**STŘEDNÍ PRŮMYSLOVÁ ŠKOLA BRNO, PurkYŇOVA, příspěvková Organizace**



Realizace webhostingu na linuxové distribuci

Karel Bašta

V4D

**Profilová část maturitní zkoušky**

**MATURITNÍ PRÁCE**

**BRNO 2022**

# Prohlášení o autorství práce

Prohlašuji, že jsem maturitní práci Realizace webhostingu na linuxové distribuci vypracoval samostatně a použil jen zdroje uvedené v seznamu literatury.

Prohlašuji, že:

* Beru na vědomí, že zpráva o řešení maturitní práce a základní dokumentace k aplikaci bude uložena v elektronické podobě na intranetu SPŠ Brno, Purkyňova, příspěvková organizace.
* Beru na vědomí, že bude má maturitní práce včetně zdrojových kódů uložena v knihovně SPŠ Brno, Purkyňova, příspěvková organizace, dostupná k prezenčnímu nahlédnutí. Škola zajistí, že nebude pro nikoho možné pořizovat kopie jakékoliv části práce.
* Beru na vědomí, že SPŠ Brno, Purkyňova, příspěvková organizace, má právo celou moji práci použít k výukovým účelům a po mém souhlasu nevýdělečně moji práci užít ke své vnitřní potřebě.
* Beru na vědomí, že pokud je součástí mojí práce jakýkoliv softwarový produkt, považují se za součást práce i zdrojové kódy, které jsou předmětem maturitní práce, případně soubory, ze kterých se práce skládá. Součástí práce není cizí ani vlastní software, který je pouze využíván za přesně definovaných podmínek, a není podstatou maturitní práce.

Karel Bašta

Pod Nemocnicí 585/37, Brno, 625 00

V Brně dne 17. 4. 2022 .........................................................

Vedoucí práce: Mgr. Bc. Leo Nitče

Odborný konzultant: Petr Čížek

# Anotace

Tato maturitní práce se zabývá vytvořením virtuálního počítače určeného pro hosting webových stránek a aplikací. Přístup k počítači je na dálku umožněn pomocí příkazové řádky dle bezpečnostní politiky školy. Počítač umožňuje přístup k webovým stránkám a aplikacím pomocí internetového prohlížeče a poskytuje jim přístup k databázovým prostředkům. Počítač je nakonfigurován pro minimální potřebu údržby a pro možnost rozšíření poskytovaných služeb za pomocí šablon.

# Obsah

[Prohlášení o autorství práce 2](#_Toc101211087)

[Anotace 4](#_Toc101211088)

[Obsah 5](#_Toc101211089)

[Teoretický úvod 8](#_Toc101211090)

[Seznam použitých zkratek 8](#_Toc101211091)

[1 Rozbor řešení 9](#_Toc101211092)

[1.1 Použité technologie 9](#_Toc101211093)

[1.1.1 VMware Workstation player 9](#_Toc101211094)

[1.1.2 Linuxová distribuce Debian 9](#_Toc101211095)

[1.1.3 Balíčkový manažer apt 9](#_Toc101211096)

[1.1.4 Balíčkový manažer npm 9](#_Toc101211097)

[1.1.5 Démon systemd 10](#_Toc101211098)

[1.1.6 UFW 10](#_Toc101211099)

[1.1.7 Fail2ban 10](#_Toc101211100)

[1.1.8 SSH 10](#_Toc101211101)

[1.1.9 Vim 10](#_Toc101211102)

[1.1.10 Git 11](#_Toc101211103)

[1.1.11 NGINX 11](#_Toc101211104)

[1.1.12 MariaDB 11](#_Toc101211105)

[1.1.13 PHP 11](#_Toc101211106)

[1.1.14 Cockpit 11](#_Toc101211107)

[1.1.15 Nextcloud 12](#_Toc101211108)

[1.1.16 PCMS 12](#_Toc101211109)

[1.2 Stanovené cíle 13](#_Toc101211110)

[1.2.1 Smysl počítače 13](#_Toc101211111)

[1.2.2 Zabezpečení a stabilita počítače 13](#_Toc101211112)

[1.2.3 Náročnost na údržbu 13](#_Toc101211113)

[1.2.4 Instalace webového serveru a aplikací 14](#_Toc101211114)

[1.2.5 Zabezpečené připojení ke klientům 14](#_Toc101211115)

[2 Realizace 15](#_Toc101211116)

[2.1 Úvod do instalace 15](#_Toc101211117)

[2.2 Instalace Debianu 16](#_Toc101211118)

[2.2.1 Instalační proměnné 16](#_Toc101211119)

[2.2.2 Rozdělení disku 16](#_Toc101211120)

[2.2.3 Nastavení správce balíků 16](#_Toc101211121)

[2.2.4 Nastavení zavaděče GRUB 16](#_Toc101211122)

[2.2.5 Dokončení instalace 17](#_Toc101211123)

[2.3 Vnitřní zabezpečení 17](#_Toc101211124)

[2.3.1 Zabezpečení uživatelů 17](#_Toc101211125)

[2.4 Softwarové zabezpečení počítače 17](#_Toc101211126)

[2.4.1 Instalace a nastavení firewallu 17](#_Toc101211127)

[2.4.2 Instalace Fail2ban 18](#_Toc101211128)

[2.5 Nastavení údržby 19](#_Toc101211129)

[2.5.1 Instalace textového editoru 19](#_Toc101211130)

[2.5.2 Nastavení automatických aktualizací 19](#_Toc101211131)

[2.5.3 Nastavení vzdáleného přístupu 19](#_Toc101211132)

[2.6 Instalace NGINX 21](#_Toc101211133)

[2.7 Instalace PHP 24](#_Toc101211134)

[2.8 Instalace databázového serveru 25](#_Toc101211135)

[2.9 Instalace Node.js 26](#_Toc101211136)

[2.10 Realizace statických stránek 27](#_Toc101211137)

[2.10.1 Výchozí stránka 27](#_Toc101211138)

[2.10.2 Repositář materiálů pro studenty 27](#_Toc101211139)

[2.11 Instalace webových aplikací 28](#_Toc101211140)

[2.11.1 Cockpit 28](#_Toc101211141)

[2.11.2 Nextcloud 28](#_Toc101211142)

[2.11.3 PCMS 32](#_Toc101211143)

[2.11.4 Tvorba databáze 32](#_Toc101211144)

[2.12 Zavedení webových stránek 34](#_Toc101211145)

[2.12.1 Zabezpečení přístupu k určitým webovým stránkám 34](#_Toc101211146)

[2.12.2 Zavedení virtuálního hosta do NGINX 34](#_Toc101211147)

[2.12.3 Povolení HTTP a HTTPS ve firewall 34](#_Toc101211148)

[Závěr 35](#_Toc101211149)

[Seznam ilustrací 36](#_Toc101211150)

[Seznam tabulek 36](#_Toc101211151)

[Zdroje 36](#_Toc101211152)

[Přílohy 37](#_Toc101211153)

# Teoretický úvod

Hostování internetových stránek a aplikací z virtuálního počítače má spoustu výhod mezi které patří hlavně možnost rychlého přesunu počítače z jedné pracovní stanice na druhou a možnost zálohování funkčních verzí počítače při případném vzniku chyby.

# Seznam použitých zkratek

DVD – Digital Video Disc

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

IP – Internet Protocol

UFW – Uncomplicated FireWall

HTTP(S) – Hypertext Transfer Protocol (Secure)

SSL – Secure Sockets Layer

JS – JavaScript

PHP – PHP: Hypertext Preprocesor

JSON – JavaScript Object Notation

XML – eXtensible Markup Language

DB – DataBase

# Rozbor řešení

## Použité technologie

### VMware Workstation player

Program, ve kterém byl samotný virtuální počítač vytvořen. Jedná se o program umožňující zadarmo spustit a spravovat jeden virtuální počítač v jeden čas. Volba tohoto programu byla doporučena školou pro možnost jednoduchého přenosu počítače do školní sítě.

### Linuxová distribuce Debian

Debian je svobodný operační systém vyvinutý stejnojmennou společností používající jádro Linux. Jelikož velká část základních nástrojů, které spolu s jádrem vytvářejí kompletní operační systém, pochází z projektu GNU, nazýváme tento systém GNU/Linux. Tato distribuce byla vybrána kvůli její přizpůsobitelnosti, jednoduchosti a stabilitě, kterou nám nabízí zcela zdarma. Zároveň se jedná o jednu z nejpoužívanějších distribucí pro naše účely.

### Balíčkový manažer apt

Rychlá, praktická a výkonná metoda instalace balíčků. Jedná se o systém správy balíčků vyvinutý pro GNU/Linux společností Debian. Jeho přítomnost nám velmi usnadňuje život v linuxovém prostředí. Předinstalovaný v operačním systému. Využíváme ho hlavně díky možnosti zavedení vlastních repositářů, které budeme potřebovat pro instalaci dalších programů.

### Balíčkový manažer npm

Npm umožňuje instalovat a spravovat balíčky pro použití v aplikacích node.js. Jedná se o nejrozšířenější balíčkový manažer pro node, tudíž je pro nás práce s ním nejjednodušší. Většina projektů a návodů v node.js jsou tvořeny právě s npm jako hlavním prostředkem pro získávání potřebných knihoven.

### Démon systemd

Jedná se o kořenový proces, který zavádí a spravuje veškeré procesy na pozadí. Umožňuje nám tvořit vlastní takové procesy pro např. webové aplikace. Při startu počítače se tento proces spouští jako první a až tento proces zavádí dle konfigurace procesy další. Předinstalovaný v operačním systému.

### UFW

Velmi užitečný program pro vytváření pravidel pro komunikaci po síti. Umožňuje nám povolit a zakázat komunikaci pro každý port zvlášť jednoduchou syntaxí. Všechna tato pravidla jsou převedena do integrované tabulky pro komunikaci, podle které se Debian řídí. Tento program nám slouží jako alternativa pro výchozí způsob úpravy těchto tabulek hlavně díky jeho jednoduché syntaxi.

### Fail2ban

Program pro zabezpečení možných útoků na počítač. Program kontroluje pokusy o přihlašování zvenku a po velkém množství neúspěšných pokusů za určitý čas zablokuje všechny žádosti ze zdrojové IP adresy. Jedná se o zabezpečení hlavně pro vzdálený přístup k počítači.

### SSH

Aplikace zajišťující vzdálený přístupu k počítači po síti. Umožňuje zabezpečený přístup k příkazové řádce a přenos souborů. Aplikace využívá velmi používaný SSH protokol, který umožňuje i případný přenos souborů.

### Vim

Textový editor používaný pro úpravu konfiguračních souborů. Volba tohoto editoru byla čistě osobní preference, jelikož textové editory pro příkazový řádek nemají v základu moc prostoru pro velké funkční rozdíly.

### Git

Program pro správu verzí souborů v repositáři. V našem případě bude program využit pouze pro získávání projektů z internetu, jelikož se jedná o jediný způsob šíření souborů z takových repositářů na portálech github.com či bitbucket.com

### NGINX

Webový a reverzní proxy server pracující s protokoly HTTP(S) & SSL. Jedná se o hlavní část tohoto projektu. Umožňuje nám asynchronně poskytnout zabezpečené připojení k aplikacím a distribuovat statický obsah. Volba této technologie oproti konkurenční byla hlavně díky jejímu vysokému výkonu za nízkou cenu náročnosti na paměť a jednoduchosti konfigurace a instalace.

### MariaDB

Databázové řešení pro veškeré potřeby procesů a aplikací. Byl vybrán z licenčních důvodů, jednoduchosti instalace a kompatibility s užitými programy.

### PHP

Programovací jazyk určený k tvorbě webových aplikací. V projektu je využité rozhraní FastCGI pro generování dynamických stránek využité aplikací Nextcloud.

### Cockpit

Cockpit je webové grafické rozhraní určené pro správu serveru. Toto rozhraní umožňuje mnohem jednodušší konfiguraci procesů, kontejnerů, úložiště a pro zobrazení logů aplikací a systému samotného. Toto rozhraní bylo instalováno pro použití při údržbě serveru a mělo by být používáno jako hlavní způsob jeho správy. Vzdálený přístup by měl být až alternativou pro tento způsob.

### Nextcloud

Jedná se škálovatelné cloudové řešení zajišťující úložný prostor žákům s možností rozšíření funkcí. Při řešení cloudového úložného prostoru byla tato platforma mojí první volbou díky mojí zkušenosti s ní. Je jednoduše konfigurovatelné z webového prostředí a zároveň pomáhá při konfiguraci a aktualizaci. Zároveň obsahuje možnost konfigurace z příkazové řádky. Na internetu poskytuje nádherně zpracovanou dokumentaci, díky které není problém vyřešit jakýkoli požadavek.

### PCMS

PCMS (Purkyňka CMS) je systém pro správu obsahu webu Střední školy Purkyňovy zaměřený na rychlost, bezpečnost a spolehlivost za pomoci moderních webových technologií.

## Stanovené cíle

### Smysl počítače

Hlavním úkolem výsledku této práce je zajistit možnost bezpečného, výkonného a jednoduchého hostování webových stránek sestrojených v jakémkoli prostředí od statických HTML stránek po nejmodernější aplikace sestrojené v Node.js. Pro možnost využití tohoto počítače je přiložen manuál pro administrátory popisující postupy pro nasazení takových aplikací.

### Zabezpečení a stabilita počítače

Veškerá síťová komunikace s tímto počítačem a webové stránky takové by měli podléhat nějakým bezpečnostním návykům. Hlavními bezpečnostní prvky jsou využití šifrovaného spojení za pomocí SSL certifikátů a HTTP hlavičky popisující, jak nakládat s přenesenými daty. Dále musí být zabezpečený počítač samotný proti útokům z vnější sítě. Důležitým parametrem počítače také bude jeho stabilita v případě problémů se zatížením, výpadkům elektřiny či internetu.

### Náročnost na údržbu

Počítač by neměl mít potřebu každodenní přítomnosti administrátora. Měl by být schopen autonomní operace a provést základní údržbu sám jako například automatické stahování bezpečnostních aktualizací pro balíčky. Měl by být sám schopný udržet funkční stav všech aplikací a případné problémy zaznamenat do předem určených logů. Balíčky poskytnuté při předání virtuálního počítače by měli vystačit pro bezpečný chod všech aplikací a při minimální údržbě být schopné správně pracovat nejméně do skončení podpory jednotlivých balíčků od jejich vydavatelů. Podpora použité verze Debianu je podle plánů vydavatelů do roku 2026, což neplatí ovšem pro všechny nainstalované balíčky. Z tohoto důvody byly instalovány přednostně verze balíčků s dlouhou dobou bezpečnostních aktualizací. Tato doba bude zmíněna u každé z instalovaných aplikací.

### Instalace webového serveru a aplikací

Instalace webových aplikací byla provedena dle instrukcí v dokumentaci samotných aplikací. Toto neplatí pro samotný webový server NGINX, jelikož ten byl přímo kompilován na počítači z důvodu integrace vlastních modulů pro poskytnutí například možnosti komunikace po protokolu h3, který tato aplikace v základu nenabízí. Dodržené postupy budou popsány v sekci realizace pro jednotlivé aplikace.

### Zabezpečené připojení ke klientům

Zabezpečení připojení ke klientům bude mít na starosti webový server NGINX, který by za pomoci školou poskytnutých SSL certifikátů měl zaručit šifrovaný přenos dat na dostatečné úrovni. Dále by měl zaručit přenos bezpečnostních hlaviček pro správné naložení s přenesenými daty a automaticky přesměrovat veškeré pokusy o komunikaci pomocí nezabezpečeného protokolu HTTP na zabezpečený protokol HTTPS. Síla této bezpečnosti by měla být na úrovni „A+“ dle měření pomocí testu na stránkách [https://www.ssllabs.com/](https://www.ssllabs.com/ssltest).

# Realizace

## Úvod do instalace

K instalaci byl stažen výchozí instalační obraz ze stránek [https://www.debian.org](https://www.debian.org/download), která v době instalace obsahovala internetovou instalaci Debianu „bullseye“ verze 11.2.0 pro počítače s architekturou amd64. Jednalo se o nejnovější vydanou verzi této linuxové distribuce, která měla pouze odhadovaný termín konce podpory roku 2026.

Tato distribuce byla zvolena oproti distribucím Ubuntu, CentOS a Fedora díky atraktivnímu balíčkovému manažerovi apt a minimálním systémovým požadavkům. Díky minimálnímu množství předinstalovaných balíčků a „bloatware“ bylo možné nainstalovat pouze balíčky potřebné a nezatěžovat úložiště balíčky nepoužitými.

Tento instalační obraz byl vložen při vytváření nového virtuálního počítače do CD/DVD mechaniky v programu VMware. V programu byly dále vybrány možnosti instalace operačního systému Linux verze Debian 10.x 64-bit, jelikož současná verze VMware ještě neměla nativní podporu Debianu verze 11. Dále byl počítači vytvořen počítači virtuální disk o velikosti 20 GB jako jeden soubor. V sekci přizpůsobení hardwaru počítače bylo vyjmuto vše kromě paměti, procesoru, CD/DVD mechaniky s instalačním obrazem, síťovou kartou v konfiguraci síťového mostu a display ve výchozím nastavení. Pro vysoký výkon při instalaci byla nastavena operační paměť počítače na 4 GB a byly přiděleny 4 jádra procesoru. Tento výkon bude možno omezit dle potřebného výkonu a počtu klientů.

## Instalace Debianu

### Instalační proměnné

K instalaci samotného linuxu bylo využito grafického prostředí, který stažený instalační obraz nabídl po zapnutí virtuálního počítače. Jako jazyk použitý při instalaci byla vybrána čeština, umístění Česko a česká klávesnice. Po dokončení automatického nastavení sítě bylo počítači přiděleno jméno „Jeff“ a doménové jméno „sspbrno.cz“.

Při dotazu na zadání hesla pro uživatele root nebylo zadáno žádné heslo a krok byl přeskočen, čímž byl uživatel root uzamčen a automaticky byla přidělena dalšímu vytvořenému uživateli práva na použití příkazu „sudo“ pro využití k administraci systému. Tento příkaz dočasně dává uživateli zvýšená práva pro vykonávání příkazů. Ty by mohli ve špatných rukou způsobit velké škody, tudíž je důležité uživatelskou skupinu „sudo“ přidat pouze administrátorskému účtu a tomu nastavit silné heslo.

Jméno nového uživatele bylo nastaveno na „Administrátor“ a uživatelské jméno na „administrator“. Poté bylo nastaveno heslo „NU32fEyzTCjsjZvC“, kterému byla nastavena nutná změna při předání počítače.

### Rozdělení disku

Po nastavení hesla uživatele byla zvolena možnost asistovaného rozdělení disku bez jakéhokoli použití technologie LVM. Rozdělení bylo provedeno pro VMware disk a s možností všech souborů v jedné oblasti. Rozdělení bylo potvrzeno a poté začala automatická instalace systému.

### Nastavení správce balíků

Po instalaci systému byla zvolena možnost použití síťového zrcadla pro správce balíků. Bylo nastaveno pro české zrcadlo „deb.debian.org“ bez proxy.

### Nastavení zavaděče GRUB

Po nastavení obrazu pro správce balíků byl nainstalován zavaděč GRUB na primární disk /dev/sda.

### Dokončení instalace

Po ukončení instalace zavaděče byl počítač restartován a byla vyjmuta CD/DVD mechanika z počítače.

## Vnitřní zabezpečení

### Zabezpečení uživatelů

Pro předejití situace, kde by byl přístupný terminál přihlášený na účtu root bylo výchozí textové rozhraní mezi systémem a uživatelem neboli shell změněno na cestu /usr/sbin/nologin, což zamezilo jakýkoli přístup na tento účet. K tomu byl použitý příkaz:

sudo usermod -s /usr/sbin/nologin root

Tímto byla odstraněna možnost se na tento účet přihlásit například pomocí příkazu: „sudo su“. Takový pokus by nyní skončil v chybové hlášce: „Účet je momentálně nedostupný.“

## Softwarové zabezpečení počítače

Před instalací jakéhokoli balíčku byla aktualizována databáze dostupných balíčků pomocí příkazu:

sudo apt update

Poté byly nainstalovány všechny dostupné aktualizace pomocí příkazu:

sudo apt upgrade -y

Příznak „-y“ přeskočil potřebu potvrzení aktualizace.

### Instalace a nastavení firewallu

Pomocí balíčkového manažera apt byl nainstalován balíček ufw a následně byl firewall aktivován pomocí následujících příkazů:

sudo apt install ufw

sudo ufw enable

### Instalace Fail2ban

Pro zamezení „brute-force“ útoku na vzdálený přistup byl nainstalován balíček fail2ban pomocí balíčkového manažera apt. Po instalaci byl balíček již automaticky nakonfigurován pro dostatečnou ochranu. V této konfiguraci po třech neúspěšných pokusech o přihlášení pomocí vzdáleného přístupu budou zamítnuty všechny požadavky o další přihlášení ze zdrojové IP požadavku zařízení po dobu 10 minut. Instalace proběhla pomocí následujícího příkazu:

sudo apt install fail2ban

## Nastavení údržby

### Instalace textového editoru

Pro účely modifikace konfiguračních souborů byl čistě kvůli osobní preferenci pomocí balíčkového manažera apt nainstalován balíček vim.

sudo apt install vim

### Nastavení automatických aktualizací

Balíček „unattended-upgrades“ je velkým pomocníkem pro údržbu zařízení. Tento balíček dle nastavené periody pravidelně aktualizuje databázi dostupných balíčků a samostatně provádí bezpečnostní aktualizace. Aktualizace, které by mohli mít dopad na funkčnost balíčků samostatně neprovádí, ale upozorní na ně při přihlášení. Tento balíček byl instalován pomocí balíčkového správce apt. Balíček je plně funkční již po samotné instalaci a je nastaven na výchozí kontrolu aktualizací v 6 hodin ráno každého dne.

sudo apt install unattended-upgrades

### Nastavení vzdáleného přístupu

#### Instalace OpenSSH serveru

Pro umožnění připojení se ke vzdálené příkazové řádce bylo potřeba nainstalovat balíček openssh-server pomocí balíčkového manažera apt. Tento balíček nám umožňuje se vzdáleně přihlásit pomocí aplikace ssh.

sudo apt install openssh-server

#### Konfigurace OpenSSH serveru

Před povolením vzdáleného přístupu ve firewallu musíme správně nastavit server pro zvýšení bezpečnosti. Konfigurace pro server na cestě /etc/ssh/sshd\_config byla nahrazena konfigurací v příloze. Ta změnila výchozí port pro komunikaci z portu 22 na port 25522. Dále byl zakázán přístup pomocí hesla, jelikož se pro přihlášení bude využívat zásadně rsa klíčů. Po změně konfigurace byl můj klíč přenesen pro možnost práce na dálku.

#### RSA klíče

Jedná se speciální klíče, které se generují na stanicích, ze kterých chceme vzdálený přístup provozovat. Jejich užití zvyšuje zabezpečení vzdáleného přístupu.

#### Povolení ssh skrz firewall

Jakmile byl ssh server dostatečně zabezpečen, byla povolena komunikace po portu 25522 pomocí ufw. Komunikace stačila povolit pouze pomocí TCP, jelikož ssh nepoužívá UDP.

sudo ufw allow 25522/tcp

## Instalace NGINX

Jakmile bylo zařízeno správné pracovní prostředí, přišla na řadu instalace balíčky NGINX poskytující správné a bezpečné připojení na vnitřní aplikace z vnější. Poskytování připojení pomocí NGINX místo přímého spojení s aplikací je důležité z bezpečnostního hlediska a možnosti přijmutí více požadavků díky asynchronním možnostem tohoto balíčku.

#### Instalace

K instalaci NGINX bylo využito instalačního skriptu z mého GitHub repositáře, který kompiluje celou aplikace od základů pro možnost integrace vlastních modulů, které by jinak přidat do NGINX nebylo možné. Hlavní důvod této volby instalace je modul poskytující připojení pomocí nového HTTP protokolu s číselným označením 3. Tento protokol komunikuje pomocí UDP s využitím protokolu QUIC, který nahrazuje běžný komunikační protokol TCP/IP. Tento skript byl stažen pomocí předinstalované aplikace wget. Poté byla přidělena skriptu práva pro spuštění a skript byl spuštěn se zvýšenými právy pomocí příkazu sudo.

wget <https://raw.githubusercontent.com/bastakka/nginx-autoinstall/master/nginx-autoinstall.sh>

chmod +x nginx-autoinstall.sh

sudo ./nginx-autoinstall.sh

Skript nás nyní vyzval k výběru požadované akce, kde byla vybrána instalace NGINX.

Welcome to the nginx-autoinstall script.

What do you want to do?

1) Install or update Nginx

2) Uninstall Nginx

3) Update the script

4) Install Bad Bot Blocker

5) Exit

Select an option [1-5]: 1

Dále byla vybrána možnost nejnovější verze NGINX v této době 1.21.6

This script will install Nginx with some optional modules.

Do you want to install Nginx stable or mainline?

1) Stable 1.20.1

2) Mainline 1.21.6

Select an option [1-2]: 2

Skript nyní nabídl, které moduly si přejeme nainstalovat. Na dotazy bylo odpovězeno následovně:

Modules to install:

HTTP/3 (⚠️ Patch by Cloudflare, will install BoringSSL, Quiche, Rust and Go) [y/n]: y

Cloudflare's TLS Dynamic Record Resizing patch [y/n]: y

Cloudflare's full HPACK encoding patch [y/n]: y

PageSpeed 1.13.35.2 [y/n]: y

Brotli [y/n]: y

Headers More 0.33 [y/n]: y

GeoIP (BROKEN) [y/n]: n

Fancy index [y/n]: y

ngx\_cache\_purge [y/n]: n

nginx\_substitutions\_filter [y/n]: n

ngx\_http\_lua\_module [y/n]: n

nginx WebDAV [y/n]: n

nginx VTS [y/n]: y

nginx RTMP [y/n]: n

nginx testcookie [y/n]: n

nginx ModSecurity [y/n]: n

Vybrané moduly jsou bezpečnostní a výkonnostní nadstavby pro NGINX. Po dokončení instalace byl proces automaticky spuštěn, nastaven pro automatické spuštění po start počítače a zároveň byl balíčkový manažer apt nastaven tak, aby nezasahoval do knihoven a balíčků této služby. Žádný zásah administrátora nebyl nutný.

#### Konfigurace

NGINX byl nastaven pro automatické použití komprimace pro snížení doby přenosu a pro vyšší výkon při komunikaci s více klienty. Konfigurační soubor byl přiložen v příloze.

Konfigurační soubory pro povolení přístupu k jednotlivým typům aplikacím zvenku byl také přiloženy. Podporovány jsou teoreticky jakékoli aplikace schopné komunikovat pomocí HTTP (soket) protokolu (Python, Node, Rust, C#...) a PHP aplikace. Tyto soubory byly vloženy do adresáře /etc/nginx.

#### Zabezpečení

Pro zabezpečení komunikace pomocí HTTP protokolu byla do adresáře /etc/nginx vložena složka snippets s konfigurací pro bezpečnou komunikaci pomocí SSL, komunikaci skrze proxy k požadované aplikaci a komunikaci pomocí PHP. Této konfigurace se bude využívat u virtuálního hosta pro NGINX.

Dále byl do této složky zkopírován Diffie–Hellmanův klíč pro vytváření klíčů pro komunikaci pomocí SSL. Tento klíč byl zkopírován ze stránek [https://ssl-config.mozilla.org](https://ssl-config.mozilla.org/ffdhe2048.txt) pomocí následující příkazu

sudo curl https://ssl-config.mozilla.org/ffdhe2048.txt > /etc/nginx/dhparam.pem

## Instalace PHP

Pro instalaci správné verze PHP bylo nutné přidat do balíčkového manažera repositář obsahující tuto verzi PHP. Verze 8.0 byla zvolena pro její dlouhou dobu podpory oproti verzi 7.4 a její vyšší kompatibility s aplikacemi oproti verzi 8.1. například aplikace Nextcloud nepodporuje PHP verze 8.1.

|  |  |  |
| --- | --- | --- |
| Verze PHP | Aktivní od | Bezpečnostní aktualizace do |
| 7.4 | 28. 11. 2019 | 28. 11. 2022 |
| 8.0 | 26. 11. 2020 | 26. 11. 2023 |
| 8.1 | 25. 11. 2021 | 25. 11. 2024 |

Tabulka 1: Verze PHP

Do adresáře /etc/apt/sources.list.d byl přidán konfigurační soubor přidávající repositář PHP verze 8, který se poté nainstaloval pomocí následujících příkazu:

curl -sSL https://packages.sury.org/php/README.txt | sudo bash -x

sudo apt update

sudo apt install php8.0-fpm

Při instalaci tohoto balíčku byl automaticky zaveden proces a jeho brána pro komunikaci, pomocí kterého budeme poskytovat PHP webové stránky.

## Instalace databázového serveru

Podobně jako pro PHP bylo nutné přidat repositář i pro databázový server MariaDB, který byl vybrán oproti například MySQL kvůli licenčním důvodům. Repositář byl nainstalován pomocí následující příkazu:

curl -sSL https://r.mariadb.com/downloads/mariadb\_repo\_setup | sudo bash

Příkaz automaticky přidal repositář pro nejnovější podporovanou verzi tohoto databázového serveru pro náš počítač. Databázový server byl poté nainstalován pomocí příkazu:

sudo apt install mariadb-server

Nyní byl server připraven k použití pro aplikace a jeho proces byl také automaticky spuštěn a nastaven pro zapnutí při startu počítače.

## Instalace Node.js

Obdobně jako u předchozích dvou aplikací jsme museli přidat repositář pro požadovanou verzi Node.js. Nejvhodnější kandidát pro naše účely byla verze 16, která v této době poskytovala nejdelší podporu.

|  |  |  |
| --- | --- | --- |
| Verze Node.js | Aktivní od | Bezpečnostní aktualizace do |
| 12 | 23. 4. 2019 | 30. 4. 2022 |
| 14 | 21. 4 2020 | 30. 4. 2023 |
| 16 | 20. 4. 2021 | 30. 4. 2024 |
| 17 | 19. 10. 2021 | 1. 6. 2022 |
| 18 | 19. 4. 2022 | 30. 4. 2025 |

Tabulka 2: Verze Node.js

Více vhodnou verzí by v tomto případě už byla jenom verze 18, která bohužel v době instalace počítače ještě nevyšla.

Repositář byl přidán a balíček byl nainstalován pomocí následujících příkazů:

curl -sSL https://deb.nodesource.com/setup\_16.x | sudo bash

sudo apt install nodejs

Node.js obsahuje vlastní balíčkový manažer npm, který byl poté aktualizován pomocí následujícího příkazu:

sudo npm i -g npm@latest

Tímto byla instalace Node.js u konce.

## Realizace statických stránek

### Výchozí stránka

V nově vytvořené složce /var/www/html byla vytvořena webová stránka vítající návštěvníky s rozcestníkem pro využití poskytovaných služeb. Tuto stránku dostane každý při navštívení výchozí domény. Vlastnictví adresáře /var/www bylo rekurzivně přeneseno na uživatele www-data, za kterého operuje NGINX pomocí příkazu:

sudo chown -R www-data:www-data /var/www

Tato stránka je v příloze. Všechny další stránky v tomto adresáři byly tvořeny za uživatele www-data pomocí příkazu sudo.s příznakem „-u“ a jménem uživatele „www-data“.

### Repositář materiálů pro studenty

Byla vytvořena nová složka /var/www/static.

sudo -u www-data mkdir /var/www/static

K té byl v NGINX nastaven host pro možnost veřejného prohlížení adresáře v prohlížeči. Tento adresář je určen pro statický obsah pro využití jako studijní pomůcka například obrazy linuxových distribucí či dokumenty aj.

Pro hezčí grafické prostředí při prohlížení bylo využito nainstalovaného modulu fancy index a grafického vzhledu dostupného na stránkách [https://github.com/](https://github.com/Naereen/Nginx-Fancyindex-Theme), který byl stažen do tohoto adresáře. Z tohoto vzhledu byl přesunut adresář Nginx-Fancyindex-Theme-dark do kořenové složky a NGINX byl nastaven pro jeho využití. Originální soubory poté byli smazány.

## Instalace webových aplikací

### Cockpit

#### Instalace

Balíček cockpit byl nainstalován pomocí balíčkového manažera apt následujícím příkazem:

sudo apt install cockpit

#### Konfigurace

Balíček byl nakonfigurován pomocí přiloženého souboru pro použití za reverzním proxy serverem. Konfigurace zabezpečuje správné cesty a rozeznání IP adres. Konfigurační soubor byl vložen do adresáře /etc/cockpit.

### Nextcloud

K instalaci aplikace Nextcloud byl striktně dodržen návod na stránkách [https://docs.nextcloud.com/](https://docs.nextcloud.com/server/latest/admin_manual/installation). Poté byla instalace řízena webovým rozhraním této aplikace.

#### Řešení požadavků

Jako první před instalací Nextcloud bylo nutno nainstalovat potřebné moduly pro PHP, které aplikace využívá a potřebuje ke svému správnému běhu. Tyto moduly byli nainstalovány pomocí balíčkového manažera apt následujícím příkazem:

sudo apt install php8.0-gd php8.0-mysql php8.0-curl php8.0-mbstring php8.0-intl php8.0-gmp php8.0-bcmath php8.0-imagick php8.0-xml php8.0-zip php8.0-ctype php8.0-dom php8.0-gd php8.0 php8.0-posix php8.0-apcu

Poté byly staženy zdrojové soubory do složky /var/www pomocí příkazu wget. Nejnovější stabilní verze aplikace Nextcloud byla v dobu instalace verze 23.0.2. Příkaz vypadal následovně:

sudo -u www-data wget <https://download.nextcloud.com/server/releases/nextcloud-23.0.2.zip>

Soubory byly následně extrahovány pomocí příkazu unzip.

sudo -u www-data unzip nextcloud-23.0.2.zip

#### Tvorba databáze

Před instalací bylo ještě nutné nakonfigurovat databázový server. Ten byl nakonfigurován pomocí souboru my.cnf na cestě /etc/mysql. Soubor je v příloze

Nyní byla vytvořena databáze pro nextcloud pomocí terminálové aplikace našeho databázového serveru. Do této aplikace jsme se dostali pomocí příkazu:

sudo mysql

V této aplikaci jsme nejdříve vytvořili uživatele pro vnitřní přístup k této databázi se jménem nextcloud a heslem „gfMa2qZ44ewHyhpP“ pomocí příkazu:

CREATE USER 'nextcloud'@'localhost' IDENTIFIED BY 'gfMa2qZ44ewHyhpP';

Heslo v tomto případě bude možno ponechat, jelikož přístup k databázi bude pouze vnitřní a bude použito pouze jednou při instalaci Nextcloud. Poté byla vytvořena samotná databáze pomocí příkazu:

CREATE DATABASE IF NOT EXISTS nextcloud CHARACTER SET utf8mb4 COLLATE utf8mb4\_general\_ci;

A následně byly uděleny práva pro tuto databázi vytvořenému účtu pomocí:

GRANT ALL PRIVILEGES on nextcloud.\* to 'nextcloud'@'localhost';

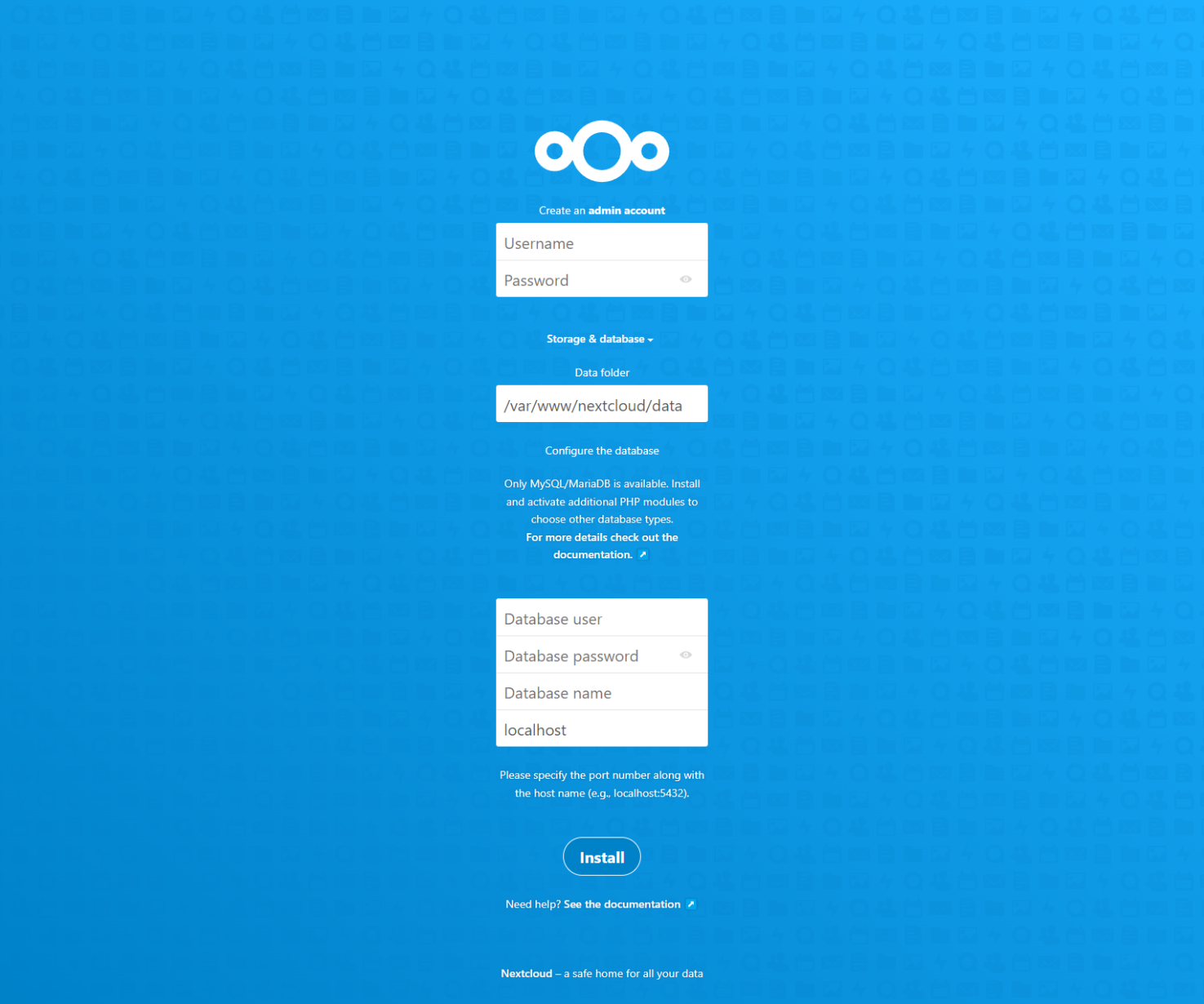
Nyní bylo možné uložit změny a odejít.

FLUSH privileges;

quit;

#### Zavedení nextcloud do NGINX

V tomto kroku bylo nutné zavést vstup do složky nextcloud do NGINX pomocí PHP serveru. Vstup byl zaveden pomocí dočasného virtuálního hosta, který je obsažen v konečném hostovi. Po tomto kroku bylo možné se dostat k webovým stránkám, kde mohla instalace pokračovat. Stránka vypadala takto:



Obrázek 1: Úvodní stránka Nextcloud při instalaci

#### Instalace

Nyní jsme ve webovém rozhraní vytvořili administrátorský účet v Nextcloud, ze kterého budeme tuto aplikaci spravovat, se jménem „administrator“ a heslem „jNqJWBz9Ltvvne6X“. Toto heslo byla muset být po předání změněno. Adresář pro data byl ponechán prozatím ve výchozí hodnotě.

Do nastavení databáze byly zapsány hodnoty:

nextcloud

gfMa2qZ44ewHyhpP

nextcloud

localhost:3306

Toto jsou naše dříve nastavené hodnoty z tvorby databáze a výchozí adresa databázového serveru pomocí vnitřní komunikace na portu 3306. Bylo stisknuto tlačítko Install.

Aplikace nyní sama naplnila databázi potřebnými tabulkami a nabídla nám nainstalovat doporučené aplikace. Tato nabídka byla odmítnuta. Nextcloud byl tímto nainstalován a bylo možné využívat jeho webového prostředí.

#### Nastavení procesů na pozadí

Jelikož Nextcloud je PHP aplikace, tak nemá žádný proces, který by zařizoval na pozadí časově nastavené úkoly. Z tohoto důvodu bylo nutné vytvořit nový pravidelný proces pro tyto úkoly pomocí démona systemd. Tento úkol a jeho časovač byly vloženy do složky /etc/systemd/system a následně povoleny pomocí následujícího příkazu:

sudo systemctl enable nextcloudcron.timer

Tento proces byl nastaven pro pravidelné spuštění úkolů na pozadí každých pět minut. Nextcloud nabízí také možnost spuštění těchto úkolů na pozadí pokaždé kdy jakýkoli uživatel navštíví webové rozhraní. To je ovšem v našem případě velmi nestabilní, jelikož nemáme v tento moment pravidelné návštěvníky.

#### Nastavení konfigurace pro Nextcloud

Konfiguračního souboru pro nextcloud na cestě /var/www/nextcloud/config/config.php byl upraven pro využití paměti cache a pro výchozí zobrazení v českém jazyce.

Upravený soubor je v příloze.

#### Nastavení proměnné PATH pro PHP

Pro správnou funkci aplikace byly z konfiguračního souboru pro webové PHP na cestě /etc/php/8.0/fpm/pool.d/www.conf odstraněny komentářové znaky „;“ z následujících řádků:

;env[HOSTNAME] = $HOSTNAME

;env[PATH] = /usr/local/bin:/usr/bin:/bin

;env[TMP] = /tmp

;env[TMPDIR] = /tmp

;env[TEMP] = /tmp

Toto umožnilo PHP využívat nainstalovaných aplikací na počítači.

#### Nastavení maximálního využití operační paměti pro PHP

Pro plynulý chod aplikace Nextcloud i při velké zátěži byl změněn limit využití operační pro PHP. Tato změna byla v souboru /etc/php/8.0/fpm/php.ini na řádku 430, kde byla hodnota navýšena na 1G.

#### Dokončení instalace

Po dokončení všech předchozích kroků bylo možné využití aplikace Nextcloud i při velké zátěži v případě umožnění přístupu žákům.

### PCMS

PCMS se skládá ze dvou aplikací, které mezi sebou komunikují pomocí databáze. Mezi tyto aplikace patří backend a frontend.

### Tvorba databáze

Pomocí terminálové aplikace našeho databázového serveru byl vytvořen nový uživatel se jménem „pcms“ a heslem „UUgrZEBRxnCKUV2p“ a nová databáze pro PCMS. Heslo je v tomto případě opět pouze pro vnitřní účely a nebude muset být upraveno.

sudo mysql

CREATE USER 'pcms'@'localhost' IDENTIFIED BY 'UUgrZEBRxnCKUV2p';

CREATE DATABASE pcms;

FLUSH privileges;

quit;

#### Instalace

Před instalací bylo nezbytné nainstalovat pomocí balíčkového manažera apt balíčky, které PCMS vyžaduje.

sudo apt install ffmpeg imagemagick

Z repositáře na serveru [https://bitbucket.org/](https://bitbucket.org/purkynka-renewal/sspbrnoweb-monorepo/src) byl do adresáře /srv stažen zdrojový kód pomocí příkazu git:

git clone <https://bastakka@bitbucket.org/purkynka-renewal/sspbrnoweb-monorepo.git>

Poté byl pomocí balíčkového manažera npm stažen potřebný balíček pnpm.

sudo npm i -g pnpm

Nyní byl vytvořen konfigurační soubor .env, který se nachází ve výchozím adresáři aplikace. Tento konfigurační soubor je v příloze. V tomto konfiguračním souboru se určuje komunikace s databází a v našem případě i způsob komunikace se serverem NGINX.

Ve výchozím adresáři repositáře byla provedena instalace backend aplikace a jejich vyžadovaných balíčků pomocí následujících příkazů.

pnpm i -P

pnpm build:backend

Po kterém byl spuštěn nově vytvořený proces „pcms“ v démonu systemd.

A nyní bylo možné nainstalovat frontend pomocí následujícího příkazu.

pnpm buid:frontend

Tímto instalace skončila

#### Zavedení do démona systemd

Poté byli do démona systemd byly zavedeny procesy pro všechny aplikace pomocí servisních souborů v příloze do složky /etc/systemd/system.

Instalace PCMS tímto krokem pro nás skončila. Obsluha této aplikace je popsána v místní maturitní dokumentaci od pana Adama Žingora.

## Zavedení webových stránek

### Zabezpečení přístupu k určitým webovým stránkám

Pro zabezpečení přístupu k webovým stránkách určeným pouze pro administrátory byl nainstalován balíček generující přihlašovací údaje pro webové stránky pomocí balíčkového manažera apt.

sudo apt install apache2-utils

Následně byl pro webový adresář /status poskytující údaje o webovém provozu ke každé aplikaci vygenerován v adresáři /var/auth/status soubor users s přihlašovacími údaji pro přístup. Bylo zvoleno jméno „administrator“ a heslo „ch2Z7aUJSrMAynCK“. Soubor je možné upravovat pomocí textového editoru. Soubor byl vytvořen pomocí příkazu:

sudo -u www-data htpasswd -c users administrator

### Zavedení virtuálního hosta do NGINX

Webové stránky a aplikace byly do NGINX zavedeny pomocí jednoho virtuálního hosta v příloze, který zabezpečuje připojení ke každé z aplikací podle adresáře při žádosti na výchozí doménu. Tímto způsobem bylo možné od sebe oddělit aplikace i přes vlastnictví pouze jedné domény. Každé z aplikací byli přiděleny logy v adresáři /var/log/nginx ve fomátu *název\_aplikace-access* a *název\_aplikace-error* ve kterých je možné sledovat žádosti a chyby při zpracování požadavku. Zároveň byla nastavena paměť cache pro využití při obsluze.

### Povolení HTTP a HTTPS ve firewall

Poslední krok potřebný k úspěšnému přístupu ke stránkám je povolení komunikačních portů 80 a 443 ve firewallu. Povoleny byly pomocí aplikace ufw následujícím způsobem:

sudo ufw allow 80/tcp

sudo ufw allow 443

Na portu 80 byla povolena komunikace pouze pro TCP, jelikož se pomocí UDP na tomto portu komunikovat nebude.

# Závěr

Výsledkem této maturitní práce je virtuální počítač, který splňuje požadavky pro možnost hostingu webových stránek a aplikací. Zároveň s počítačem byl předán vyhotovený uživatelský manuál určen pro administrátory, kde jsou popsány všechny důležité příkazy pro správu.

Počítač obsahuje potřebné prostředky pro bezpečnou komunikaci po internetu.

Tato práci mi dovolila zkombinovat znalosti nabité z osobní zkušenosti a praktické zkušenosti z reálného prostředí v dnešních korporacích, což mi pomohlo k zjištění hranic mých možností v této oblasti.

# Seznam ilustrací

[Obrázek 1: Úvodní stránka Nextcloud při instalaci 30](#_Toc96957193)

# Seznam tabulek

[Tabulka 1: Verze PHP 24](#_Toc96957191)

[Tabulka 2: Verze Node.js 26](#_Toc96957192)

# Zdroje

1. *Debian* [online] 2022 [cit. 2022-03-09] Dostupné z: <https://www.debian.cz>
2. *Apt* [online] 2005 [cit. 2022-03-09] Dostupné z: [https://www.debian.org](https://www.debian.org/doc/manuals/apt-howto/ch1.cs.html)
3. *Systemd* [online] 2022 [cit. 2022-03-09] Dostupné z: [https://www.freedesktop.org](https://www.freedesktop.org/wiki/Software/systemd/)
4. *UFW* [online] 2022 [cit. 2022-03-09] Dostupné z: [https://wiki.ubuntu.com](https://wiki.ubuntu.com/UncomplicatedFirewall)
5. *Fail2ban* [online] 2022 [cit. 2022-03-09] Dostupné z: [https://github.com](https://github.com/fail2ban/fail2ban)
6. *SSH* [online] 2019 [cit. 2022-03-09] Dostupné z: [https://wiki.ubuntu.cz](https://wiki.ubuntu.cz/ssh)
7. *Vim* [online] 2019 [cit. 2022-03-09] Dostupné z: [https://wiki.ubuntu.cz](https://wiki.ubuntu.cz/vim)
8. *Git* [online] 2022 [cit. 2022-03-09] Dostupné z: [https://git-scm.com](https://git-scm.com/book/cs/v2)
9. *NGINX* [online] 2022 [cit. 2022-03-09] Dostupné z: <https://nginx.org>
10. *MariaDB* [online] 2022 [cit. 2022-03-09] Dostupné z: <https://mariadb.org/>
11. *PHP* [online] 2001-2022 [cit. 2022-03-09] Dostupné z: [https://wiki.php.net](https://wiki.php.net/)
12. *Node.js* [online] 2022 [cit. 2022-03-09] Dostupné z: <https://nodejs.org/>
13. *Cockpit* [online] 2022 [cit. 2022-03-09] Dostupné z: <https://cockpit-project.org/>
14. *Nextcloud* [online] 2022 [cit. 2022-03-09] Dostupné z:<https://docs.nextcloud.com>
15. *PCMS* [online] 2022 [cit. 2022-03-09] Dostupné z: [https://bitbucket.org/](https://bitbucket.org/purkynka-renewal/sspbrnoweb-monorepo/src/master/)

# Přílohy

1. Složka s virtuálním počítačem
2. Uživatelský manuál
3. Konfigurační soubor pro OpenSSH server
4. Konfigurační soubory pro systemd
5. Konfigurační soubory pro NGINX
6. Konfigurační soubor pro Cockpit
7. Výchozí webová stránka
8. Konfigurační soubor pro MariaDB
9. Konfigurační soubory pro PCMS