přes nezabezpečenou síť. Umožňuje interaktivní přihlášení, vzdálené spouštění příkazů, X11 spojení a TCP/IP spojení, včetně posílání souborů.

Kdykoli jsou data odeslána, jsou zašifrována, a když dorazí do cíle, jsou rozšifrována. Tím pádem program, případně uživatel, může pracovat normálně, a to i bez znalosti skutečnosti, že je spojení zašifrováno. SSH je dostatečně silné a používají jí i kritické aplikace.

SSH pracuje na architektuře Klient/Server. Server přijímá či odmítá požadavky a příchozí spojení k počítači, na kterém běží. Klient pak tyto požadavky posílá. Všechny přenosy jsou zašifrovány a ochráněny proti změně a probíhají jen po úspěšném přihlášení klienta k serveru. SSH je bezpečnější alternativa protokolu Telnet. [34] [35] [36]

Tabulka 1 - Protokoly, které může SSH nahradit pro lepší zabezpečení

SSH	Protokol
SSH (přístup terminálem)	RSH, Rlogin, Rexec
SFTP	FTP, SMB, NFS, RCP
SSH/Chat	IRC
SSH/Backups	RSync

#### 4.4.2.1 SFTP

SFTP – SecureFile Transport Protocol je subsystém použitý společně s SSH. SFTP se velmi podobá FTP, nicméně nabízí možnost přenášet soubory ze serveru (nebo na server) ke klientovi (nebo od klienta) zabezpečeným spojením.

Nevýhodou SFTP je ztráta výkonu, protože všechny odeslaná data se před odesláním musí zašifrovat. Pokud jsou obě strany, jak klient, tak server, výkonově silné, ztráta bude zanedbatelná. Pokud bude ale přinejmenším jedna ze stran výkonově slabá, bude rychlost přenosu omezena.[36]

#### 4.4.3 Samba

Samba je síťový nástroj, který umožňuje sdílení souborů a tiskáren mezi různými operačními systémy. Samba umožňuje OS Windows přistupovat na zdroje sdílené Unixovým systémem a naopak umožňuje Unixovému systému přistupovat ke zdrojům OS Windows.[37] [38]

SMB/CIFS mohou pracovat buďto přes TCP/IP nebo přes NetBEUI (případně přes IPX/SPX). V Unixových systémech se NetBEUI netěší takové podpory a proto je většinou použito TCP/IP. Samba pracuje na principu server – klient, kde klient posílá požadavky serveru a server odpovídá.[39]

#### 4.4.4 LAMP

LAMP – Linux, Apache, MySQL, PHP – je balík softwaru používaný pro tvorbu a zpřístupnění dynamických webových stránek a webových aplikací. Linux zde zastupuje operační systém, Apache zastupuje webový server, MySQL systém řízení báze dat a PHP skriptovací jazyk na straně serveru. [40]

Tyto jednotlivé části balíku lze nicméně vhodně zaměňovat za jiné, pro dané účely vhodnější zástupce.

##### 4.4.4.1 Web server

Web server je počítač, respektive program, běžící na počítači, který přijímá HTTP požadavky. A jeho úkolem je požadavky zpracovat a odpovědět – doručit obsah klientům. Obsah jsou nejčastěji HTML dokumenty, obrázky, skripty a další. Pokud server nemůže obsah poskytnout, pošle klientovi chybovou zprávu.

HTTP je protokol, kterým mezi sebou klient a server komunikují.

Web server podporuje propojení s dalším softwarem, jako je PHP, ASP a další. Běžně je tento software spojen také s databází. To umožňuje poskytování dynamického obsahu klientům. Poskytování statického obsahu je sice rychlejší, ale neumožňuje klientovi zobrazovat dynamický obsah. [40]

Mezi nejpoužívanější webové servery patří Apache, IIS a nginx. Dalším zajímavým web serverem je lighttpd, který vyniká především svou nenáročností a je dobře použitelný na slabém hardwaru.

#### 4.4.4.2 PHP

PHP – rekurzivně **PHP: Hypertext Preprocessor**, původně **PersonalHomePage** – je skriptovací jazyk na straně serveru. PHP je použito především pro vývoj webových aplikací a dynamických webových stránek.

Pokud klient po serveru požaduje PHP stránku, PHP skript je proveden na straně serveru a ke klientovi je odeslána až výsledná zpracovaná stránka. PHP kód bývá často vložen přímo do HTML kódu.

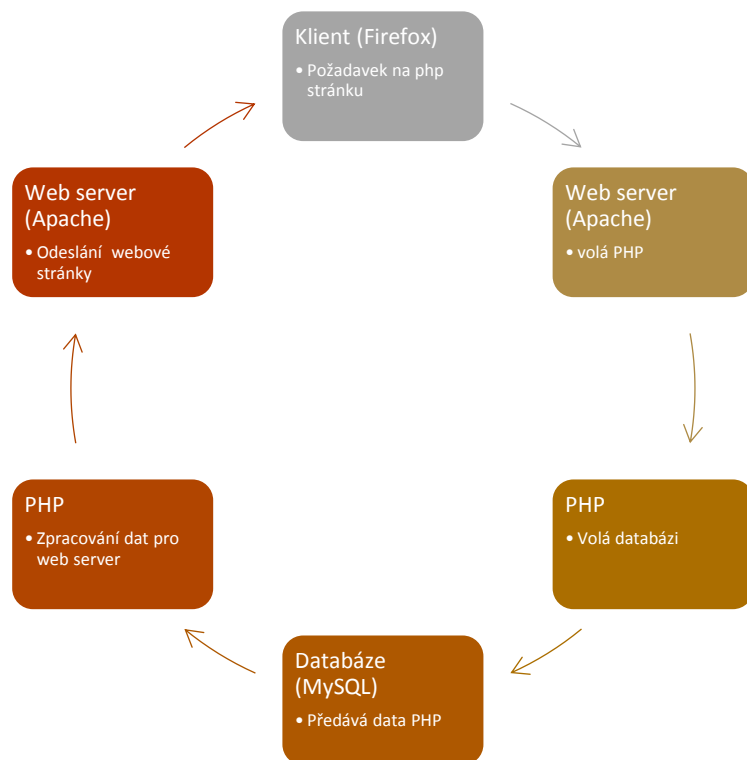
PHP je podporováno na velkém množství platform a operačních systémů. Kód je mezi nimi možné přenášet většinou bez jakýchkoli úprav. PHP obsahuje knihovny, které mu umožňují pracovat se vším, co je potřeba pro webovou aplikaci: práce s texty, grafikou, soubory, databázemi a internetovými protokoly. [40] [41]

#### 4.4.4.3 Systém řízení báze dat

Zkráceně SŘBD (anglicky DBMS) je software, který umožňuje přístup k databázi a zprostředkovává data uvnitř této databáze. Tvoří tedy můstek mezi jinou aplikací požadující data z databáze a samotnými daty databáze.

SŘBD přidává, upravuje a odstraňuje záznamy do databáze a definuje její strukturu.[40][42]

Mezi Systémy řízení báze dat se řadí například Oracle databáze, DB2, MariaDB, MySQL nebo SQLite.



Obrázek 2 - Zjednodušený diagram práce webového serveru s dynamickou webovou stránkou

#### 4.4.5 VLC server

VLC media player je přehrávač multimédií s otevřeným zdrojovým kódem podporující obrovské množství formátů a streamovacích protokolů. Umožňuje také streamy vysílat.

Je k dispozici pro široké spektrum platform (MS Windows, distribuce Linuxu – včetně ARM architektury, zařízení Apple, a další). VLC media player je tvořen neziskovou společností VideoLAN.[43]

VLC media player obsahuje mnoho doplňků pro specifické služby. Jeden z těchto doplňků je http lua modul. Tento modul vytvoří pro VLC media player webový server, skrz který lze VLC media player ovládat. Ovládání přehrávače je tedy možné ze vzdáleného počítače pomocí webového prohlížeče, respektive pomocí http spojení.[44]

Právě díky možnosti http spojení je možné příkazy zasílat jinak, než pomocí webového prohlížeče. Například je možné použít aplikaci v telefonu nebo tabletu (která tak nahrazuje webový prohlížeč).[45]

#### 4.4.6 P2P

P2P – Peer to Peer – je pojmenování pro model sítě, kde jsou si všichni členové rovni. Na rozdíl od modelu Klient/Server zde není žádný hlavní server.

Od modelu Klient/Server se P2P liší také tím, že čím více je v síti aktivních strojů, tím větší je přenosová kapacita sítě, protože si data navzájem vyměňují. Klienti jsou samozřejmě limitováni fyzickou kapacitou své linky, kterou jsou připojeni do sítě. [46]

Mezi programy, které používají model P2P se řadí nejen klienti pro stahování tzv. torrentů (Bittorrent, Vuze, Deluge, kTorrent, rTorrent a další), často obsahující nelegální obsah nebo malware. P2P využívají ale i mnohem zajímavější klienti pro sdílení dokumentů mezi zařízeními uživatele (například BittorrentSync) a další zajímavé projekty. Například univerzitní systém LionShare pro sdílení dat nebo Osiris, který umožňuje uživatelům vytvářet anonymní webové portály distribuované přes P2P síť.

### 4.5 RAID

#### 4.5.1 Co je to RAID

RAID (anglická zkratka pro Redundant Array of Inexpensive, nebo také Independent, Disks, tedy redundantní pole levných nebo nezávislých disků) používá několik disků k dosažení ochrany dat, zvýšení výkonu nebo zvýšení kapacity datového nosiče.

Umožňuje uložení stejného souboru na různých discích (tím je zaručena redundance dat a v případě poškození jednoho disku jsou data stále k dispozici) nebo také rozdělení souboru střídavě na všechny disky (to zvyšuje výkon datových nosičů, protože data jsou rozdělena na menší segmenty a ty se pak zapisují na všechny disky zároveň). Stejně tak ale může tyto metody kombinovat a pomocí složitějších konfigurací dosáhnout jak vysokého výkonu, tak redundance dat. RAID tedy používá různé „úrovně“ (jinak řečeno architektury) pro dosažení požadovaného cíle. Použitá úroveň definuje, jak budou data rozmístěna na discích zapojených do RAID pole. [47] [48]

RAID může být řízený **softwarově** nebo **hardwarově**.

**Softwarově** řízený RAID využívá prostředky hosta (procesor, paměť atd.). Buď to se jedná o čistě softwarové řešení nebo do systému může být přidán asistenční hardware usnadňující hostovi práci s polem. [49]

**Hardwarově řízený RAID** pro běh používá vlastní prostředky. Vytížení se tedy přesouvá z hosta na HW RAID zařízení.

## **4.5.2 Běžné úrovně RAID [48] [50]**

### 4.5.2.1 Level 0

Používá metodu „striping“, tedy prokládání bloků jednotlivých souborů mezi všemi disky. Neposkytuje ale žádnou ochranu dat. Dokonce při selhání jednoho disku jsou ztracena všechna data z celého pole.

Tato úroveň nicméně poskytuje nejvyšší výkonnost a kapacita úložiště je součet kapacit všech použitých disků.

### 4.5.2.2 Level 1

Používá metodu „mirroring“, tedy zrcadlení dat na discích. To znamená, že při selhání jednoho disku data nejsou ztracena lze s nimi dále pracovat.

Zapisovací rychlost je taková, jako nejpomalejšího disku v poli. Čtecí rychlost je maximálně součet rychlostí všech disků v poli, umožňuje tedy číst různá data z různých disků zároveň.

### 4.5.2.3 Level 2

Používá metodu „error-correctingcoding“. Není často používána. Rozkládá data na bity, nikoli na bloky.

### 4.5.2.4 Level 3

Používá metodu „Bit-interleaved parity“. Používá vyhrazený paritní disk. Díky němu je možné v případě poškození jednoho z disků dopočítat původní obsah.

Nevýhodou je, že paritní disk může zpomalovat práci pole.

### 4.5.2.5 Level 4

Používá metodu „Dedicated parity drive“. Umožňuje prokládání na úrovni bloků (jako u úrovně RAID 0) společně s paritním diskem. Pokud jeden z disků selže, lze dopočítat původní data a zapsat je na nový disk. Nevýhodou je, že paritní disk může zpomalovat práci pole.



#### 4.5.2.6 Level 5

Používá metodu „Blockinterleaveddistributed parity“. Umožňuje prokládání disků společně s informací pro případnou obnovu. Tato informace je ale tentokrát pokaždé na jiném disku, není zde tedy paritní disk zpomalující RAID pole. Výsledkem je výborná výkonnost a zároveň dobrá ochrana dat.

#### 4.5.2.7 Level 6

Používá metodu „Independent data diskswith double parity“. Používají se dva paritní disky. Obě parity jsou počítány jiným způsobem. Poskytuje ochranu i při selhání dvou disků. Stejně jako u úrovně RAID 5 jsou paritní informace rozprostřeny po všech discích, aby nedocházelo k přetížení jednoho disku.

#### 4.5.2.8 Level 10

Používá metodu „A stripeofmirrors“. Jedná se o kombinaci úrovní RAID 1 a RAID 0. Data jsou redundantní na každé větvi RAID 0 a mezi těmito větvemi se data prokládají. Je tedy odolný vůči výpadku disku na každé větvi.

Výhodou je velká rychlost, naopak nevýhodou je jen 50% kapacita RAID pole.

#### 4.5.2.9 Level 01

Používá metodu „A MirrorofStripes“. Znovu se jedná o kombinaci úrovní RAID 1 a RAID 0. Data jsou prokládána na každé větvi RAID 1. Znovu je zde ochrana proti výpadku disku.

#### 4.5.2.10Level 7

Odvozené od RAID 3 a RAID 4, ale využívá vyrovnávací paměť.

#### 4.5.2.11Level 50

V této metodě se prokládá několik RAID 5 polí. Je dosaženo vyšší rychlosti, ale v každé větvi je zapotřebí jeden paritní disk.

#### 4.5.2.12 Level 60

V této metodě se prokládá několik RAID 6 polí. V každé větvi jsou dva paritní disky. Dosahuje se tedy vyššího zabezpečení než u úrovně RAID 50. Na druhou stranu je zapotřebí další disk v každé větvi.

#### 4.5.2.13 Level 100

V této metodě je zapotřebí minimálně 8 disků. Jsou zde dvě úrovně prokládání a disky jsou vždy zrcadleny.

## 4.6 Vstupně-výstupní plánovač

Anglicky I/O Scheduler. Vstupně-výstupní plánování je metoda, kterou používají počítačové operační systémy k rozhodnutí, který datový blok bude kdy zapsán do datového nosiče.

Různé plánovače mají různé cíle a jsou optimalizovány pro určitou činnost nebo mají nastavené různé priority: [51] [52]

### 4.6.1 CFQ

Completely Fair Queuing – Řadí požadavky do front podle procesu. Doba, jakou je umožněno procesu přistupovat k disku se určuje podle jeho priority.

Tento plánovač je nejvýkonnější na magnetických – pohyblivých – discích, protože mimo jiné řeší i fyzické umístění dat na disku. Často je volen jako výchozí.

### 4.6.2 NOOP

No-Op – NoOptimisation. První příchozí požadavek je zpracován první, druhý je zpracován jako druhý a tak dále. Tomuto způsobu se říká FIFO – First In First Out.

Využití především u hardwarových RAID polí, kde si HW pole plánování řeší samo a systém se tak nezatěžuje s plánováním.

### 4.6.3 DEADLINE

Dává každému I/O požadavku expirační dobu, odtud „Deadline“. Když je I/O požadavek zapsán, zapíše se i I/O požadavky ve (fyzickém) okolí, aby se předešlo „poskakování“

hlavy (v případě použití mechanických „pohyblivých“ disků). Předpokládá se jeden proces zatěžující datový nosič, a tak se neřeší spravedlivost, pokud existuje více procesů s I/O požadavky. Dosahuje velmi dobrých hodnot na pamětech typu flash. Používá se také pro úložiště s databázemi.

## **4.7 Souborové systémy**

Souborový systém popisuje organizaci dat na datovém nosiči (neboli médiu). S daty tak mohou pracovat všechny stroje schopné souborovému systému (organizaci dat) porozumět.

Souborové systémy se používají i na optických médiích.

### **4.7.1 Ext2 – Second extendedfilesystem**

Již starší souborový systém, který nepoužívá žurnálování. Nahradil Ext. [53]

### **4.7.2 Ext4 - Fourtextendedfilesystem**

Modernější souborový systém, v dnešní době je v Linuxových distribucích často jako výchozí, kde nahradil Ext3. Používá žurnálování. [54]

### **4.7.3 F2FS - Flash-FriendlyFileSystem**

Souborový systém optimalizovaný pro flash paměti (mající rozdílnou charakteristiku než rotující magnetické disky), jako jsou SSD, eMMC nebo SD karty. Je stále ve vývoji. [55]

### **4.7.4 NTFS**

Souborový systém NTFS má uzavřený zdrojový kód. Byl vyvinut firmou Microsoft pro svůj operační systém Windows. Linux umožňuje použití svobodného NTFS ovladače v podobě ntfs-3g.[56]

## 5 Vlastní práce

### 5.1 Výběr vhodného jednodeskového počítače pro serverové aplikace

Při výběru je nutné rozvrhnout, k jakému účelu bude jednodeskový počítač používán, tedy jaké součásti jsou pro daný záměr důležité. Jak velká je zapotřebí spolehlivost. Co se očekává od podpory produktu autorem. Kolik je možno investovat. Ale i jak velká deska může být nebo pokud bude viditelná, jestli jsou k dispozici vhodné krabičky, do kterých se počítač vloží.

Pokud by byl záměr zařízení použit pro projekt, na kterém jsou závislí přímo klienti, bylo by vhodné využít deskový počítač s co největší podporou autora, aby v případě problému bylo vše v pořádku co nejdříve.

Pokud by byl záměr použit počítač pro připojení k projekci, zobrazování reklamy, jako počítač pro návštěvníky, veřejný rezervační systém či informační panel v hale, bude se od zařízení vyžadovat stabilní video výstup a dostatečný grafický výkon uživatelského rozhraní. Pokud bude počítač viditelný, měl by mít co nejhezčí obal a zároveň by neměl být moc velký.

Tato práce je zaměřená na univerzálnější použití jednodeskového počítače pro domácnost, malou firmu, popřípadě pro kancelář. Očekává se tedy, že bude velmi dobře zvládat funkce síťového prvku (a úložiště), bude mít dostatečný výkon na provoz serverových démonů, rychlé připojení k síti a bezproblémovou stabilitu. Naopak není požadován grafický výkon (jak 2D pro uživatelské rozhraní, tak 3D) a vzhledem k ne moc příznivým možnostem na trhu není také požadována velká podpora ze strany autora desky. To především pro to, že existuje obrovská komunita, která se podporuje navzájem a do značné míry tak doplňuje podporu autora. Příkladem může být například projekt Armbian, spravující software pro téměř 30 deskových počítačů a fungující čistě komunitně. Cena desky pro není rozhodující, spíše orientační, jelikož se desky pohybují ve zcela stejné cenové kategorii. Nicméně je vyžadována naprostá spolehlivost zařízení, ať už softwaru nebo hardwaru. Nesmí docházet k žádným výpadkům ani restartování zařízení. Ani chybám na žádném použitém rozhraní (USB, SATA, síťové rozhraní a další) nemohou být tolerovány. Zařízení musí zvládnout běh 24/7/365. Velikost ani vzhled obalu nejsou pro cíl této práce důležité.

### 5.1.1 Porovnání vlastností a vhodnost pro nasazení

Vlastnosti jednodeskových počítačů jsou sjednoceny do přehledné tabulky v příloze (**Příloha 1 - Tabulka porovnání vlastností jednodeskových počítačů.xlsx**)

#### 5.1.1.1 RaspberryPi B+

RaspberryPi B+ je díky své spolehlivosti a podpoře autorem velmi dobrý a univerzální kandidát pro téměř jakýkoli projekt. Návrh desky je dobře zpracován, existuje na ní mnoho doplňků a komunita je obrovská.

RaspberryPi B+ není vhodný počítač na pro použití jako server s potřebou rychlého připojení do sítě. A to hned z několika důvodů: Nemá SATA rozhraní, tudíž se disky mohou připojit pouze přes USB. Síťové rozhraní je zde pouze o rychlosti 100 Mbit/s, což pro síťové použití není příliš vhodné. Ethernetový výstup dokonce vede přes USB hub, na který jsou připojeny i všechny USB výstupy. Ukládání dat ze sítě na připojený disk skrze USB je tedy pomalé a řízení velkého provozu přes USB sběrnici zahltí procesor mající pouze jedno jádro. Pokud je přenos šifrován (například SFTP), rychlost přenosu je kriticky nízká.

Kvůli pomalému připojení k síti a pomalému připojení k úložišti není RaspberryPi vhodný deskový počítač pro cíl této práce.

#### 5.1.1.2 BananaPi M1

BananaPi M1 patří také mezi spolehlivé jednodeskové počítače nemající problém se stabilitou. Podpora autorem zde ale není prakticky žádná a jím poskytované obrazy SD karet někdy nejsou ani funkční. Zdrojový kód k úpravám poskytuje velmi sporadicky a především jiné, než ze kterých sestavuje publikované obrazy.

Přesto, především díky komunitní činnosti, bez které by byl počítač hůře použitelný, patří tento deskový počítač k těm nejzajímavějším pro cíl této práce. Jeho součásti totiž nejvíce odpovídají požadovaným vlastnostem: Nachází se zde 1 Gbit/s výstup pro připojení do sítě zaveden přímo do procesoru (čili není použit žádný USB hub jako u RaspberryPi), dvou jádrový procesor, poskytující dostatek výkonu pro přenosy přes rychlé síťové rozhraní, dostatečný 1 GB operační paměti, ale také SATA rozhraní, které je opět zavedeno přímo

do procesoru bez využití USB můstku. Mimo to najdeme na desce celkem 3 sběrnice USB zavedené opět přímo do procesoru, čili je možné použít plných 3xUSB 2.0 linek.

Díky rychlému připojení k síti a možnosti rychlého připojení úložiště je BananaPi M1 vhodný pro cíl této práce.

#### 5.1.1.3 BananaPi R1

BananaPi R1 patří mezi ty nejslabší kandidáty snad pro jakýkoli projekt. Jedinou světlou výjimkou by mohlo být použití desky k výukovým účelům konfigurace sítě. Parametry desky uvádí prakticky to samé, jako model BananaPi M1, ale jsou přidány 4 ethernet výstupy a Wi-Fi. Na první pohled by šlo o ideální síťový prvek zcela vhodný pro cíl této práce, nicméně ve skutečnosti jsou všechny ethernetvýstupy vedeny na jeden rozbočovač umožňující maximální celkový tok po výstupech 1 Gbit/s. Wifi konektivita je na tom ještě o něco hůře, parametry sice slibuje standard 802.11n a rychlost 300 Mbit/s, ale skutečnost je mnohem horší a signál není ani stabilní. Počítač měl také dlouho problémy s napájením SATA disku, což posléze komunita vyřešila úpravou konfiguračních souborů.

Díky všem těmto nedostatkům není tento deskový počítač vhodný pro cíl práce.

#### 5.1.1.4 Odroid C1

Odroid C1 je velmi zajímavá deska především díky výkonu svého procesoru a 1 Gbit/s výstupu pro síť, kterému softwarová podpora byla přidána až později po vydání (do té doby výstup umožňoval jen rychlost 100 Mbit/s), je vhodný pro použití jako síťové zařízení.

Bohužel není k dispozici SATA výstup a všechny 4 USB výstupy jsou zavedeny do jednoho USB hubu integrovaného na desce. Tím pádem sice Odroid C1 umožňuje rychlé připojení do sítě, ale neumožňuje žádnou možnost připojení k rychlému úložišti.

Proto je Odroid nevhodný pro cíl této práce.

#### 5.1.1.5 Orange PiOne, Orange Pi PC a Orange Pi 2

Všechny tři deskové počítače Orange Pi mají velmi dobrou komunitní podporu, nicméně jejich absence SATA výstupu a 1 Gbit/s síťového výstupu je pro cíl této práce překážkou.

Nedosahuje schopností BananaPi M1 a oproti RaspberryPi má mnohem menší podporu i komunitu.

### 5.1.2 Finální výběr

Pro cíl této práce je zřejmým výhercem BananaPi M1. Jeho výkon (síťový i výpočetní) a podpora komunitním operačním systémem Armbian z něho dělá výbornou možnost pro tento i podobné projekty. Linuxové jádro je komunitou stále doplňováno a téměř vše již funguje i v aktuálním vývojovém kernelu (verze 4.5).

Proto bude v práci nadále použit právě jednodeskový počítač BananaPi M1.

## 5.2 Příprava a instalace operačního systému

Nejprve je nutné na SD kartu připravit operační systém. Bude využito komunitního projektu Armbian – buďto stáhnutím jimi sestavených obrazů nebo sestavením vlastních obrazů podle svých preferencí pomocí příjemného konzolového průvodce.

Pro stažení výchozích obrazů stačí navštívit web projektu Armbian. Obrazy jsou publikovány v archivu společně s aplikací, která umožňuje instalaci systému na kartu z operačního systému MS Windows.

Pro sestavení obrazu podle vlastních preferencí je zapotřebí připravit prostředí. Pro bezproblémový chod skriptů je vhodné použít doporučený operační systém Ubuntu verze 14.04.

Využití standardního účtu je v příkazech naznačeno znakem \$ a příkazy vyžadující oprávnění uživatele root jsou naznačeny znakem # tak, jak by to naznačoval terminál.

Instalace programu git. Je vhodný pro stažení skriptů Armbian a nutný pro jejich chod.

```
# apt-get -yinstallgit
```

Klonování skriptů Armbian na počítač a vytažení hlavního souboru o úroveň výše:

```
$ gitclone --depth 1 https://github.com/igorpecovnik/lib  
$ cp lib/compile.sh .
```

po dokončení klonování je hned možné program spustit:

```
# ./compile.sh
```

Skript stáhne všechny potřebné závislosti, tedy potřebný software k provádění operací skriptu.

Poté se dotáže na desku, operační systém, možnost X serveru a verzi jádra. Poté začne automatizovaný proces klonování repositářů zavaděče, jádra a ovladačů, které jádro neobsahuje a poté skript aplikuje na zdrojové kódy úpravy (*patches*). Následuje kompilace všech stažených zdrojových kódů a vytvoření prázdného obrazu karty.

Do tohoto obrazu jsou staženy a nainstalovány všechny ostatní potřebné balíky (programy) a také programy zkompileované v minulém kroku. Pokud byl zvolen i X server na začátku, stáhnou a nainstalují se také všechny potřebné balíky pro spuštění grafického XFCE sezení.

Výstupem je znovu archiv, ve kterém se nachází obraz karty a aplikace pro instalaci systému na SD kartu z MS Windows.

Pod systémem Ubuntu bude příkaz na zapsání obrazu karty vypadat takto:

```
# ddbs=1M if=nazevsouboru.rawof=/dev/sdx
```

Kde **dd** je volaný příkaz, **bs=1M** je parametr určující zápis po 1MB blocích, **if** značí zdroj a **of** destinaci zápisu. **/dev/sdx** značí, že obraz nekopírujeme na logický oddíl, ale přímo na fyzické zařízení (kartu **sdx**).

Po dokončení přenosu je vhodné zavolat příkaz sync:

```
# sync
```

který zapíše všechna nezapsaná data na všechna fyzická zařízení. Poté můžeme kartu odpojit ze čtečky a zapojit do BananaPi M1.

Nyní můžeme provést první start BananaPi M1. První start trvá déle, než budou trvat další starty a je vhodné zapojit síťový kabel, jinak se start nemusí zdařit.

Od teď je možné pracovat jak s přímo připojenou klávesnicí, tak přes SSH vzdáleně na portu 22.



## 5.3 Konfigurace serverových aplikací

### 5.3.1 Konfigurace základního systému

Při prvním spuštění systém požádá o nastavení hesla uživateli root a o vytvoření nového uživatele a nastavení hesla tomuto uživateli.

#### 5.3.1.1 Aktualizace systému

Nyní je vhodné aktualizovat instalované balíky.

```
# apt-get update &&apt-getdist-upgrade
```

První příkaz zaktualizuje repositáře a druhý zaktualizuje balíky.

#### 5.3.1.2 Konfigurace souborových systémů.

Konfigurace souborových systému připojených po startu se nastavuje v souboru `/etc/fstab`. Je vhodné, aby byl adresář `/tmp` alokovan v paměti pro vyšší rychlost operací využívající dočasné soubory v `/tmp` a také pro snížení počtu zápisů na SD kartu. `/tmp` adresář je již ve výchozí konfiguraci souboru `/etc/fstab` alokovan do paměti, ale jeho maximální velikost je zbytečně malá. Je vhodné jí zvýšit alespoň na 512 MB. Také je možné přidat parametry **noatime** a **nodiratime**. Díky těmto parametrům se nebude v adresáři `/tmp` značkovat čas přístupu k souborům.

```
# nano /etc/fstab
```

zde stačí upravit první řádek:

```
tmpfs /tmptmpfsnoatime,nodiratime,nodev,nosuid,size=512M 0 0
```

Také logování je možné přesunout do paměti. Toho lze dosáhnout přidáním následujícího řádku:

```
none /var/log tmpfsnodev,nodiratime,nosuid,noatime,size=512M,mode=1777 0 0
```

Přípojný bod `/` je nastaven pro výkon již ve výchozím stavu. Je možné docílit vyššího výkonu přidáním parametru **barrier=0**. Použití `barrier=0` a `commit=600` může vést ke ztrátě nezapsaných dat.

```
/dev/mmcbk0p1 / ext4 barrier=0,noatime,nodiratime,data=writeback,commit=600 0 0
```

Pro změny je nutné počítač restartovat. Pro ověření nastavení je možné nahlédnout do souboru **/etc/mtab**.

```
# cat /etc/mtab
```

Výstup by měl vypadat takto:

```
/dev/root/ ext4 rw,noatime,nodiratime,nobarrier,commit=600 0 0
...
tmpfs/tmptmpfsrw,nosuid,nodev,noatime,nodiratime,size=524288k 0 0
none/var/logtmpfsrw,nosuid,nodev,noatime,nodiratime,size=524288k 0 0
```

### 5.3.1.3 Připojení a konfigurace disku pomocí SATA výstupu

Pomocí příkazu `fdisk` lze zjistit, jak je v systému veden SATA disk. „`| grep /dev/`“ upraví výpis tak, aby obsahoval jen relevantní informace:

```
# fdisk -l | grep /dev/
```

Výstup:

```
Disk /dev/mmcbk0: 7.4 GiB, 7948206080 bytes, 15523840 sectors
/dev/mmcbk0p1    2048 15523839 15521792 7.4G 83 Linux
Disk /dev/sda: 596.2 GiB, 640135028736 bytes, 1250263728 sectors
```

Výpis indikuje, že připojený pevný disk je v systému veden jako **/dev/sda**.

Nyní je zřejmé, který disk připravit. Následně je nutné připravit oddíl:

```
wipefs -a /dev/sda# smaže všechny oddíly na disku
fdisk /dev/sda# spustí interaktivní utilitu fdisk na disku /dev/sda
fdisk: n## vytvoří nový oddíl...
fdisk: p## ...typu primární
fdisk: w## zapíše změny na disk
```

Naformátování souborovým systémem ext4 se provede programem **mkfs.ext4**. Následující příkaz zformátuje oddíl se souborovým systémem ext4 bez rezervních bloků (-m 0) a bude optimalizován pro velké soubory (-T largefile).

```
# mkfs.ext4 -m 0 -T largefile /dev/sda
```

Nyní je možné nastavit, aby se disk připojoval po každém startu. Protože pořadí disků se může změnit, pokud je připojeno více disků, není možné se spoléhat pouze na označení **/dev/sda**. Pomocí příkazu **blkid** lze zjistit UUID připojeného disku, respektive jeho oddílu. UUID je k definici disku mnohem vhodnější.

```
# blkid /dev/sda1
```

Výstupem je:

```
/dev/sda1: UUID="<UUID disku>" TYPE="ext4" PARTUUID="<partuuid>"
```

Následuje vytvoření adresáře, kam bude disk připojen, úprava oprávnění a přidání nového řádku do souboru **/etc/fstab** pro automatické připojení disku po spuštění.

```
# mkdir /mnt/sata&&chown -R uživatel:skupina /mnt/sata&&  
1 | echo "UUID=<UUID disku> /mnt/sata ext4  
. | nodiratime,noatime,barrier=0,data=writeback,commit=360 0 1" >> /etc/fstab
```

Nyní je vhodné ještě zkonfigurovat disk po hardwarové stránce. To má na starosti příkaz **hdparm**. Parametr „-q“ značí zpracování následujícího příkazu v tichosti, „-a“ velikost čtení diskem dopředu a „-S“ značí odpočet, po kterém bude disk uspán, nebyl-li po specifikované době použit.

```
# hdparm -q -a 2048 -q -S120 /dev/disk/by-uuid/<UUID disku>
```

Protože se nastavení po restartu ztratí, je nutné příkaz spouštět při každém startu. Toho se dá jednoduše docílit přidáním příkazu do **/etc/rc.local** na samostatný řádek před „exit 0“.

Pomocí příkazu **hdparm** je možné vyzkoušet přibližnou přenosovou rychlost disku (-t) a ujistit se tak, že nenastala chyba v automatické konfiguraci jádrem.

```
# hdparm -t /dev/sda
```

Výstup zhruba odpovídá reálným vlastnostem disku:

```
/dev/sda:  
Timingbuffered disk reads: 258 MB in 3.00seconds = 85.93 MB/sec
```

### 5.3.2 Konfigurace SSH serveru

Konfigurace SSH serveru probíhá v souboru `sshd_config`:

```
# nano/etc/ssh/sshd_config
```

Ve výchozím nastavení je použit port 22. To je také port, kam se potencionální útočník pokusí podívat jako první a mohl by se přinejmenším pokusit o útok hrubou silou. Port je proto v konfiguračním souboru změnit na jiný, například 54538:

```
5| Port 54538
```

Je vhodné také zakázat SSH přihlášení uživatele `root`, protože pod ním se potenciální útočník může pokoušet přihlásit.

To se provede úpravou parametru `PermitRootLogin` z „yes“ na „no“:

```
28| PermitRootLogin no
```

Parametrem `X11Forwarding` lze zakázat směrování X11 pro zvýšení bezpečnosti.

```
64| X11Forwarding no
```

Dále je vhodné omezit přihlášení pomocí `ssh` pouze pro uživatele, kteří to potřebují. Toho lze docílit přidáním řádku s povolenými uživateli:

```
AllowUsers<uživatelská jména>
```

Všechna nastavení jsou použita i pro SFTP server.

Přenos komprimovaného archivu přes gigabitovou síť (pomocí SFTP) do adresáře `/tmp`, který je připojen v paměti RAM, probíhá maximálně rychlostí 12.6 MiB/s.

Je vhodné doporučit silnější více faktorové zabezpečení SSH serveru. [36] [57] [58]

### 5.3.3 Konfigurace pro FTPd serveru

Protože šifrování přenosu velkého objemu dat přes SFTP je pro server obrovskou zátěží omezující přenosovou rychlost u lokálních přenosů, je možné pro přenos dat po lokální síti použít anonymní nešifrované FTP a to vhodně konfigurovat.

Instalace FTP serveru:

```
# apt-get install proftpd
```

Konfigurace probíhá v souboru `/etc/proftpd/proftpd.conf`. Zde je vhodné pro vyšší bezpečnost změnit port:

```
Port 45125
```

Protože celé spojení je nezašifrované, není vhodné, aby se uživatelé přihlašovali svými údaji k běžnému účtu, a aby mohl připojený uživatel jinam, než jen do specifického umístění.

#### 5.3.3.1 Zabránění přihlašování běžného uživatele

Na konec konfiguračního souboru se pro zabránění přihlášení uživatelovými údaji vloží následující kód:

```
<Limit LOGIN>  
DenyAll  
</Limit>
```

#### 5.3.3.2 Povolení anonymního uživatele

Nyní je zapotřebí povolit anonymní účet.

Ten je ve výchozím konfiguračním souboru předpřipraven – je zapotřebí pouze umazat jeden `#` z každého řádku oddílu `<Anonymous~ > ... </Anonymous>`.

#### 5.3.3.3 Uzamknutí anonymního uživatele v definovaném adresáři

Pro uzamknutí anonymního uživatele ve chtěném adresáři je nutné změnit `~`, definovanou na začátku oddílu v tagu `<Anonymous~>`.

Anonymnímu uživateli je cíleno umožnit přístup na disk připojený k /mnt/sata do adresáře public:

```
<Anonymous/mnt/sata/public>
...
</Anonymous>
```

#### 5.3.3.4 Povolení přihlášení jako anonymní uživatel

Protože je nyní zakázáno jakékoli přihlášení (bod 1), je nutné specificky v oddílu <Anonymous></Anonymous> přihlášení povolit:

```
<Anonymous/mnt/sata/public>

<Limit LOGIN>
AllowAll
</Limit>
...

</Anonymous>
```

Tímto nastavením je docíleno nemožnosti přihlášení běžného uživatele k nešifrovanému FTP, jiného než výchozího portu, povolení anonymního uživatele, uzamknutí anonymního uživatele na místo, kde jsou uložena pouze veřejná data a možnost posílat po lokální síti na server data velmi rychle.

Přenos komprimovaného archivu přes gigabitovou síť (pomocí nešifrovaného FTP) do adresáře /tmp, který je připojen v paměti RAM, probíhá přibližně rychlostí 80 MiB/s.

#### 5.3.4 Konfigurace RAID pole

Ke konfiguraci softwarového RAID pole lze použít program mdadm.

V této práci je RAID pole použito pro vytvoření rychlého a úsporného paměťového prostoru pro dočasná data, která jsou postradatelná – existuje pro ně duplikát. Toto pole má při provozu menší spotřebu než rotující magnetický disk, a proto je pro svůj účel (rychlého a úsporného paměťového prostoru pro postradatelná dočasná data) výhodnější.

V poli bude použito celkem 8 paměťových úložišť rozdělených do dvou čtveřic. Každá tato čtveřice paměťových úložišť bude připojena na svůj rozbočovač a ten bude připojen

ke sběrnici. Vzhledem k tomu, že každý rozbočovač je připojen k vlastní sběrnici, je možné dosáhnout rychlostí odpovídajícím nejvýše 2xUSB2.0 specifikaci.

Instalace je provedena jednoduchým příkazem:

```
# apt-getinstallmdadm
```

Po připojení první čtveřice je nutné zjistit jejich evidenci v systému:

```
fdisk -l
```

Je zvolen předpoklad, že připojená čtveřice je v systému evidována následovně: /dev/sdb, /dev/sdc, /dev/sdd a /dev/sde.

Nad touto čtveřicí je nutné vytvořit první RAID 0:

```
mdadm --create --verbose /dev/md0 --level=stripe--raid-devices=4 /dev/sdb /dev/sdc /dev/sdd /dev/sde
```

Po připojení druhé čtveřice je zvolen předpoklad, že její prvky vede systém následovně: /dev/sdf, /dev/sdg, /dev/sdh a /dev/sdi.

I nad touto čtveřicí je nutné vytvořit pole RAID 0:

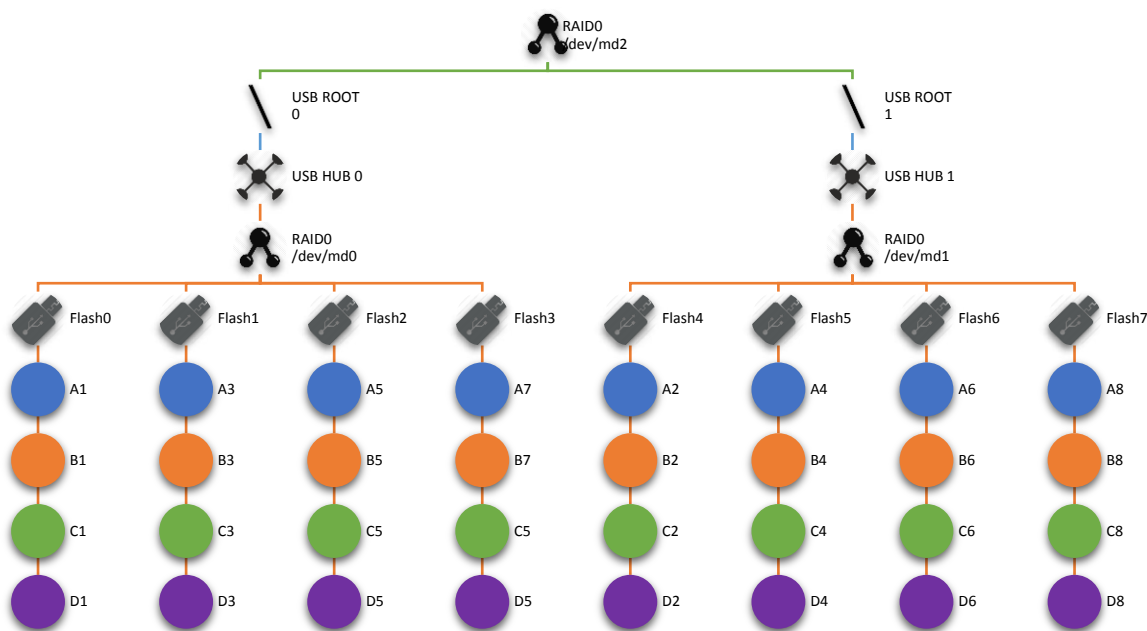
```
mdadm --create --verbose /dev/md1 --level=stripe--raid-devices=4 /dev/sdf /dev/sdg /dev/sdh /dev/sdi
```

Nyní je vytvořena dvojice RAID 0 polí. Ty je pro maximalizaci výkonu možné spojit opět do RAID 0 pole a získat tím tak propustnost dvou linek USB vedených přímo do procesoru dohromady.

```
mdadm --create --verbose /dev/md2 --level=stripe --raid-devices=2 /dev/md0 /dev/md1
```

Aby se disky nastavily i po restartu, je nutné odeslat aktuální konfiguraci do konfiguračního souboru:

```
mdadm --detail --scan>> /etc/mdadm/mdadm.conf
```



Obrázek 3 - RAID pole z 8 úložišť

Pomocí příkazu **hdparm -t** je znovu možné vyzkoušet přibližnou přenosovou rychlost čtení právě vytvořeného pole **/dev/md2**

```
# hdparm -t /dev/md2
```

Výstup:

```
/dev/md2:
```

```
Timingbuffered disk reads: 226 MB in 3.02seconds = 74.94 MB/sec
```

V příloze (**Příloha 2–Testy výkonu raidpole.zip**) jsou zahrnuty rozsáhlé výsledky výkonu raid pole /dev/md1 (tedy pouze jedné větve /dev/md2) v závislosti na použitém plánovači a souborovém systému. Zde uveďme pouze nejnáročnější test, pro další testy a pro detaily je vhodné nahlédnout do přílohy. Test spočívá v souběžném spuštění 4 rozdílných příkazů k zápisu na pole a poté sečtení výsledných hodnot:

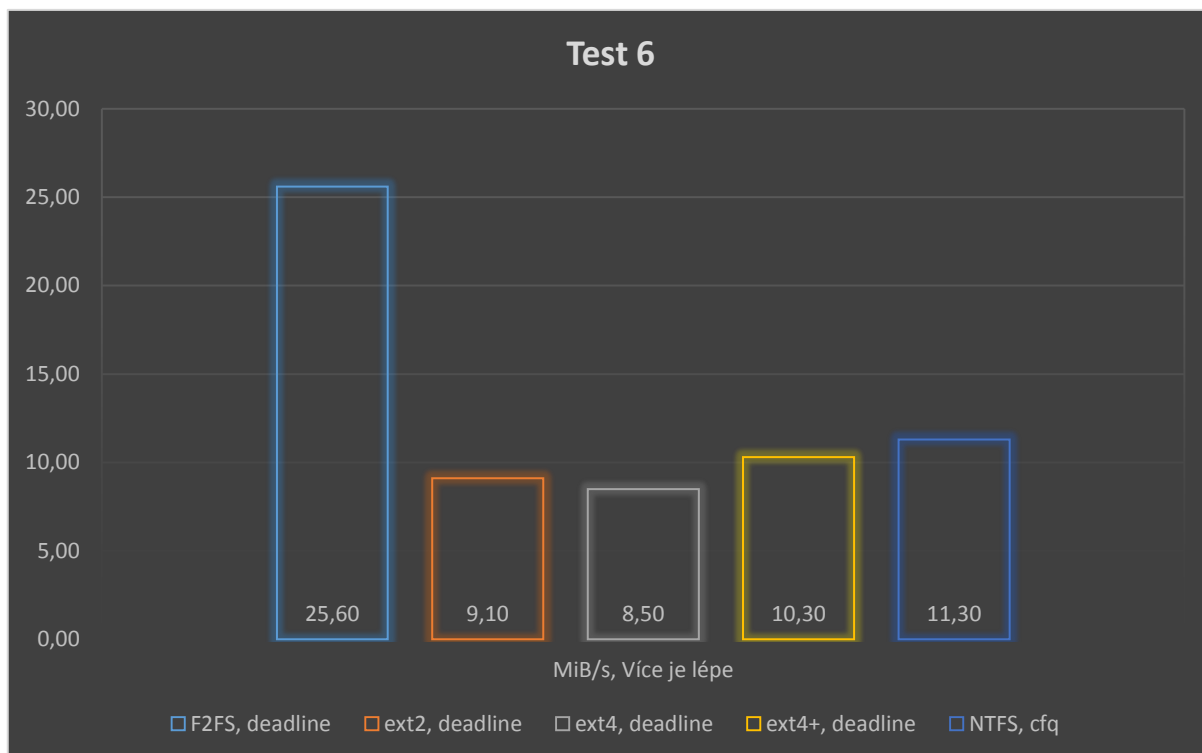
```
ddbs=1M count=2048 if=/dev/zeroof=test1a conv=fdatasync
```

```
ddbs=4k count=500k if=/dev/zeroof=test3a conv=fdatasync
```

```
ddbs=512 count=4M if=/dev/zeroof=test4a conv=fdatasync
```

```
ddbs=2000M count=1 if=/dev/zeroof=test2a conv=fdatasync
```





Obrázek 4 - Test závislosti souborového systému a plánovače na výkon RAID pole

Z grafu je zřejmé, že nejvýkonnější variantou je F2FS společně s deadline plánovačem.

### 5.3.5 Konfigurace webového serveru – Lighttpd, PHP, SQLite a Wordpress

Kvůli nižšímu výkonu BananaPi je vhodnější použít odlehčené aplikace. To platí i pro webový server a služby pod něj spadající. Tato část je tedy zaměřena právě na odlehčené alternativy MySQL a Apache – SQLite a Lighttpd. Všechny funkce je nutné otestovat. K tomu bude využit Wordpress, jelikož jeho nasazení je poměrně časté a je k němu zapotřebí funkční php a databáze.

Instalace programů se provede příkazem apt-get:

```
# apt-getinstalllighttpdphp5-cgisqlite3php5-sqlite
```

Poté je zapotřebí povolit fastcgi a fastcgi-php moduly:

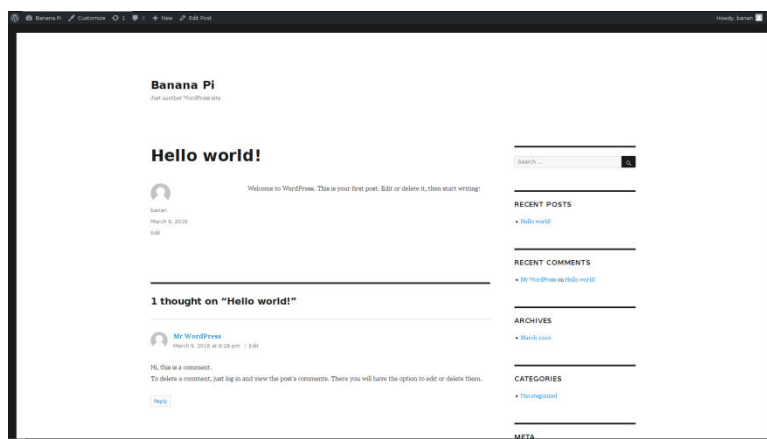
```
# lighty-enable-modfastcgi
# lighty-enable-modfastcgi-php
```

Nyní je vše připraveno pro webové stránky s podporou PHP a databáze.

Wordpress je možné stáhnout z domovské stránky. Aby fungoval s SQLite, je do něj nutné doinstalovat doplněk SQLiteIntegration. Wordpress se pouze rozbalí do adresáře /var/www/html a doplněk SQLiteIntegration se rozbalí do adresáře /var/www/html/wordpress/wp-content/plugins/. Do souboru /var/www/html/wordpress/wp-config.php se umístí následující řádek:

```
define('DB_TYPE', 'sqlite')
```

a vše je připraveno k instalaci pomocí jednoduchého průvodce. Ten je dostupný přes webový prohlížeč na IP adrese BananaPi. Po nastavení základních údajů o webu je zobrazena titulní stránka s titulkem „Hello World!“:



Obrázek 5 - Hello world, Wordpress

### 5.3.6 Konfigurace Samba serveru

Instalace Samba serveru se provede jednoduchým příkazem apt-get:

```
sudoapt-getinstall samba
```

Nyní je nutné Samba server nakonfigurovat. Konfigurace se provádí v konfiguračním souboru `/etc/samba/smb.conf`:

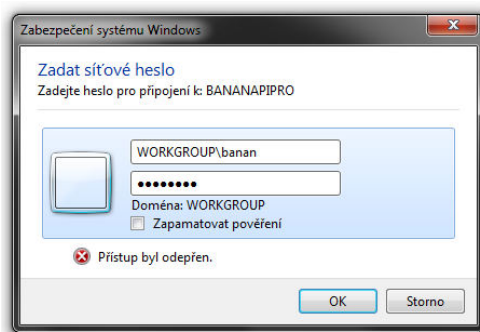
```
workgroup = <workgroup>          # pracovní skupina či doména použitá v síti
wins support = yes                # podpora wins
[bananapi]
comment= BananaPi
path=/mnt/                        # Adresář, který se sdílí
browseable=Yes                   # Povoluje procházení adresářů
writeable=Yes                    # Povoluje zápis do adresářů
public=no                         # následující 4 řádky konfiguruji oprávnění ke
onlyguest=no                     # sdíleným složkám.
createmask=0777                  #
directorymask=0777              #
```

Ke sdíleným souborům pomocí samby se nepřistupuje pomocí běžných přihlašovacích údajů, ale pomocí přihlašovacích údajů samby. Je nutné vytvořit heslo, které se bude používat pro přihlášení:

```
smbpasswd -a <uživatelské jméno>
```

Z operačního systému Windows je pak možné přistoupit k BananaPi běžně pomocí Windows Exploreru po zadání údajů:

```
<workgroup>\<uživatelské jméno>
<heslo zadané pomocí smbpasswd -a>
```



Obrázek 6 - Přihlašovací výzva operačního systému Microsoft Windows

### 5.3.7 Konfigurace P2P klienta

Většina P2P klientů podporuje webové rozhraní. To je ale většinou omezené a neposkytuje dostatečné množství nastavení. Výjimkou je program Deluge, který poskytuje plnohodnotné grafické webové rozhraní. Instalace Deluge a jeho webového rozhraní se provede jednoduchým příkazem apt-get:

```
# apt-get install deluged deluge-web
```

Samotná konfigurace, jako číslo použitého portu, šifrování a další, probíhá v UI ve webovém prohlížeči na portu 8112. Výchozí heslo je *deluge*.

### 5.3.8 Konfigurace VLC

Instalace VLC se provede jednoduchým příkazem apt-get:

```
# apt-get install vlc
```

Program VLC je samozřejmě možné ovládat různými způsoby, jinak by nebyl vhodný pro nasazení na server bez připojeného displeje. Jednou z těchto možností je ovládání programu přes webové rozhraní, resp. přes HTTP.

Nastartování a otestování webového rozhraní se provede následujícím příkazem:

```
# cvlc -I http --http-port <PORT>--http-password=<HESLO>
```

Kde **--http-port PORT** definuje použitý port a **--http-password=<HESLO>** použité heslo.

Heslo a port je lepší definovat v `~/.config/vlc/vlcrc` pomocí parametru `http-password=<HESLO>` a `--http-port PORT`, protože při deklaraci v konzoli zůstanou údaje viditelné (například správcem procesů):

```
Command  
/usr/bin/vlc -I dummy -I http --http-port 43822 --http-password=test  
/usr/bin/vlc -I dummy -I http --http-port 43822 --http-password=test
```

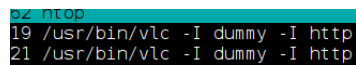
Obrázek 7 - VLC, viditelné heslo a port

Přesunutí definice do konfiguračního souboru `~/.config/vlc/vlcrc`:

```
echo "http-port=port" >>~/.config/vlc/vlcrc  
echo "http-password=heslo" >>~/.config/vlc/vlcrc
```

Nyní již není nutné údaje zadávat a tak nejsou viditelné:

```
cvlc -I http --http-port
```



```
02 ntop  
19 /usr/bin/vlc -I dummy -I http  
21 /usr/bin/vlc -I dummy -I http
```

Obrázek 8 - VLC, neviditelné heslo a port

Pokud na systému běží systemd, je lepší příkaz definovat jako službu:

```
# nano/etc/systemd/system/vlc.service
```

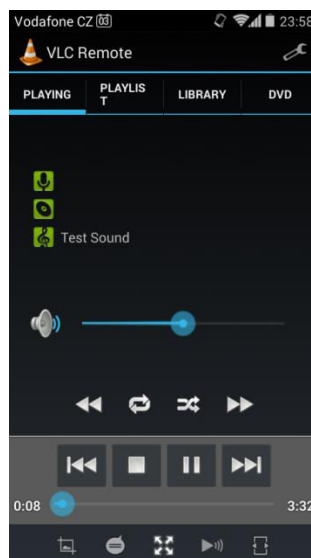
Vytvoření struktury služby:

```
[Unit]  
Description=VideoOnLANService  
After=network.target  
  
[Service]  
Type=forking  
User=banan  
ExecStart=/usr/bin/cvlc -I http  
Restart=on-abort  
  
[Install]  
WantedBy=multi-user.target
```

Povolení služby:

```
# systemctlenable /etc/systemd/system/vlc.service
```

VLC je tedy možné ovládat skrze různé externí aplikace, které s webovým serverem programu VLC komunikují pomocí HTTP příkazů.



Obrázek 9 - Vzdálené ovládání VLC pomocí aplikace VLC Remote

#### 5.4 Srovnání s dalšími možnostmi pořízení serveru

BananaPi M1 společně s příslušenstvím, tedy adaptérem a kabely k napájení 2.5“ SATA disku, vyjdou přibližně na 1 500 Kč. Zařízení umožňuje dobrou a spolehlivou funkci univerzálního síťového zařízení.

Cena pronájmu (virtuálního) serveru sice začíná již od 100 Kč za měsíc, ale je nutné doplácet již za jednotky GB mnohem více peněz. Je to řešení, kterému je v tomto případě lepší se vyhnout – zprvce není k pronajatému serveru na internetu dostatečně rychlé datové připojení (oproti lokálnímu serveru) a za druhé tomuto řešení tato práce vytváří alternativu.

Cena vlastního běžného serveru, který by měl za úkol operace zmíněné v této práci, by byla zbytečně vysoká. Ceny serverů s úsporným procesorem Intel Atom se pohybují okolo 12 tisíc Kč a více. Výjimky existují například v podobě základní desky s integrovaným procesorem JetWay JNF9D-2550, která bez daně stojí 3125 Kč, ale je zapotřebí dokoupit paměť, zdroj a další příslušenství. Cena je tedy při koupi serveru s Intel Atom procesorem několikanásobně vyšší. Nicméně také dostatečně dobře zastoupí funkci univerzálního síťového zařízení s malou spotřebou.

## 6 Závěr

Hlavního cíle této práce bylo dosaženo – byl nalezen jednodeskový počítač vhodně použitelný jako menší úsporný server a tento server nakonfigurovat.

Všechny cílený software (FTP server, webový server, systém řízení báze dat, PHP, SSH/SFTP server, multimediální server, Samba a P2P klient) byly úspěšně nainstalovány, nakonfigurovány a otestovány.

Celková potřeba paměti zařízení při běhu všech serverových aplikací nepřekročila 10% kapacity zařízení, tedy přibližně 100 MB. Největší potřebu paměti měl P2P klient Deluge, napsaný v jazyce Python. P2P klient měl spotřebu 3 %, základní systém 4 % a zbylé 3 % obsadil zbytek serverových aplikací.

Jednodeskový počítač po konfiguraci dosahuje velmi slušných přenosových rychlostí po síti, a je tak možné jej použít nejen jako síťové úložiště do domácností a menších firem.

Zjištěnou alternativou pro Banana Pi M1 jsou jednodeskové počítače s procesorem Intel Atom, které ale nejsou tak cenově dostupné.

## 7 Bibliografie

- [1] Man in the Cloud (MITC) Attacks. In: *Imperva* [online]. Redwood City: Imperva, 2015 [cit. 2016-03-13]. Dostupné z:  
[https://www.imperva.com/docs/HII\\_Man\\_In\\_The\\_Cloud\\_Attacks.pdf](https://www.imperva.com/docs/HII_Man_In_The_Cloud_Attacks.pdf)
- [2] CHARLTON, Alistair. iCloud accounts at risk of brute force attack as hacker exploits 'painfully obvious' password flaw. In: *International Business Times* [online]. London: International Business Times, 2015 [cit. 2016-03-13]. Dostupné z:  
<http://www.ibtimes.co.uk/icloud-accounts-risk-brute-force-attack-hacker-exploits-painfully-obvious-password-flaw-1481623>
- [3] GREENWALD, Glenn. *No place to hide: Edward Snowden, the NSA and the surveillance state*. 1st ed. London: Hamish Hamilton, c2014. ISBN 978-0-241-14669-9.
- [4] HARDING, Luke. *The Snowden Files: The Inside Story of the World's Most Wanted Man*. 1. -: Guardian Faber Publishing, 2014. ISBN 9781783350360.
- [5] R., McGovern. High Tech, Low Legality: The NSA Spies on American Citizens. *High Tech, Low Legality: The NSA Spies on American Citizens*. AMERICAN EDUCATIONAL TRUST, 2006, **2006**(2), 26.
- [6] MACASKILL, Ewen, Julian BORGER, Nick HOPKINS, Nick DAVIES a James BALL. GCHQ taps fibre-optic cables for secret access to world's communications. *The Guardian* [online]. 2013, **2013**(21), 7 [cit. 2016-03-13]. ISSN 0261-3077. Dostupné z: [http://cyber-peace.org/wp-content/uploads/2013/06/GCHQ-taps-fibre-optic-cables-for-secret-access-to-worlds-communications\\_-\\_UK-news\\_-\\_The-Guardian.pdf](http://cyber-peace.org/wp-content/uploads/2013/06/GCHQ-taps-fibre-optic-cables-for-secret-access-to-worlds-communications_-_UK-news_-_The-Guardian.pdf)
- [7] GERRITSEN, Armin. *CISC vs RISC*. CPU Site.(C) Mar, 1999.
- [8] VAJDA, Ferenc. *CISC vs. RISC: Bridging the Processor Architecture - Software Gap*. Budapešť, 1995. Dostupné také z:  
<https://cds.cern.ch/record/399377/files/p275.pdf>



- [9] DANG, Alan a Chris ANGELINI. ARM Vs. x86: The Secret Behind Intel Atom's Efficiency. *Tom's Hardwar* [online]. 2012, **2012**(-), 5 [cit. 2016-03-13]. Dostupné z: <http://www.tomshardware.com/reviews/atom-z2760-power-consumption-arm,3387.html>
- [10] HRUSKA, Joel. The final ISA showdown: Is ARM, x86, or MIPS intrinsically more power efficient?. *Extreme Tech* [online]. 2014, **2014**(-), 3 [cit. 2016-03-13]. Dostupné z: <http://www.extremetech.com/extreme/188396-the-final-isa-showdown-is-arm-x86-or-mips-intrinsically-more-power-efficient>
- [11] About Raspberry Pi Foundation. In: *Raspberry Pi Foundation* [online]. Londýn: Raspberry Pi Foundation, 2015 [cit. 2016-03-13]. Dostupné z: <https://www.raspberrypi.org/about/>
- [12] BUSH, Steve. Dongle computer lets kids discover programming on a TV. *ElectronicsWeekly* [online]. 2011, **2011**(-), 1 [cit. 2016-03-13]. Dostupné z: <http://www.electronicweekly.com/market-sectors/embedded-systems/dongle-computer-lets-kids-discover-programming-on-a-2011-05/>
- [13] *Raspberry Pi Forum* [online]. Londýn: Raspberry Pi Foundation, 2016 [cit. 2016-03-13]. Dostupné z: <https://www.raspberrypi.org/forums/viewforum.php?f=15>
- [14] *Astro Pi* [online]. Londýn: Astro Pi, 2015 [cit. 2016-03-13]. Dostupné z: <https://astro-pi.org>
- [15] Five million sold!. In: *Raspberry Pi Blog* [online]. Londýn: Raspberry Pi Foundation, 2015 [cit. 2016-03-13]. Dostupné z: <https://www.raspberrypi.org/blog/five-million-sold/>
- [16] UPTON, Eben. Raspberry Pi 3 on sale now at \$35. In: *Raspberry Pi Blog* [online]. Londýn: Raspberry Pi Foundation, 2016 [cit. 2016-03-13]. Dostupné z: <https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/>
- [17] Raspberry Pi 1 Model B+. In: *Raspberry Pi Blog* [online]. Londýn: Raspberry Pi Foundation, 2014 [cit. 2016-03-13]. Dostupné z: <https://www.raspberrypi.org/products/model-b-plus/>

- [18] *Raspberry Pi Store* [online]. Londýn: Raspberry Pi Foundation, - [cit. 2016-03-13]. Dostupné z: <http://www.raspberrypi.com/>
- [19] *Orange Pi Web* [online]. Shenzhen City: Xunlong Software CO.,Limited, 2016 [cit. 2016-03-13]. Dostupné z: <http://www.orangepi.org/>
- [20] SHAIKG, Sameer. *Complete Computer Hardware Only*. PediaPress, b.r.. Dostupné také z: <https://books.google.cz/books?id=WSYjdR425p4C>
- [21] Armbian - Banana Pi. *Armbian* [online]. Armbian, 2016 [cit. 2016-03-13]. Dostupné z: <http://www.armbian.com/banana-pi/>
- [22] Banana Pi M1 Specs. *Banana Pi* [online]. b.r. [cit. 2016-03-13]. Dostupné z: <http://www.banana-pi.org/m1.html>
- [23] Banana Pi R1 Specs. *Banana Pi* [online]. b.r. [cit. 2016-03-13]. Dostupné z: <http://www.banana-pi.org/r1.html>
- [24] Banana Pi M2 Specs. *Banana Pi* [online]. Banana Pi, b.r. [cit. 2016-03-13]. Dostupné z: <http://www.banana-pi.org/m2.html>
- [25] Banana Pi M3. *Linux-sunxi Community* [online]. Linux-sunxi Community, 2016 [cit. 2016-03-13]. Dostupné z: [http://linux-sunxi.org/Banana\\_Pi\\_M3](http://linux-sunxi.org/Banana_Pi_M3)
- [26] C1 freezing (too) often. In: *Odroid Forum* [online]. Odroid Forum, 2015 [cit. 2016-03-13]. Dostupné z: <http://forum.odroid.com/viewtopic.php?f=117&t=13216>
- [27] C1 freezes when USB mouse and keyboard disconnected. In: *Odroid Forum* [online]. Odroid Forum, 2015 [cit. 2016-03-13]. Dostupné z: <http://forum.odroid.com/viewtopic.php?f=117&t=13241>
- [28] Odroid C1. *HardKernel* [online]. HardKernel, 2014 [cit. 2016-03-13]. Dostupné z: [http://www.hardkernel.com/main/products/prdt\\_info.php?g\\_code=G141578608433&tab\\_idx=2](http://www.hardkernel.com/main/products/prdt_info.php?g_code=G141578608433&tab_idx=2)
- [29] *Raspbian* [online]. Raspbian, 2016 [cit. 2016-03-13]. Dostupné z: <https://www.raspbian.org/>

- [30] *Armbian* [online]. Armbian, 2016 [cit. 2016-03-13]. Dostupné z: <http://www.armbian.com/>
- [31] FILE TRANSFER PROTOCOL (FTP). *The Internet Engineering Task Force* [online]. The Internet Engineering Task Force, 1985 [cit. 2016-03-13]. Dostupné z: <https://tools.ietf.org/html/rfc959>
- [32] MILLER, Philip. *TCP/IP: the ultimate protocol guide*. Boca Raton, Fla: BrownWalker Press, 2009. ISBN 15-994-2493-2.
- [33] HAUGDAHL, J. *Network analysis and troubleshooting*. Reading, Mass.: Addison-Wesley, c2000. ISBN 02-014-3319-2.
- [34] The Secure Shell (SSH) Connection Protocol. *The Internet Engineering Task Force* [online]. The Internet Engineering Task Force, 2006 [cit. 2016-03-13]. Dostupné z: <https://www.ietf.org/rfc/rfc4254.txt>
- [35] BARRETT, Daniel J a Richard E SILVERMAN. *SSH, the secure shell: the definitive guide*. 1st ed. Cambridge [Mass.]: O'Reilly, 2001. ISBN 0596000111.
- [36] DWIVEDI, Himanshu. *Implementing SSH. Strategies for Optimizing the Secure Shell*. Hoboken: John Wiley & Sons, 2003. ISBN 9780764557255.
- [37] RODERICK W. SMITH., *Linux Samba Server Administration Craig Hunt Linux Library*. Hoboken: John Wiley & Sons, 2006. ISBN 9780782153132.
- [38] TS, Jay, Robert ECKSTEIN a David COLLIER-BROWN. *Samba pocket reference*. 2nd ed. Sebastopol, CA: O'Reilly, c2003. ISBN 0596005466.
- [39] Just what is SMB?. *Samba - opening Windows* [online]. Samba - opening Windows, 2002 [cit. 2016-03-13]. Dostupné z: <https://www.samba.org/cifs/docs/what-is-smb.html>
- [40] *Setting up LAMP Getting Linux, Apache, MySQL, and PHP Working Together*. Hoboken: John Wiley & Sons, 2006. ISBN 9780782151121.
- [41] PHP Documentation. *PHP* [online]. PHP, b.r. [cit. 2016-03-13]. Dostupné z: <http://php.net/manual/en/langref.php>

- [42] DBMS (Database Management System) - systém řízení báze dat. *Managment Mania* [online]. Praha: Managment Mania, 2012 [cit. 2016-03-13]. Dostupné z: <https://managementmania.com/cs/dbms-database-management-system-system-rizeni-baze-dat>
- [43] VLC media player. *VideoLAN* [online]. VideoLAN, 2014 [cit. 2016-03-13]. Dostupné z: [https://wiki.videolan.org/VLC\\_media\\_player/](https://wiki.videolan.org/VLC_media_player/)
- [44] VLC media player: Documentation:Modules/http intf. *VideoLAN* [online]. VideoLAN, 2013 [cit. 2016-03-13]. Dostupné z: [https://wiki.videolan.org/Documentation:Modules/http\\_intf/](https://wiki.videolan.org/Documentation:Modules/http_intf/)
- [45] Google Play - Remote for VLC (Fork). *Google Play* [online]. Google Play, 2014 [cit. 2016-03-13]. Dostupné z: <https://play.google.com/store/apps/details?id=uk.co.samicemalone.client.android.vlc.remote&hl=cs>
- [46] NOVOTNÝ, Miroslav. KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ. *Peer-to-Peer síť*. Praha, 2009. Dostupné také z: [www.ksi.mff.cuni.cz/lectures/NSWI035/DHT.pdf](http://www.ksi.mff.cuni.cz/lectures/NSWI035/DHT.pdf)
- [47] REESE, Will. Increase performance, reliability and capacity with software RAID. *Linux Journal*. Linux Journal, 2008, **2008**(175).
- [48] RAID setup. *Linux Kernel* [online]. Linux Kernel, 2013 [cit. 2016-03-13]. Dostupné z: [https://raid.wiki.kernel.org/index.php/RAID\\_setup](https://raid.wiki.kernel.org/index.php/RAID_setup)
- [49] DDF Fake RAID. *Linux Kernel* [online]. Linux Kernel, 2013 [cit. 2016-03-13]. Dostupné z: [https://raid.wiki.kernel.org/index.php/DDF\\_Fake\\_RAID](https://raid.wiki.kernel.org/index.php/DDF_Fake_RAID)
- [50] RAID (redundant array of independent disks). *TechTarget* [online]. TechTarget, 2015, **2015**, 1 [cit. 2016-03-13]. Dostupné z: <http://searchstorage.techtarget.com/definition/RAID>
- [51] I/O Schedulers and I/O Performance. In: *Safari* [online]. Seabstopol: Safari, b.r. [cit. 2016-03-13]. Dostupné z: <https://www.safaribooksonline.com/library/view/linux-system-programming/0596009585/ch04s06.html>

- [52] Choosing an I/O Scheduler for Red Hat® Enterprise Linux® 4 and the 2.6 Kernel. In: *Ret Hat* [online]. Westford: Ret Hat, 2005 [cit. 2016-03-13]. Dostupné z: <http://www.redhat.com/magazine/008jun05/features/schedulers/>
- [53] LINUX KERNEL ORGANIZATION. *The Second Extended Filesystem Documentation*. San Jose, 1993. Dostupné také z: <https://www.kernel.org/doc/Documentation/filesystems/ext2.txt>
- [54] LINUX KERNEL ORGANIZATION. *Ext4 Filesystem Documentation*. San Jose, 2008. Dostupné také z: <https://www.kernel.org/doc/Documentation/filesystems/ext4.txt>
- [55] LINUX KERNEL ORGANIZATION. *WHAT IS Flash-Friendly File System (F2FS)*. San Jose, 2013. Dostupné také z: <https://www.kernel.org/doc/Documentation/filesystems/f2fs.txt>
- [56] TUXERA INC. *Ntfs-3g: Third Generation Read/Write NTFS Driver*. San Jose, 2006. Dostupné také z: <http://www.tuxera.com/community/ntfs-3g-manual/>
- [57] CHRISTOPHER NEGUS, François Caen. *SUSE Linux Toolbox 1000+ Commands for openSUSE and SUSE Linux Enterprise*. Hoboken: John Wiley & Sons, 2008. ISBN 9780470283004.
- [58] HOFFMAN, Chris. How to Secure SSH with Google Authenticator's Two-Factor Authentication. *How To Geek* [online]. 2012, **2012**, 1 [cit. 2016-03-13]. Dostupné z: <http://www.howtogeek.com/121650/how-to-secure-ssh-with-google-authenticators-two-factor-authentication/>

## **8 Přílohy**

Přílohy jsou dostupné na optickém médiu, které je přiložené v deskách této práce.

**Příloha 1 – Tabulka porovnání vlastností jednodeskových počítačů.xlsx**

**Příloha 2 – Testy výkonu raid pole.xlsx**

**Příloha 3 – Konfigurační soubory.zip**