

Platforma pro offline aktualizaci WordPressu

Bc. Zdeněk Józsa

Diplomová práce
2019



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2018/2019

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Zdeněk Józsa**
Osobní číslo: **A16562**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Bezpečnostní technologie, systémy a management**
Forma studia: **kombinovaná**

Téma práce: **Platforma pro offline aktualizaci WordPressu**

Téma anglicky: **A Platform for the Offline Updating of WordPress**

Zásady pro vypracování:

1. Specifikujte možnosti offline aktualizací platformy Wordpress.
2. Navrhněte infrastrukturu pro aktualizace.
3. Navrhněte způsob zabezpečení komunikace mezi agenty a centrálním serverem.
4. Proveďte testovací implementaci navrženého řešení.
5. Otestujte funkčnost řešení na poskytnutých scénářích aktualizací.



Rozsah diplomové práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. RAPHAEL HERTZOG AND ROLAND MAS. The Debian administrator's handbook: Debian squeeze from discovery to mastery. Lulu Com, 2012. ISBN 9791091414005.
2. WILLIAMS, Brad, David. DAMSTRA a Hal STERN. Professional WordPress: design and development. Second edition. Indianapolis, IN: John Wiley & Sons, 2013. Wrox professional guides. ISBN 978-1-118-44227-2.
3. SOBELL, Mark G. Mistrovství v Linuxu: příkazový řádek, shell, programování. Brno: Computer Press, 2007. Mistrovství. ISBN 978-80-251-1726-2.
4. KRČMÁŘ, Petr. Linux: tipy a triky pro bezpečnost. Praha: Grada, 2004. ISBN 8024708124.
5. BRANDEJS, Michal. Linux: praktický průvodce. 2. vyd., (V Konvoji 1.). Brno: Konvoj, 2003. ISBN 80-7302-050-5.
6. KYSELA, Martin. 333 tipů a triků pro Linux. Brno: CP Books, 2005. ISBN 80-722-6866-x.

Vedoucí diplomové práce:

Ing. David Malaník, Ph.D.

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

30. listopadu 2018

Termín odevzdání diplomové práce:

17. května 2019

Ve Zlíně dne 14. prosince 2018

doc. Mgr. Milan Adámek, Ph.D.
děkan



doc. RNDr. Vojtěch Křesálek, CSc.
ředitel ústavu

Jméno, příjmení: Bc. Zdeněk Józsa

Název diplomové práce: Platforma pro offline aktualizaci WordPressu

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen s příjmem-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 23.5.2019

Bc. Zdeněk Józsa, v.r.
podpis diplomanta

ABSTRAKT

Diplomová práce se zabývá návrhem a implementací platformy pro offline aktualizaci redakčního systému WordPress. Tento systém je ve velké míře celosvětově využíván k tvorbě webových stránek, přičemž jsou kladeny vysoké požadavky na jeho zabezpečení. Jedním ze způsobů, jak dosáhnout vysoké úrovně zabezpečení, je provádění pravidelných aktualizací. Systém WordPress sám o sobě nabízí několik možností, jak aktualizaci provést. Při správě vyššího počtu instalací systému WordPress však vznikla potřeba centrálního řízení těchto aktualizací a zvýšení úrovně dohledu při jejich provádění. Za tímto účelem bylo navrženo a implementováno softwarové řešení, které umožňuje prostřednictvím administračního rozhraní distribuci aktualizací pro jednotlivé instalace systému WordPress z centrálního serveru. Součástí implementovaného řešení je zabezpečení datového přenosu při provádění jednotlivých aktualizací.

Klíčová slova: WordPress, offline aktualizace, zabezpečení

ABSTRACT

The thesis deals with the design and implementation of the platform for offline updating of the WordPress content management system. This system is used extensively worldwide to create websites, with high security requirements. One way to achieve a high level of security is to perform regular updates. WordPress itself offers several ways to perform the update. However, when managing a higher number of WordPress installations, there was a need for centralized management of these updates and an increased level of oversight in their implementation. For this purpose, a software solution has been designed and implemented that allows the distribution of updates for individual installations of WordPress from a central server through the administration interface. Part of the proposed solution is the security of data transmission when performing individual updates.

Keywords: WordPress, offline updates, security

PODĚKOVÁNÍ

Na tomto místě bych rád poděkoval především vedoucímu mé diplomové práce panu *Ing. Davidu Malaníkovi, Ph.D.* za poskytnutí konzultací, vynaložený čas, ochotu, věcné připomínky a návrhy, kterými byl nápomocen při zhotovení této práce.

Dále bych rád poděkoval za podporu své rodině, která mi pomohla vytvořit prostředí, v kterém jsem mohl nerušeně pracovat.

MOTTO

„Na světě není nic mocnějšího než myšlenka, která přišla včas.“

Victor Hugo

OBSAH

ÚVOD.....	10
I TEORETICKÁ ČÁST.....	11
1 SPECIFIKACE MOŽNOSTÍ OFFLINE AKTUALIZACÍ PLATFORMY WORDPRESS.....	12
1.1 AUTOMATICKÉ AKTUALIZACE.....	12
1.1.1 Konfigurace automatických aktualizací.....	13
1.1.1.1 Aktualizace pomocí wp-config.php.....	13
1.1.1.2 Aktualizace pomocí filtrů.....	16
1.2 AKTUALIZACE PROSTŘEDNICTVÍM ADMINISTRAČNÍHO ROZHRAŇÍ.....	16
1.3 POUŽITÍ PLUGINU.....	17
1.3.1 Easy Updates Manager.....	18
1.3.2 Disable All WordPress Update.....	20
1.4 MANUÁLNÍ AKTUALIZACE.....	20
1.4.1 Stažení aktualizací balíčku.....	21
1.4.2 Doporučený postup aktualizace.....	21
2 NÁVRH INFRASTRUKTURY PRO AKTUALIZACE.....	22
2.1 CENTRÁLNÍ SERVER PRO ŘÍZENÍ DISTRIBUCE AKTUALIZACÍ.....	23
2.1.1 Administrační rozhraní.....	24
2.1.1.1 Přehled.....	24
2.1.1.2 Agenti.....	24
2.1.1.3 Uživatelé.....	26
2.1.1.4 Aktualizace.....	26
2.1.1.5 Logy.....	26
2.1.2 Zabezpečení administračního rozhraní.....	27
2.1.2.1 Autentizace uživatele.....	27
2.1.2.2 Princip zabezpečení uživatelského hesla.....	28
2.1.2.3 Autorizace uživatele.....	29
2.1.2.4 Role a oprávnění.....	30
2.1.2.5 Nette framework.....	31
2.1.3 Databázové schéma administračního rozhraní.....	31
2.2 AGENTI AKTUALIZACÍ.....	32
2.2.1 Server.....	33
2.2.2 Princip činnosti systému WordPress.....	33
2.2.2.1 Databázové schéma systému WordPress.....	34
3 NÁVRH ZPŮSOBU ZABEZPEČENÍ KOMUNIKACE MEZI AGENTY A CENTRÁLNÍM SERVEREM.....	36
3.1 SSH PROTOKOL.....	36
3.2 RSA ŠIFROVÁNÍ.....	37
3.3 ZPŮSOB SPOJENÍ MEZI CENTRÁLNÍM SERVEREM A AGENTY.....	38
II PRAKTICKÁ ČÁST.....	39
4 TESTOVACÍ IMPLEMENTACE NAVRŽENÉHO ŘEŠENÍ.....	40

4.1	CENTRÁLNÍ SERVER	40
4.1.1	Operační systém	40
4.1.2	Další komponenty	41
4.1.2.1	Apache HTTP server	41
4.1.2.2	PHP	41
4.1.2.3	MySQL databáze	41
4.2	AGENTI	43
4.2.1	Operační systém	43
4.2.2	Další komponenty	44
4.2.2.1	Apache HTTP server	44
4.2.3	PHP	44
4.2.4	MySQL.....	44
4.2.5	Instalace zdrojových souborů systému WordPress	44
4.2.6	Instalace systému WordPress	45
4.2.7	Ověření funkčnosti instalace	46
5	TESTOVÁNÍ FUNKČNOSTI ŘEŠENÍ NA POSKYTNUTÝCH SCÉNÁŘÍCH AKTUALIZACÍ	47
5.1	TESTOVÁNÍ FUNKČNOSTI ADMINISTRAČNÍHO ROZHRANÍ.....	47
5.1.1	Přihlašovací stránka	47
5.1.2	Přihlášení uživatele	47
5.1.3	Omezení práv jednotlivých uživatelských rolí.....	48
5.1.4	Funkce v jednotlivých sekcích administračního systému	48
5.2	TESTOVÁNÍ AKTUALIZACE AGENTŮ.....	48
5.2.1	Navázání spojení s agentem přes SSH	48
	ZÁVĚR	49
	SEZNAM POUŽITÉ LITERATURY.....	50
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	51
	SEZNAM OBRÁZKŮ	52
	SEZNAM PŘÍLOH.....	53

ÚVOD

Redakční systém WordPress je obecně považován za jeden z celosvětově nejpoužívanějších redakčních systémů pro tvorbu webových stránek. Je kolem něj soustředěna široká uživatelská i vývojářská komunita, přičemž jsou kladeny vysoké požadavky na kvalitu a zabezpečení tohoto systému. Z těchto důvodů dochází k pravidelnému vydávání nových verzí, prostřednictvím nichž je možno provádět aktualizace tohoto systému. Vzhledem k tomu, že každá nová verze přináší jak opravy chyb, tak vylepšení stávajících a implementaci nových funkcí, je žádoucí s ohledem na udržení dostatečné úrovně zabezpečení provádět aktualizace systému co nejdříve po vydání nové verze, aby nemohlo dojít například ke zneužití některých chyb vyskytujících se ve verzi původní.

Systém WordPress v současnosti nabízí na výběr z několika možností aktualizací. Jsou to automatické aktualizace, které se provádí bez nutnosti zásahu správce systému, dále aktualizace prováděné prostřednictvím administračního rozhraní, které se tedy realizují v závislosti na provedené akci správce systému, dále může být za účelem provedení aktualizace využito speciálního pluginu, pomocí něž lze nastavit různá kritéria, dle kterých je aktualizace provedena, a nakonec lze realizovat také manuální aktualizaci, při které je zapotřebí stáhnout aktualizací soubory a ručně je zkopírovat do adresářové struktury systému WordPress.

Výše uvedené možnosti provádění aktualizací jsou blíže popsány v první kapitole, která je spolu s dalšími dvěma kapitolami obsahem teoretické části diplomové práce. Druhá kapitola popisuje návrh infrastruktury pro provádění tzv. offline aktualizací, což je specifický způsob aktualizací, které jsou řízeny z centrálního serveru a je možno je provádět pro různý počet instalací systému WordPress. Ve třetí kapitole je popsán způsob zabezpečení komunikace mezi centrálním serverem a jednotlivými agenty, na kterých je systém WordPress nainstalován. Je zde vysvětlen jak způsob zabezpečení administračního rozhraní, tak princip zabezpečení přenosu dat, který má zamezit jejich možnému odposlechu nebo pozměnění. Praktická část práce začíná čtvrtou kapitolou, jež je věnována popisu testovací implementace navrženého řešení. Pátá a současně poslední kapitola této práce obsahuje popis testování funkčnosti řešení na poskytnutých scénářích aktualizací.

I. TEORETICKÁ ČÁST

1 SPECIFIKACE MOŽNOSTÍ OFFLINE AKTUALIZACÍ PLATFORMY WORDPRESS

K redakčnímu systému WordPress jsou vydávány pravidelné aktualizace, jejichž instalace je nezbytná za účelem možnosti využití novějších a vylepšených funkcí a v neposlední řadě také k zajištění zabezpečení systému proti útokům. Pochopení toho, co tyto nové funkce přinášejí, může pomoci s plánováním nových WordPress projektů. Nejjednodušší způsob, jak toho dosáhnout, je navštívení oficiálních WordPress vývojových stránek na adrese <http://make.wordpress.org/core/>¹.

Aktualizaci lze provést několika způsoby, mezi něž patří automatické aktualizace, aktualizace prostřednictvím administračního rozhraní, aktualizace za pomoci speciálního pluginu nebo manuální aktualizace, při níž jsou původní soubory systému manuálním kopírováním v rámci souborového systému WordPress nahrazeny soubory novými. Jednotlivé způsoby aktualizací jsou popsány v následujících podkapitolách.

1.1 Automatické aktualizace

Redakční systém WordPress poskytuje možnost automatických aktualizací bez nutnosti zásahu ze strany uživatele. Vzhledem k tomu, že se automatické aktualizace aktivují prostřednictvím PHP funkcí systému WordPress, je však v tomto případě nutnost zásahu programátora. Za účelem automatických aktualizací existují tzv. Core updates, které se dále dělí do tří typů:

1. Core development updates, známé jako „bleeding edge“
2. Minor core updates, verze pro údržbu a zabezpečení
3. Major core release updates

¹ WILLIAMS B., D. DAMSTRA a H. STERN. *Professional WordPress: design and development*. Indianapolis: John Wiley & Sons, 2013. ISBN 978-1-118-44227-2.

Ve výchozím nastavení má každá instalace systému WordPress povoleny automatické aktualizace pro Minor core updates. Toto nastavení bylo zavedeno pro verze WordPress 3.7+².

Výhodou automatických aktualizací je úplná kontrola nad jejich řízením, tedy povolením a zakázáním jejich jednotlivých typů.

Nevýhodou těchto aktualizací je nemožnost kontroly nad jejich řízením z administrace systému, jelikož je nutno je povolit a zakázat přímo v programovém kódu souboru `wp-config.php`.

1.1.1 Konfigurace automatických aktualizací

Automatické aktualizace lze konfigurovat pomocí jednoho ze dvou způsobů:

1. definování konstant v souboru `wp-config.php`
2. přidání filtrů za použití pluginu

1.1.1.1 Aktualizace pomocí `wp-config.php`

Automatické aktualizace systému WordPress lze konfigurovat pomocí souboru `wp-config.php`. Použitím definic odpovídajících konstant lze automatické aktualizace úplně zakázat a aktualizace Core updates lze zakázat nebo nakonfigurovat podle typu aktualizace.

Konstanta pro vypnutí všech aktualizací

Za účelem zakázání všech typů automatických aktualizací je zapotřebí přidat do souboru `wp-config.php` následující řádek kódu s definicí konstanty:

² Updating WordPress. In: WordPress [online]. [cit. 2019-05-17]. Dostupné z: https://codex.wordpress.org/Updating_Wordpress

```
define('AUTOMATIC_UPDATER_DISABLED', true);
```

Konstanta pro konfiguraci aktualizací jádra (Core Updates)

Za účelem povolení automatických aktualizací pro hlavní vydání nebo vývojové účely se používá konstanta `WP_AUTO_UPDATE_CORE`. Definici této konstanty lze provést třemi způsoby, resp. konstanta může nabývat tří různých hodnot, z nichž každá zajistí jiné chování:

- `true` – vývojářské, Minor updates a Major updates jsou povoleny
- `false` – vývojářské, Minor updates a Major updates jsou zakázány
- `'minor'` – Minor updates jsou povoleny, vývojářské a Major updates jsou zakázány

Defaultní hodnota konstanty `WP_AUTO_UPDATE_CORE` na stránkách pro vývojové účely nastavena na `true`, ta ostatních stránkách je tato hodnota nastavena na `'minor'`.

```
0 10 20 30 40 50 60 70
20
21 // ** MySQL settings - You can get this info from your web host ** //
22 /** The name of the database for WordPress */
23 define('DB_NAME', 'database_name_here');
24
25 /** MySQL database username */
26 define('DB_USER', 'username_here');
27
28 /** MySQL database password */
29 define('DB_PASSWORD', 'password_here');
30
31 /** MySQL hostname */
32 define('DB_HOST', 'localhost');
33
34 /** Database Charset to use in creating database tables. */
35 define('DB_CHARSET', 'utf8');
36
37 /** The Database Collate type. Don't change this if in doubt. */
38 define('DB_COLLATE', '');
39
40 /** Disable all updates */
41 define('AUTOMATIC_UPDATER_DISABLED', true);
42
43 /**#@+
44 * Authentication Unique Keys and Salts.
45 *
46 * Change these to different unique phrases!
47 * You can generate these using the {@link https://api.wordpress.org/secret-ke
48 * You can change these at any point in time to invalidate all existing cookie
49 *
50 * @since 2.6.0
51 */
52 define('AUTH_KEY', 'put your unique phrase here');
53 define('SECURE_AUTH_KEY', 'put your unique phrase here');
54 define('LOGGED_IN_KEY', 'put your unique phrase here');
55 define('NONCE_KEY', 'put your unique phrase here');
56 define('AUTH_SALT', 'put your unique phrase here');
57 define('SECURE_AUTH_SALT', 'put your unique phrase here');
58 define('LOGGED_IN_SALT', 'put your unique phrase here');
59 define('NONCE_SALT', 'put your unique phrase here');
```

Obr. 1 Zdrojový kód souboru wp-config.php³

³ Zdroj: vlastní

1.1.1.2 Aktualizace pomocí filtrů

Použití filtrů umožňuje jemné ladění automatických aktualizací.

Všechny aktualizace lze zakázat pomocí následujícího filtru:

```
add_filter('automatic_updater_disabled', '__return_true');
```

Pouze core-type automatické aktualizace lze povolit pomocí následujícího filtru:

```
add_filter('auto_update_core', '__return_true');
```

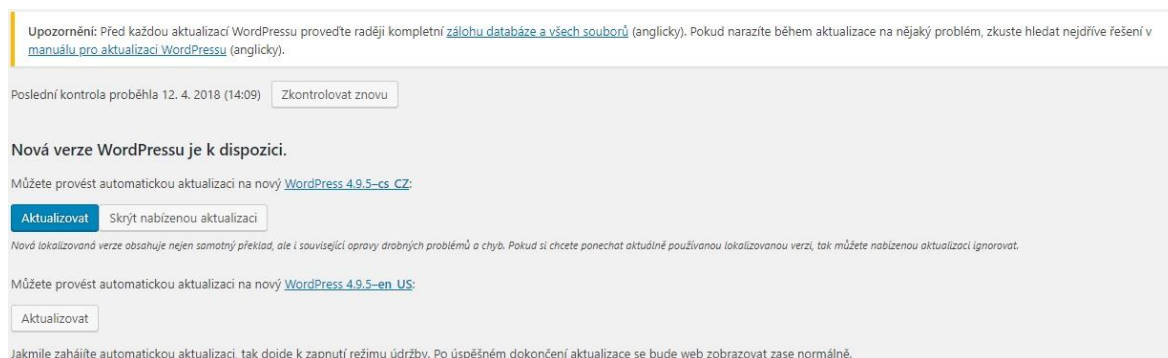
Pro samostatné povolení nebo zakázání jednoho ze tří typů aktualizací jádra (Core updates) lze použít následujících filtrů:

- `allow_dev_auto_core_updates` – povolí automatické aktualizace vývojářské
- `allow_minor_auto_core_updates` – povolí Minor updates
- `allow_major_auto_core_updates` – povolí Major updates

1.2 Aktualizace prostřednictvím administračního rozhraní

Aktualizaci systému WordPress lze spustit také z administračního rozhraní, k čemuž již není potřeba zvláštních programátorských znalostí. Po přihlášení do administrace je v případě vydání nové aktualizace v sekci Nástěnka zobrazeno upozornění ve znění „Nová verze WordPressu je k dispozici.“. Ke spuštění aktualizace již postačí kliknout na tlačítko „Aktualizovat“. Během provozu je vhodné provádět aktualizace pokaždé, když k tomu systém vyzve, a platí to pochopitelně i pro jednotlivé pluginy⁴.

⁴ KUDLÁČEK, Luboš. *WordPress: podrobný průvodce tvorbou a správou webů*. Brno: Computer Press, 2013. ISBN 978-80-251-2734-6



Obr. 2 Ukázka možnosti aktualizace prostřednictvím administračního rozhraní⁵

Výhodou aktualizace z administračního rozhraní je jednoduchá a intuitivní obsluha spuštění samotné aktualizace.

Nevýhodou je nutnost častého přihlašování do administrace za účelem kontroly, zda byla vydána nová aktualizace systému v případě, že administrátor stránek chce mít systém stále aktuální.

1.3 Použití pluginu

Aktualizace systému WordPress lze spravovat také pomocí speciálních pluginů, tedy zásuvných modulů, které je možno doinstalovat k základní instalaci WordPress. Správce systému má následně ve WordPress adresáři pluginů na výběr z několika pluginů, které poskytují možnost správy aktualizací. Níže budou stručně představeny dva z nich, které jsou primárně určeny k rozdílným způsobům správy automatických aktualizací.

Výhodou použití pluginů ke správě aktualizací je jednoduché a intuitivní ovládání, přičemž není třeba speciálních znalostí správce webu.

Za nevýhodu použití pluginů je možno považovat riziko, že daný plugin z důvodu zastavení jeho vývoje přestane být aktuální, případně jeho použití nemusí být kompatibilní s aktuální verzí systému WordPress.

⁵ Zdroj: vlastní

1.3.1 Easy Updates Manager

Plugin Easy Updates Manager umožňuje správu jednotlivých typů aktualizací systému WordPress. Základní funkce tohoto pluginu umožňuje zakázání nebo povolení všech aktualizací, které WordPress nabízí, tedy aktualizace jádra (Core updates), aktualizace jazykových mutací (Translation updates), aktualizace pluginů (Plugin updates a aktualizaci šablony (Theme updates).

Disable All Updates

This is a master switch and will enable or disable updates for the WordPress installation.

All updates are allowed, however, you still need to configure the updates below.

Enable All Updates Disable All Updates

Automatic Updates

You have selected to customize the updates below.

Default On Off Custom

Major Releases

Automatically update to major releases (e.g., 4.1, 4.2, 4.3).

Automatic major release updates are now turned off.

Enable Major Releases Disable Major Releases

Minor Releases

Automatically update to minor releases (e.g., 4.1.1, 4.1.2, 4.1.3).

Automatic minor release updates are now turned on.

Enable Minor Releases Disable Minor Releases

Development Updates

Allow your install to receive development updates from WordPress (for advanced users only)

Automatic development release updates are now turned off.

Enable Development Updates Disable Development Updates

Obr. 3 Ovládací prvky pluginu Easy Updates Manager⁶

⁶ Zdroj: vlastní

1.3.2 Disable All WordPress Update

Plugin Disable All WordPress Update nabízí možnost jak jednoduchým způsobem povolit nebo zakázat veškeré aktualizace, tedy aktualizace jádra (Core update), aktualizace šablony (Theme update) a aktualizace pluginů (Plugin update).

Všechny tyto aktualizace lze zakázat pouhým aktivováním pluginu po jeho instalaci. Opětovné povolení všech uvedených aktualizací lze provést deaktivací nainstalovaného pluginu.



Obr. 4 Ukázka možnosti aktivace / deaktivace pluginu⁷

1.4 Manuální aktualizace

Jedním ze způsobů, jak provést aktualizaci systému WordPress, je stažení aktualizčního balíčku, který obsahuje aktualizované adresáře a soubory a následné nahrazení původních adresářů a souborů, kterých se daná aktualizace týká.

⁷ Zdroj: vlastní

1.4.1 Stažení aktualizačního balíčku

Aktualizační balíček systému WordPress lze stáhnout prostřednictvím URL <https://wordpress.org/download/>, kde jsou k dispozici odkazy na archivované soubory. Ke stažení je možno zvolit variantu ZIP nebo GZIP.

1.4.2 Doporučený postup aktualizace

1. Zálohování databáze.
2. Zálohování vše souborů systému WordPress včetně souboru .htaccess.
3. Ověření existence vytvořených záloh a jejich použitelnosti.
4. Deaktivace všech pluginů
5. Ujistění, zda jsou provedeny předešlé čtyři kroky, jinak nelze systém aktualizovat
6. Stažení a extrakce balíčku WordPress z <http://wordpress.org/download/>
7. Odstranění starých souborů WordPress vyjma:
 - a. soubor wp-config.php
 - b. složka wp-content; výjimka: složky wp-content/cache a wp-content/plugins/widgets by měly být smazány
 - c. složka wp-images
 - d. složka wp-includes/languages/ - pokud není používán jazykový soubor
 - e. soubor .htaccess – pokud nejsou přidána vlastní pravidla
 - f. soubor robots.txt
8. Nahrání nových souborů z pevného disku počítače do příslušné složky aplikace WordPress
9. Spuštění aktualizačního programu WordPress a postupování podle pokynů na obrazovce.
10. Aktualizace Permalinks a .htaccess (trvalé odkazy)
11. Instalace aktualizovaných pluginů a šablon (zkontrolovat, zda jsou kompatibilní s novou verzí systému WordPress)
12. Aktivace pluginů
13. Přidání definice bezpečnostních klíčů do souboru wp-config.php

2 NÁVRH INFRASTRUKTURY PRO AKTUALIZACE

Základní vizí navrhovaného portálu je umožnit jednoduchou a pohodlnou správu aktualizací jednotlivých WordPress instalací z jednoho místa, kde by současně byl k dispozici přehled o jejich aktuálním stavu. K tomuto účelu bude sloužit administrační systém, který poběží na centrálním serveru. Z uvedeného současně vyplývá, že aktualizace budou probíhat na dálku. Musí tedy být zajištěna možnost spojení mezi centrálním serverem a jednotlivými agenty, na což je třeba brát při návrhu infrastruktury zřetel. V úvahu je nutno zahrnout také zabezpečení administračního rozhraní a samotného přenosu dat.

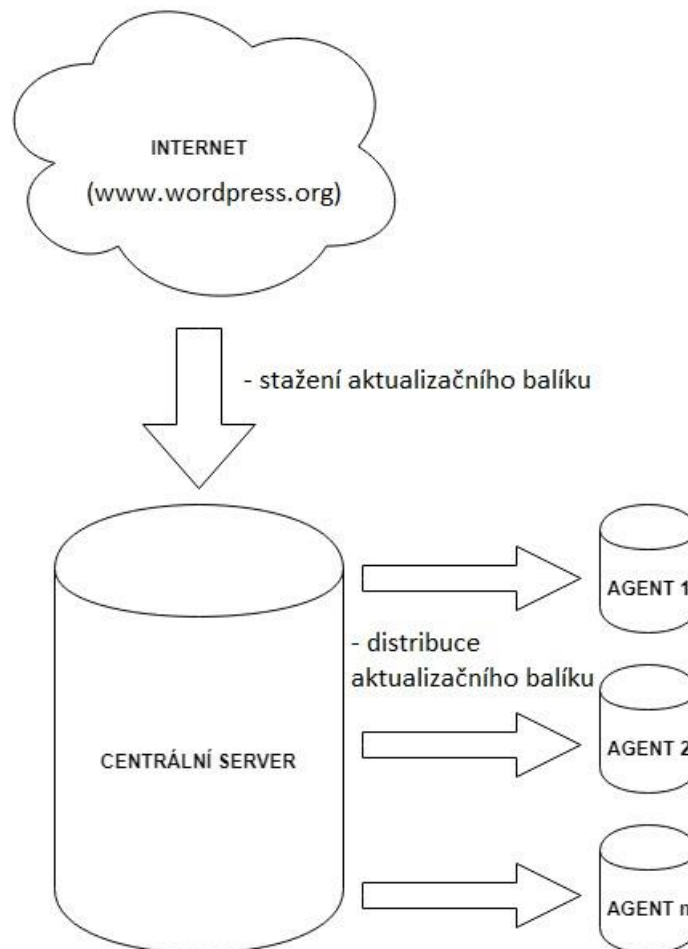
Zjednodušeně a obecně popsaná funkčnost navržené infrastruktury obnáší následující kroky:

- přihlášení do administračního systému
- stažení aktuální verze WordPress
- výběr agenta pro aktualizaci
- provedení aktualizace

Jednotlivé funkce celého systému budou do větší hloubky probrány v podkapitolách níže.

Infrastrukturu pro offline aktualizace systému WordPress lze dle výše uvedeného rozdělit na dvě základní části. První část tvoří centrální server pro řízení distribuce aktualizací, na kterém se nachází administrační rozhraní umožňující správcům systému provádět plně řízené aktualizace jednotlivých instalací systému WordPress. Druhá část je tvořena množinou agentů, tj. jednotlivými instalacemi systému WordPress, u kterých je nutno provádět pravidelnou aktualizaci.

Výsledkem navržené infrastruktury má být systém, který by umožňoval bezpečnou aktualizaci systému WordPress na jednotlivých agentech, na kterých je nainstalován.



Obr. 5 Schéma infrastruktury pro aktualizace⁸

2.1 Centrální server pro řízení distribuce aktualizací

Na centrálním serveru pro řízení distribuce aktualizací je umístěno administrační rozhraní, pomocí kterého jsou prováděny aktualizace jednotlivých instalací systému WordPress. Na serveru běží operačním systémem Linux Debian, pro distribuci aktualizací je tedy použit linuxový shell skript. Naprogramování administračního rozhraní bylo provedeno pomocí PHP frameworku Nette ve spolupráci s MySQL databází, která je použita pro ukládání dat. Toto rozhraní běží za pomoci webového serveru Apache.

⁸ Zdroj: vlastní

2.1.1 Administrační rozhraní

Administrační rozhraní, které je umístěno na centrálním serveru, bylo naprogramováno za účelem poskytnutí možnosti provádění aktualizací jednotlivých instalací systému WordPress, nad kterými by měl administrátor systému plný dohled. Jedná se o webové rozhraní, ke kterému má administrátor zabezpečený přístup prostřednictvím uživatelského jména a hesla.

Administrační rozhraní je rozděleno do čtyř sekcí, mezi nimiž lze přepínat pomocí navigačního menu. Jednotlivé části jsou popsány níže.

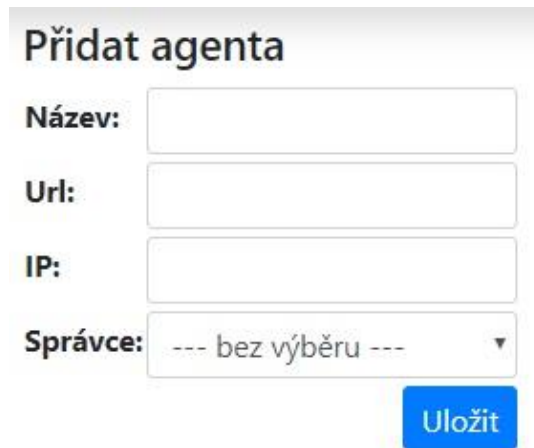
2.1.1.1 Přehled

Po úspěšném přihlášení do administračního rozhraní dojde k zobrazení sekce přehled, která je nastavena jako úvodní stránka. Jejím obsahem je přehled jednotlivých chronologicky seřazených WordPress instalací dle data a času vložení do systému administrace.

Vedle samotného seznamu jsou zobrazeny informace o nainstalovaných verzích systému WordPress je nainstalována na jednotlivých agentech, datu poslední provedené aktualizace a zda je daná verze aktuální.

2.1.1.2 Agenti

Součástí sekce agenti je formulář, který umožňuje do systému administrace ukládat informace ohledně jednotlivých instalací systému WordPress. Ukládají se informace, jako je název webu, na kterém běží daná instalace, IP adresa serveru, na kterém je daný web umístěn a datum vytvoření záznamu.



Přidat agenta

Název:

Url:

IP:

Správce: --- bez výběru --- ▼

Uložit

Obr. 6 Formulář pro vložení agenta⁹

Vedle formuláře pro vkládání výše uvedených informací je umístěna tabulka, v které je zobrazen přehled o uložených záznamech. V rámci této sekce je možno jednotlivé záznamy editovat a případně také odstranit ze systému.



#	Název	URL	IP adresa	Správce	Datum vytvoření	Datum aktualizace	Volby
1	Web 1	http://www.web1.cz	127.0.0.1	nepřiřazen	4. 5. 2019		 
2	Web 2	http://www.web2.cz	127.0.0.1	nepřiřazen	4. 5. 2019		 
3	Web 3	http://www.web3.cz	127.0.0.1	operator1	5. 5. 2019		 
4	Web 4	http://www.web4.cz	127.0.0.1	nepřiřazen	4. 5. 2019		 
5	Web 5	http://www.web5.cz	127.0.0.1	nepřiřazen	4. 5. 2019		 

Obr. 7 Tabulka s výpisem seznamu agentů¹⁰

⁹ Zdroj: vlastní

¹⁰ Zdroj: vlastní

2.1.1.3 Uživatelé

V sekci uživatelé je podobně jako v sekci agenti zobrazen formulář určený k ukládání informací o uživateli, kteří mají umožněn přístup do systému. Jedná se konkrétně o uživatelské jméno, heslo, uživatelskou roli a e-mail.

V této sekci je rovněž umístěna tabulka, která poskytuje přehled o uložených záznamech, u kterých lze také provádět editaci nebo odstranění ze systému.

2.1.1.4 Aktualizace

K samotnému řízení aktualizací jednotlivých WordPress instalací je určena sekce Aktualizace. V této sekci je zobrazen seznam agentů, který obsahuje informace, zda je nainstalována aktuální verze systému WordPress či nikoli. V případě, že není aktuální verze nainstalována, je u daného agenta zobrazeno tlačítko s nápisem aktualizovat, po jehož stisknutí dojde k aktualizaci na novou verzi systému WordPress.

2.1.1.5 Logy

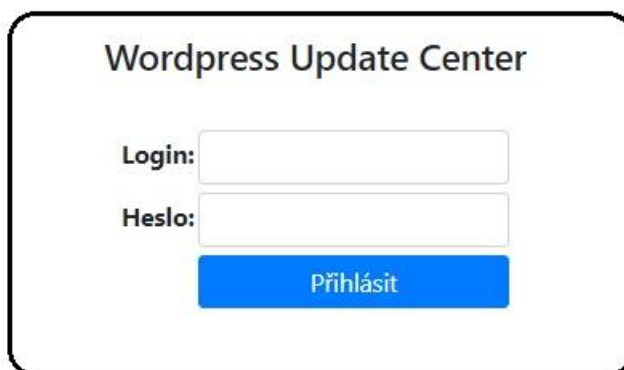
V rámci administračního systému centrálního serveru byla začleněna také sekce Logy. Zde jsou zobrazeny informace o všech jednotlivých aktualizacích, které byly provedeny. Po každé aktualizaci agenta na vyšší verzi systému WordPress je do databázové tabulky uloženo datum aktualizace, jméno a IP adresa uživatele, který aktualizaci provedl, název aktualizovaného agenta, URL a IP adresa webu, číslo původní a nově nainstalované verze. Uložené logy nelze prostřednictvím administračního rozhraní upravovat ani mazat. Uživatel, který má roli administrator, tak může zpětně dohledat informace o jakékoliv provedené aktualizaci.

#	Datum	Správce	Agent	Url	IP adresa	Původní verze	Nová verze	Timestamp	IP uživatele
1	8. 5. 2019	administrator	Web 1	http://www.web1.cz	212.115.64.87	5.1.1	5.2	2019-05-15 22:54:51	214.191.5.143
2	12. 5. 2019	admin2	Web 2	http://www.web2.cz	115.85.135.52	5.1.1	5.2	2019-05-15 22:56:10	221.162.2.201
3	12. 5. 2019	admin2	Web 3	http://www.web3.cz	184.52.67.113	5.1.1	5.2	2019-05-15 22:56:10	68.144.215.42

Obr. 8 Tabulka s výpisem logů¹¹

2.1.2 Zabezpečení administračního rozhraní

Přístup do administračního rozhraní je zabezpečen prostřednictvím přihlašovacího formuláře, do něž je nutno vložit správné uživatelské jméno a heslo. Oprávnění k jednotlivým operacím, jejichž provádění je umožněno konkrétním uživatelům, jsou přidělována prostřednictvím uživatelských rolí. Zabezpečení administračního rozhraní je tedy realizováno prostřednictvím procesů autentizace a následné autorizace uživatelů.



Obr. 9 Přihlašovací formulář do administračního systému

2.1.2.1 Autentizace uživatele

Proces autentizace uživatele, tj. ověření identity uživatele, je realizován prostřednictvím přihlašovacího formuláře, do kterého je zapotřebí vložit přihlašovací jméno a heslo. Po

¹¹ Zdroj: vlastní

zadání těchto údajů a následném odeslání formuláře dojde k ověření jejich správnosti porovnáním oproti údajům uložených v databázové tabulce users.

V případě, že se zadané údaje neshodují s údaji uloženými v databázi, přístup do administračního rozhraní není uživateli umožněn a následně dojde k upozornění, že byly zadány neplatné přihlašovací údaje.

Pokud se zadané údaje shodují s údaji uloženými v databázi, dojde k umožnění přístupu daného uživatele do administračního systému.

2.1.2.2 *Princip zabezpečení uživatelského hesla*

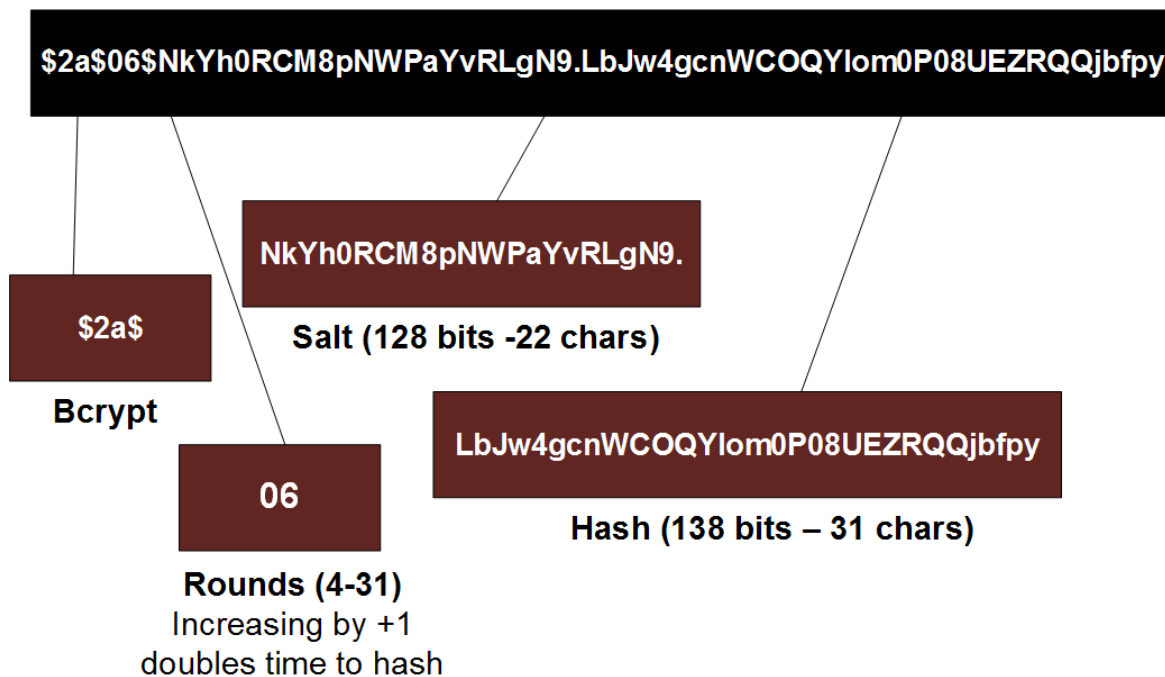
Jedním z údajů potřebných k přihlášení uživatele do administračního systému je uživatelské heslo. Jedná se o řetězec znaků, který je prostřednictvím formuláře uložen do databázové tabulky users.

Při návrhu administračního rozhraní bylo zapotřebí dbát také na zabezpečení uživatelských dat. Z tohoto důvodu byl při realizaci systému přihlašování uživatelů kladen důraz na zvýšené zabezpečení hesla, k čemuž byl zvolen specifický způsob šifrování pomocí kryptografické hašovací funkce Bcrypt.

Šifrovací algoritmus Bcrypt

Bcrypt je funkce pro odvození klíče určená pro hesla a je založena na šifře blowfish¹². Při vytváření hashe se střídavě používá náhodně generovaná kryptografická sůl, která zneumožňuje provádění útoků pomocí duhové tabulky. Výhodou použití Bcrypt algoritmu je iterativní hashování, čímž dochází ke zpomalení a zvyšuje se tak ochrana proti brute-force.

¹² LEBLANC, J., T. MESSERCHMIDT. *Identity and Data Security for Web Development: Best Practices*. Sebastopol: O'Reilly Media, 2016. ISBN 978-1-491-93701-3



Obr. 10 Ukázka struktury Bcrypt hashe¹³

2.1.2.3 Autorizace uživatele

Po úspěšné autentizaci uživatele proběhne proces autorizace, tedy udělení oprávnění k vykonávání určitých operací aktuálně přihlášenému uživateli. Udělování oprávnění je v rámci administračního systému řešeno prostřednictvím uživatelských rolí.

Každý uživatel systému má tak přidělenou jednu z uživatelských rolí, přičemž každá tato role má přidělena konkrétní práva k vykonávání určitých operací.

Informace o roli přidělené uživateli je uložena v tabulce users ve sloupci role_id.

Nastavení jednotlivých rolí je uloženo v tabulce roles, kde jsou v jednotlivých sloupcích nastavena práva, která buďto povolují nebo zakazují přístup a provádění operací v jednotlivých částech systému.

¹³ Dostupné z: <https://asecuritysite.com/encryption/bcrypt>

2.1.2.4 Role a oprávnění

Administrátor

Nejvyšší oprávnění v rámci administračního systému má role administrátor. Uživateli, který má přidělenou roli administrátora, jsou udělena oprávnění pro přístup a provádění operací v sekcích agenti, uživatelé, aktualizace a logy.

V sekci agenti je tomuto uživateli umožněno přidat nového agenta, editovat nebo odebrat stávajícího agenta.

Stejně operace jsou uživateli umožněny v sekci uživatelé, tzn. přidat nového uživatele, editovat nebo odebrat.

V sekci aktualizace může uživatel s rolí administrátora provádět aktualizace WordPress instalací na všech agentech.

Informace, které jsou zobrazeny v sekci logy, jsou určeny pouze pro čtení.

Admin

Uživatel, kterému byla přidělena role Admin, jsou udělena oprávnění pro přístup a provádění operací v sekcích agenti, aktualizace a logy.

V sekci agenti je tento uživatel oprávněn přidat nového agenta, editovat nebo odebrat stávajícího agenta.

V sekci aktualizace je tomuto uživateli dovoleno provádět aktualizace systému WordPress na všech agentech.

Uživatel s přidělenou rolí Admin nemá přístup do sekce Uživatelé, není tudíž oprávněn vytvářet nové uživatele, upravovat ani mazat již existující uživatele.

V sekci Logy je uživatel s rolí Admin oprávněn prohlížet logy, které jsou všechny určeny pouze ke čtení.

Operátor

Role Operátor disponuje pouze omezenými právy k přístupu do jednotlivých sekcí administrativního systému. Uživateli, kterému byla tato role přidělena, je po přihlášení do systému umožněno přistupovat pouze do sekce Přehled a Aktualizace. Do zbylých sekcí agenti, uživatelé a logy je uživateli s rolí operátor zamezen přístup.

Specifická práva jsou tomuto uživateli nastavena také k provádění operací v sekci Aktualizace. Jsou zde zobrazeni pouze ti agenti, jimž je daný uživatel přidělen jako správce. Tímto nemůže dojít k tomu, aby uživatel s rolí Operátor aktualizoval agenta, jemuž nebyl přidělen jako správce.

2.1.2.5 *Nette framework*

Za účelem zvýšení efektivity kódu a zabezpečení aplikace byl pro vývoj administrativního rozhraní zvolen Nette Framework. Jedná se o PHP framework od českého programátora Davida Grudla, který je navržen tak, aby bylo možno efektivně provádět validaci uživatelských vstupů a zajistit bezpečnost aplikace proti různým útokům.

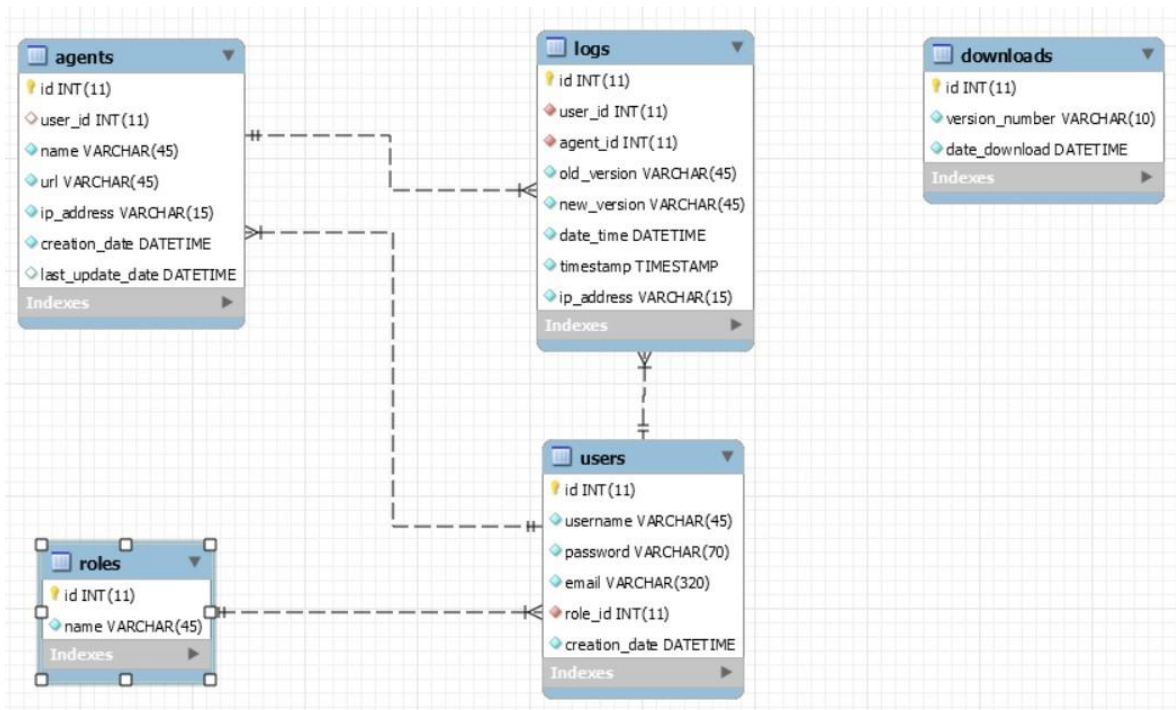
Nette disponuje několika samostatně použitelnými komponentami. Za zmínku jistě stojí šablonovací systém Latte, který zajišťuje oddělení PHP kódu od HTML. Dále je to ladící nástroj s názvem Tracy a databázová vrstva Database Explorer, která usnadňuje získávání dat z databáze bez nutnosti psaní SQL dotazů. Prostřednictvím Database Exploreru lze provádět také výběry přes více tabulek nebo agregaci výsledků.

2.1.3 **Databázové schéma administrativního rozhraní**

K tomu, aby mohlo být na centrálním serveru zprovozněno administrativní rozhraní, bylo zapotřebí použít databázi, do které by bylo možno ukládat potřebná data. Za tímto účelem byla zvolena databáze MySQL, jejíž použití je vhodné právě při realizaci webové aplikace, která byla naprogramována s využitím programovacího jazyka PHP. Tento skriptovací jazyk disponuje důležitými funkcemi, které zajišťují bezproblémovou komunikaci s databázovým serverem MySQL. Výhodou použití MySQL databáze je podpora cizích klíčů, transakcí a poddotazů, dále pak dostupnost efektivních nástrojů pro správu databází jako

např. webové rozhraní phpMyAdmin nebo vizuální nástroj MySQL Workbench od softwarové společnosti Oracle.

Součástí databáze, která byla navržena pro účely administračního rozhraní, jsou jednotlivé tabulky, do kterých jsou mimo jiné ukládána potřebná data. Jedná se o tabulky agents, roles, users a downloads.



Obr. 11 Databázové schéma administračního rozhraní¹⁴

2.2 Agenti aktualizací

Agenti aktualizací představují jednotlivé instalace systému WordPress. Každý z těchto agentů může být umístěn na samostatném severu nezávisle na ostatních agentech. Identifikace jednotlivých agentů je zajištěna prostřednictvím IP adres serverů, na kterých jsou agenti umístěni. Ke snazšímu rozpoznání agenta uživatelem v rámci administračního sys-

¹⁴ Zdroj: vlastní

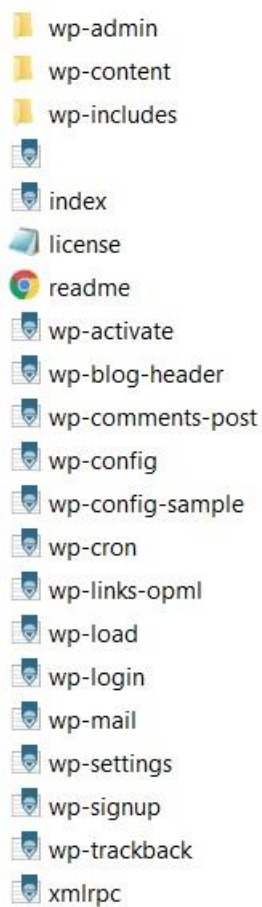
tému má každý přidělen název, který by se měl lišit od názvů ostatních agentů, aby nedošlo k jejich záměně.

2.2.1 Server

Servery, na kterých jsou umístěny WordPress instalace, běží na operačním systému Debian. Zde je také zprovozněno běhové prostředí potřebné k provozování webu. Za tímto účelem je použita kombinace Apache HTTP server, PHP a MySQL. Popis a postup instalace jednotlivých komponent je uveden ve čtvrté kapitole této práce.

2.2.2 Princip činnosti systému WordPress

Redakční systém WordPress je napsán v jazyce PHP a MySQL. K tomu, aby správně fungoval, je zapotřebí, aby byly na serverovém disku uloženy jednak veškeré zdrojové soubory, a dále musí být vytvořena databáze, která je potřebná k ukládání systémových dat a samotného obsahu webu. Rovněž je zapotřebí, aby byl zprovozněn a správně nakonfigurován webový server. V podkapitolách níže je graficky znázorněna základní adresářová struktura systému Wordpress a databázové schéma systému WordPress.

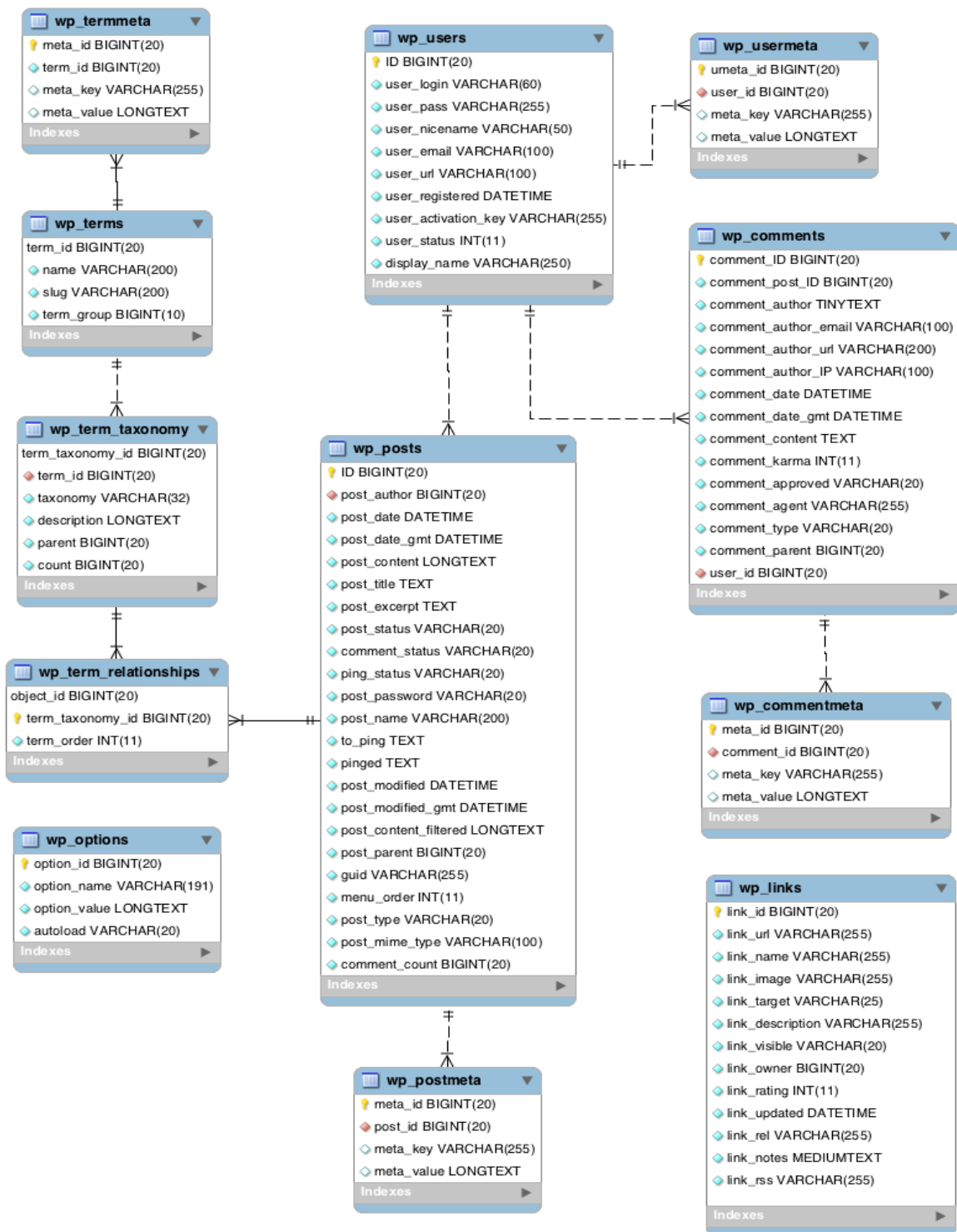


Obr. 12 Adresářová struktura složky wp-content¹⁵

2.2.2.1 Databázové schéma systému WordPress

Přehled základních tabulek, které jsou součástí databáze systému WordPress spolu se znázorněním relací mezi nimi, je uveden ER-diagramu níže.

¹⁵ Zdroj: vlastní



Obr. 13 Databázové schéma systému WordPress¹⁶

¹⁶ Dostupné z: https://codex.wordpress.org/Database_Description

3 NÁVRH ZPŮSOBU ZABEZPEČENÍ KOMUNIKACE MEZI AGENTY A CENTRÁLNÍM SERVEREM

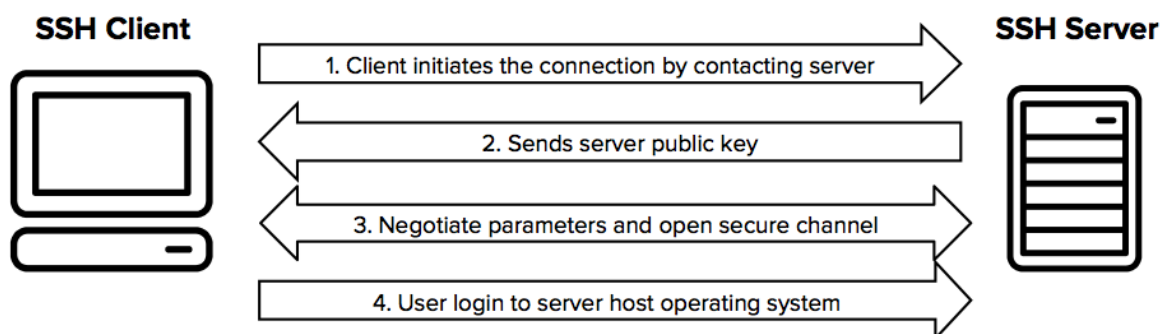
Důležitou součástí komunikace mezi agenty a centrálním serverem je zabezpečení. To je realizováno prostřednictvím zabezpečeného komunikačního protokolu SSH s využitím RSA šifrovacích klíčů. Použití SSH protokolu znemožňuje odposlech při přenosu dat, zajišťuje šifrování přenášených dat, zachování jejich integrity a autentizaci účastníků komunikace s využitím RSA klíčů.

3.1 SSH protokol

Protokol SSH nabízí zabezpečené a plně šifrované spojení¹⁷ a umožňuje provádět níže uvedené operace:

- ověřování identity hostitele
- šifrování datového přenosu
- vytvoření šifrovaného tunelu pro přenos dat

Komunikace prostřednictvím SSH protokolu funguje na principu použití asymetrických šifer. Pro použití asymetrických šifer je potřeba dvou klíčů – veřejného a tajného¹⁸.



¹⁷ KYSELA, Martin. *333 tipů a triků pro Linux*. Brno: Computer Press, 2007. ISBN 80-722-6866-X.

¹⁸ KRČMÁŘ, Petr. *Linux – Tipy a triky pro bezpečnost*. Praha: Grada, 2004. ISBN 80-247-0812-4.

Obr. 14 Schéma činnosti SSH protokolu¹⁹

3.2 RSA šifrování

RSA algoritmus využívá k šifrování dvojici klíčů, tj. veřejný a privátní. Při komunikaci dvou uživatelů je tedy zapotřebí dvou párů těchto dvojic.

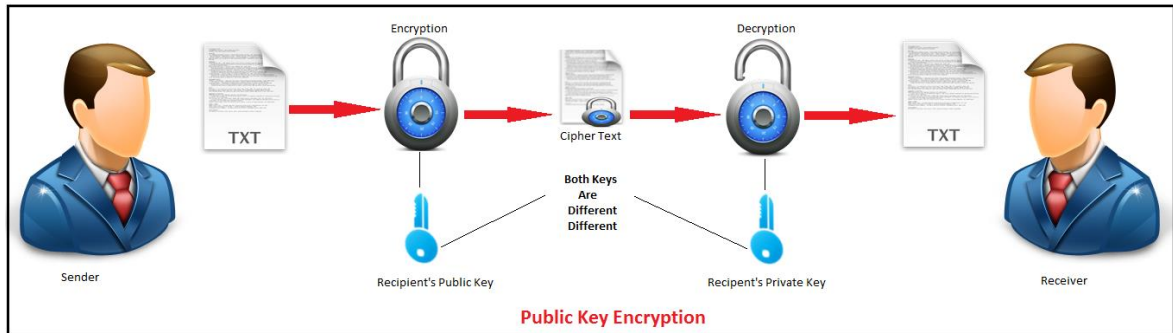
Při použití RSA algoritmu se využívá skutečnosti, že zpráva zašifrovaná veřejným klíčem nelze pomocí tohoto klíče dešifrovat. K úspěšnému dešifrování zprávy lze použít výhradně privátního klíče, který je uložen u uživatele, pro nějž byl tento klíč vygenerován.

Šifrování komunikace mezi dvěma uživateli tedy probíhá následujícím způsobem:

1. uživatel A vygeneruje veřejný klíč, který pošle uživateli B
2. uživatel A vygeneruje privátní klíč, který si ponechá uložen
3. uživatel A pošle veřejný klíč uživateli B
4. uživatel B zašifruje zprávu pomocí veřejného klíče uživatele A
5. uživatel B zašle zašifrovanou zprávu uživateli A
6. uživatel A dešifruje zašifrovanou zprávu privátním klíčem

Stejným způsobem probíhá komunikace směrem od uživatele A k uživateli B, kdy uživatel B poskytne uživateli A svůj veřejný klíč za účelem šifrování odesílaných zpráv a následně dešifruje zašifrované zprávy rovněž pomocí svého privátního klíče.

¹⁹ Dostupné z: <https://www.ssh.com/ssh/>



Obr. 15 Schéma činnosti při generování RSA klíčů²⁰

3.3 Způsob spojení mezi centrálním serverem a agenty

Spojení mezi centrálním serverem a agenty bude probíhat v závislosti na akci přihlášeného uživatele v administračním systému na centrálním serveru. V sekci Aktualizace, kde se nachází výpis jednotlivých agentů, se bude u každého z nich nacházet tlačítko s názvem Aktualizovat. Po kliknutí na toto tlačítko bude proveden shell skript za využití Secure Shell2 rozšíření jazyka PHP, který bude zajišťovat zabezpečené připojení k danému agentovi prostřednictvím SSH. Za účelem ověření identity byl vygenerován a naimplementován pár RSA klíčů. Po úspěšné autentizaci obou účastníků spojení pomocí RSA klíčů bude umožněna vzájemná komunikace, aby bylo možno provést vzdálenou aktualizaci systému WordPress na daném serveru.

²⁰ Dostupné z: <http://itechtuts.com/implement-rsa-algorithm-c/>

II. PRAKTICKÁ ČÁST

4 TESTOVACÍ IMPLEMENTACE NAVRŽENÉHO ŘEŠENÍ

V rámci praktické části diplomové práce proběhla testovací implementace navrženého řešení. Byl nainstalován centrální server, na kterém byl zprovozněn naprogramovaný administrační systém, jehož prostřednictvím má být řízena komunikace s jednotlivými agenty za účelem aktualizací systému WordPress. Dále byly zřízeny tři servery, na kterých byla následně provedena instalace různých verzí systému WordPress, a to vždy nižší, než je aktuální verze (5.2). Instalace na jednotlivých serverech proběhla vzdáleně prostřednictvím VPN a SSH připojení.

4.1 Centrální server

Před zprovozněním centrálního serveru byla vytvořena instalace operačního systému Debian na virtualizačním software VMWare. Tento postup umožnil provést efektivnější a pohodlnější instalaci jednotlivých komponent a všech nastavení systému na lokálním stroji. Po otestování funkčnosti všech nainstalovaných komponent potřebných pro běh administračního systému mohl být takto připravený operační systém následně nahrán na testovací server.

4.1.1 Operační systém

Pro běh centrálního serveru byl použit operační systém Debian verze 9.7.0. Jedná se o jednu z nejstarších linuxových distribucí. Debian je svobodný operační systém, který poskytuje více než 51000 balíčků s programy a dokumentací, které jsou připraveny pro snadnou instalaci²¹. Na základě možnosti využití mnoha z těchto volně dostupných programů a komponent potřebných pro běh administračního systému na centrálním serveru byl Debian zvolen jako operační systém vhodný k použití pro stanovené účely při plnění úkolů dle zadání diplomové práce.

²¹ Index Debian. *Debian* [online]. © 2018 [cit. 2018-05-17]. <http://www.debian.org/index.cs.html>

4.1.2 Další komponenty

Za účelem zprovoznění administračního systému na centrálním serveru bylo zapotřebí instalace několika dalších komponent, které jsou detailněji popsány níže.

4.1.2.1 *Apache HTTP server*

Jedná se o softwarový webový server, který je k použití dostupný zdarma a běží na 67% všech webových serverů na světě²². Tento software je zodpovědný za zpracování HTTP požadavků od klienta, na základě nichž odesílá odpovědi zpět na stranu klienta. V rámci implementace navrženého řešení je Apache HTTP server nepostradatelnou součástí pro fungování administračního systému na centrálním serveru.

4.1.2.2 *PHP*

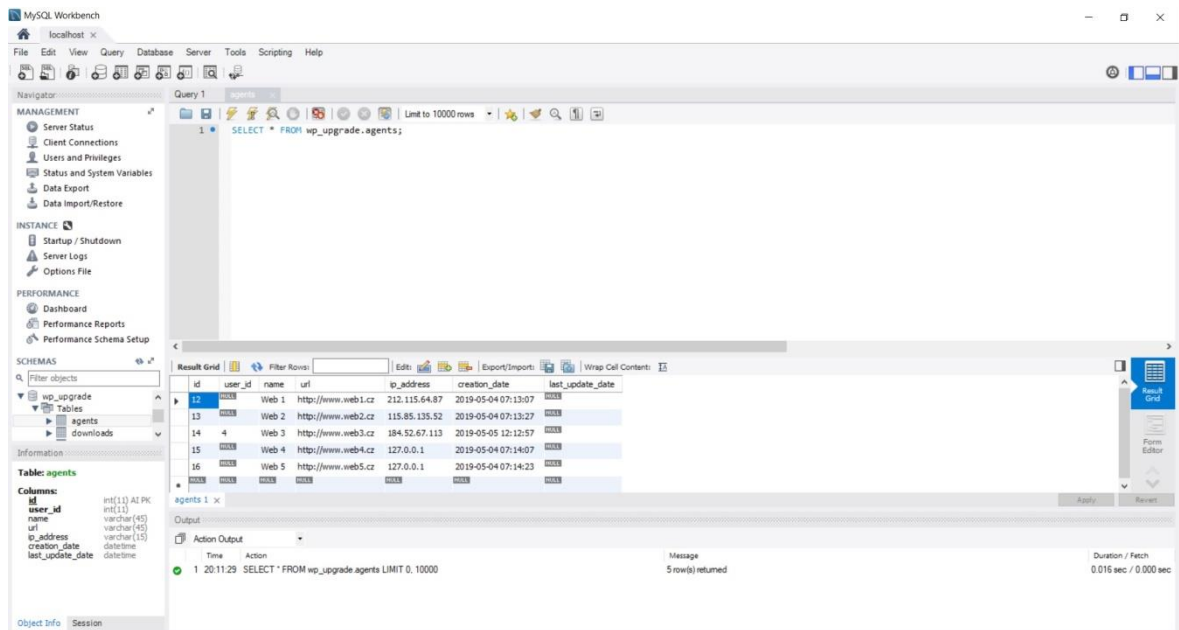
PHP je tzv. skriptovací programovací jazyk, který se využívá nejčastěji k programování webových aplikací a dynamických webových stránek. Naprogramované PHP skripty se provádějí na straně serveru. Nejdříve je vyslán dotaz ze strany uživatele, následně dochází ke zpracování požadavku a poté k odeslání výsledku zpět k uživateli. Součástí PHP nainstalovaného na serveru je také knihovna pro podporu Apache serveru a rozšíření nutné pro společné fungování s MySQL databází, která je rovněž součástí instalace na centrálním serveru.

4.1.2.3 *MySQL databáze*

MySQL je systém relační databáze, která není závislá na platformě, na níž je nainstalována, a je poskytována bezplatně. Je často nasazována právě v kombinaci operačního systému Linux společně s Apache serverem a jazykem PHP, čímž je tvořen webový server. Komunikace s MySQL databází probíhá prostřednictvím jazyka SQL. MySQL podporuje

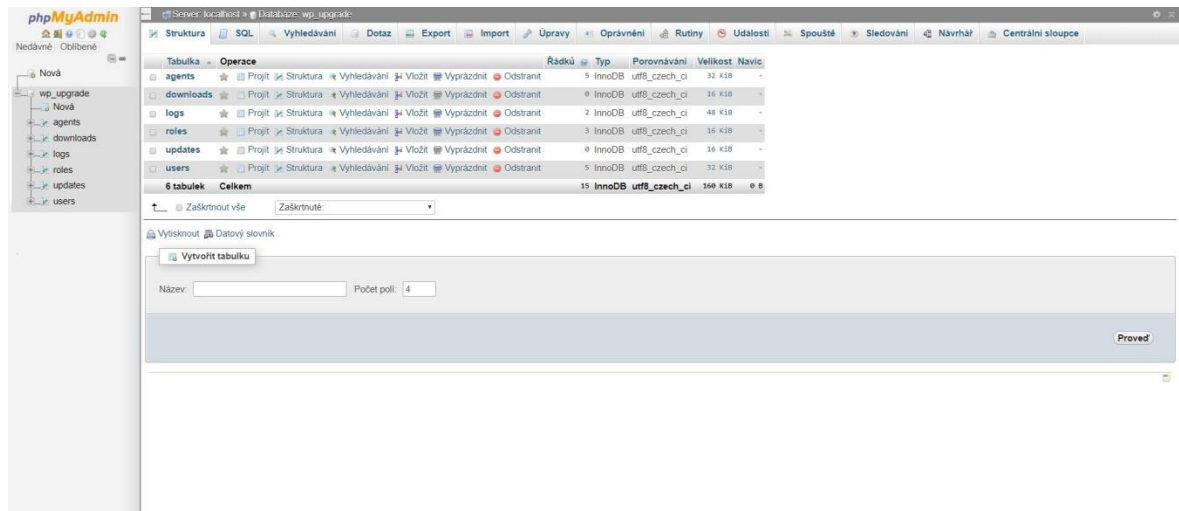
²² What is: Apache. In: Wpbeginner [online]. [cit. 2019-05-17]. Dostupné z: <http://www.wpbeginner.com/glossary/apache/>

použití cizích klíčů, transakcí, fulltextového vyhledávání, vykonávání triggerů, vytváření poddotazů a další užitečné funkce. V rámci administračního systému je komunikace s databází prováděna prostřednictvím jazyka PHP. Při tvorbě a správě databáze byl použit software MySQL Workbench a v některých případech bylo použito webové rozhraní php-MyAdmin, které je napsáno v jazyce PHP.



Obr. 16 Ukázka uživatelského prostředí softwaru MySQL Workbench²³

²³ Zdroj: vlastní

Obr. 17 Ukázka uživatelského prostředí softwaru phpMyAdmin²⁴

4.2 Agenti

Na jednotlivé agenty bylo zapotřebí nainstalovat systém WordPress v různých verzích. Toto bylo provedeno vzdáleně prostřednictvím VPN a SSH připojení. Samotná instalace byla realizována pomocí terminálu a Shell příkazů, které jsou stručně popsány v následujících podkapitolách.

4.2.1 Operační systém

V případě serverů, na kterých běží WordPress instalace, byl nasazen operační systém Linux Debian verze 9. Za účelem realizace testovací implementace bylo zapotřebí doinstalovat potřebné komponenty, aby mohl být zprovozněn webový server a nainstalován systém WordPress. Instalace komponent a WordPressu probíhala vzdáleně přes SSH připojení a dále prostřednictvím příkazů spouštěných přes terminál.

²⁴ Zdroj: vlastní

4.2.2 Další komponenty

V rámci testovací implementace byly na operační systém Debian doinstalovány níže uvedené komponenty potřebné pro provoz webového serveru a instalaci systému WordPress.

4.2.2.1 Apache HTTP server

Postup instalace:

```
apt install apache2
```

4.2.3 PHP

Ukázka instalačního příkazu:

```
apt install php7.2-common php7.2-cli php7.2-mysql php7.2 libapache2-mod-php
```

4.2.4 MySQL

Ukázka instalačního příkazu:

```
apt install mysql-server
```

Ukázka instalačního příkazu pro vytvoření databáze, uživatele a nastavení oprávnění k databázi:

```
create database dbname;
```

```
grant usage on *.* to username@localhost identified by 'password';
```

```
grant all privileges on dbname.* to username@localhost;
```

4.2.5 Instalace zdrojových souborů systému WordPress

Po nainstalování komponent potřebných pro spuštění webového serveru byla provedena instalace zdrojových souborů systému WordPress. Nejprve byl stažen archiv s instalačním

balíkem, který byl následně dekomprimován a umístěn do výchozího adresáře webového serveru.

Ukázka příkazů pro stažení a instalaci zdrojových souborů systému Wordpress:

wget http://wordpress.org/wordpress-4.8.tar.gz - stažení instalačního balíku

tar xzf wordpress-4.8.tar.gz – dekomprimace instalačního balíku

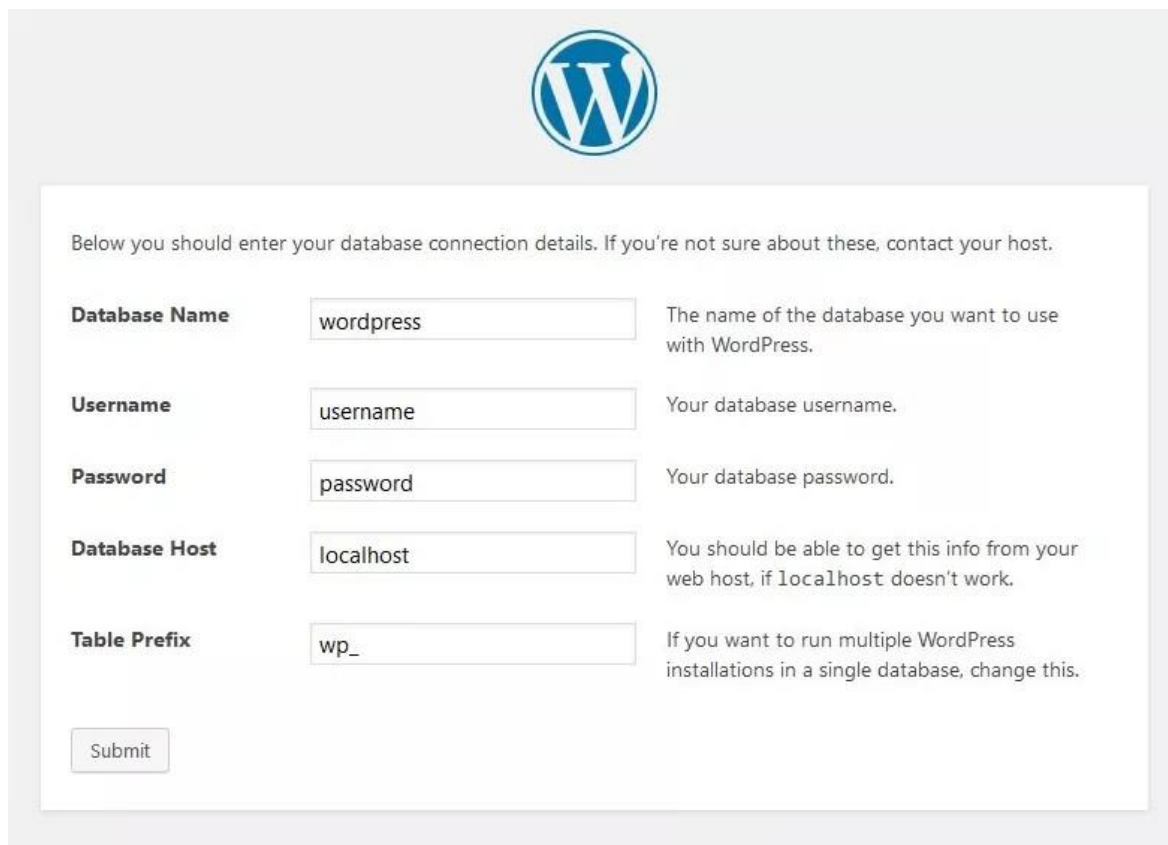
mv wordpress/ ./* - přesunutí obsahu adresáře wordpress do nadřazeného adresáře

rmdir ./wordpress/ ./* - odstranění již prázdného adresáře wordpress

rm -f wordpress-4.8.tar.gz – odstranění instalačního balíku

4.2.6 Instalace systému WordPress

Samotná instalace systému WordPress, tj. vytvoření konfiguračního souboru, databázového spojení, databázových tabulek a uložení dat do databáze proběhlo za pomoci instalačního průvodce. Ten byl spuštěn v internetovém prohlížeči po zadání IP adresy serveru. Po spuštění instalačního průvodce byly zadány přihlašovací údaje k databázi, aby bylo možné vytvořit instalaci systému WordPress úspěšně dokončit.



The image shows the WordPress installation database configuration screen. At the top center is the WordPress logo. Below it, a text box reads: "Below you should enter your database connection details. If you're not sure about these, contact your host." The form contains five input fields, each with a label and a description:

Field Label	Value	Description
Database Name	wordpress	The name of the database you want to use with WordPress.
Username	username	Your database username.
Password	password	Your database password.
Database Host	localhost	You should be able to get this info from your web host, if localhost doesn't work.
Table Prefix	wp_	If you want to run multiple WordPress installations in a single database, change this.

At the bottom left of the form is a "Submit" button.

Obr. 18 Ukázka instalačního průvodce systému WordPress²⁵

4.2.7 Ověření funkčnosti instalace

Úspěšná instalace webového serveru a systému WordPress byla ověřena jak spuštěním instalačního průvodce, tak následným zadáním IP adresy do anonymního okna internetového prohlížeče, ve kterém byl následně zobrazen obsah webu, který je součástí základní instalace systému WordPress.

²⁵ Zdroj: vlastní

5 TESTOVÁNÍ FUNKČNOSTI ŘEŠENÍ NA POSKYTNUTÝCH SCÉNÁŘÍCH AKTUALIZACÍ

V návaznosti na provedenou implementaci navrženého řešení proběhlo testování funkčnosti dle poskytnutého scénáře aktualizací. V této kapitole je popsán postup testování a jeho výsledek. V první fázi proběhlo testování funkčnosti jednotlivých sekcí administračního systému, ve druhé fázi pak byly testovány aktualizace systému WordPress na jednotlivých agentech.

5.1 Testování funkčnosti administračního rozhraní

Součástí serveru je administrační rozhraní, jehož provoz je nutný k provádění samotných aktualizací jednotlivých agentů, na kterých běží systém WordPress. Toto rozhraní je umístěno na centrálním serveru, kde také proběhlo jeho testování.

Administrační rozhraní je umístěno na adrese 10.5.11.10/wp_upgrade/www/

5.1.1 Přihlašovací stránka

Po zadání výše uvedené adresy došlo k zobrazení přihlašovací stránky administračního systému. Tímto krokem bylo ověřeno, že fungují jednotlivé komponenty webového serveru, že bylo správně provedeno nastavení Apache HTTP serveru a nastavení připojení k databázi.

5.1.2 Přihlášení uživatele

Po zadání přihlašovacích údajů do formuláře a jeho následném odeslání došlo k úspěšnému přihlášení uživatele. Pro účely testování byl vytvořen uživatel s loginem **administrator** a heslem **123456**. Tento uživatel má přidělena administrátorská práva, může tedy v rámci administračního rozhraní spravovat ostatní uživatele.

5.1.3 Omezení práv jednotlivých uživatelských rolí

Otestování funkčnosti omezení práv jednotlivých uživatelů bylo provedeno vytvořením několika uživatelů, kterým byly přiřazeny různé role. V závislosti na těchto rolích se danému uživateli po přihlášení do systému zobrazovaly pouze určité sekce administrace, jak je popsáno ve druhé kapitole.

5.1.4 Funkce v jednotlivých sekcích administračního systému

Po přihlášení do administračního systému byla prověřena funkčnost jeho jednotlivých sekcí. Po kliknutí na danou sekci levého menu byl obsah každé ze sekcí správně zobrazen. Funkčnost komponent jednotlivých sekcí, jako jsou formuláře a tabulky s ovládacími prvky, byla otestována rovněž úspěšně. Bylo vyzkoušeno přidávání agentů, jejich editace a mazání. Rovněž byly vyzkoušeny funkce přidávání uživatelů, jejich editace a mazání.

5.2 Testování aktualizace agentů

Testování funkčnosti aktualizace jednotlivých agentů spočívá funkce Aktualizovat v sekci Aktualizace v rámci administračního rozhraní. Zde má následně na pozadí dojít ke vzdálené aktualizaci systému WordPress na vyšší verzi.

5.2.1 Navázání spojení s agentem přes SSH

K navázání spojení mezi centrálním serverem a agenty byla snaha využít Secure Shell2 rozšíření jazyka PHP. Instalace tohoto rozšíření na centrálním serveru proběhla úspěšně, avšak následně se již nepodařilo správně nakonfigurovat webový server tak, aby bylo možno jeho funkce použít při volání v Nette frameworku, resp. v jazyce PHP. Při pokusu o volání funkce `ssh2_connect` došlo k pádu aplikace a zobrazení Fatal Error. Z tohoto důvodu nebylo možno uskutečnit spojení mezi centrálním serverem a agentem přes SSH.

ZÁVĚR

Cílem této diplomové práce bylo vytvořit portál pro offline aktualizaci WordPressu. S vedoucím práce byly průběžně konzultovány jednotlivé kroky při návrhu infrastruktury, kdy bylo dohodnuto, že offline aktualizace budou spravovány pomocí administračního systému naprogramovaného v jazyce PHP při použití frameworku Nette, který bude zprovozněn na centrálním serveru. Jako způsob komunikace a zabezpečení spojení mezi servery a agenty, na kterých je nainstalován systém WordPres, bylo zvoleno SSH připojení, které by mělo být ověřeno prostřednictvím dvojice RSA klíčů. Jedním z úkolů při tvorbě této práce bylo provedení testovací implementace navrženého řešení a následné otestování funkčnosti tohoto řešení.

Úspěšně se podařilo naprogramovat administrační systém, který byl rovněž implementován na centrálním serveru. Na serveru, jehož provoz byl zajištěn využitím operačního systému Debian, bylo nutno provést instalaci webového serveru, aby bylo možno funkčnost administračního systému otestovat. Testování funkcí v jednotlivých sekcích systému proběhlo vcelku úspěšně.

Při provádění testovací implementace bylo taktéž zapotřebí nainstalovat běhové prostředí na tři různé servery, na které byly následně nainstalovány různé verze systému WordPress, což bylo rovněž úspěšně provedeno.

V poslední fázi měla být ověřena funkčnost aktualizací systému WordPress na jednotlivých serverech. Testování této funkčnosti neproběhlo úspěšně a to z toho důvodu, že se nepodařilo navázat spojení mezi centrálním serverem a agenty přes SSH s využitím ověřování pomocí RSA klíčů. K tomuto účelu bylo zamýšleno využít rozšíření Secure Shell2 jazyka PHP, avšak po jeho instalaci na centrální server se nepodařilo správně nakonfigurovat webový server tak, aby funkce daného rozšíření v jazyce PHP při použití Nette frameworku běžely bezchybně.

Z výše uvedeného vyplývá, že navržený způsob zabezpečení komunikace mezi agenty a centrálním serverem nelze při současné konfiguraci komponent na centrálním serveru použít.

SEZNAM POUŽITÉ LITERATURY

Monografie

- [1] WILLIAMS B., D. DAMSTRA a H. STERN. Professional WordPress: design and development. Indiana-polis: John Wiley & Sons, 2013. ISBN 978-1-118-44227-2.
- [2] KUDLÁČEK, Luboš. WordPress: podrobný průvodce tvorbou a správou webů. Brno: Computer Press, 2013. ISBN 978-80-251-2734-6.
- [3] LEBLANC, J., T. MESSERCHMIDT. Identity and Data Security for Web Development: Best Practi-ces.Sebastopol: O'Reilly Media, 2016. ISBN 978-1-491-93701-3.
- [4] KYSELA, Martin. 333 tipů a triků pro Linux. Brno: Computer Press, 2007. ISBN 80-722-6866-X.
- [5] KRČMÁŘ, Petr. Linux – Tipy a triky pro bezpečnost. Praha: Grada, 2004. ISBN 80-247-0812-4.

Internetové zdroje

- [6] Updating WordPress. In: WordPress [online]. [cit. 2019-05-17]. Dostupné z: https://codex.wordpress.org/Updating_Wordpress.
- [7] Index Debian. Debian [online]. © 2018 [cit. 2018-05-17]. <http://www.debian.org/index.cs.html>.
- [8] What is: Apache. In: Wpbeginner [online]. [cit. 2019-05-17]. Dostupné z: <http://www.wpbeginner.com/glossary/apache/>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

PHP	Hypertext Preprocessor.
MySQL	Význam druhé zkratky.
Bcrypt	Význam třetí zkratky.
URL	Uniform Resource Locator
IP	Internet Protocol
ZIP	Souborový formát pro kompresi dat
GZIP	Souborový formát pro kompresi dat
HTML	Hypertext Markup Language
SQL	Structure Query Language
SSH	Secure Shell
RSA	iniciály autorů Rivest, Shamir, Adleman – asymetrická šifra
HTTP	Hypertext Transfer Protocol

SEZNAM OBRÁZKŮ

<i>Obr. 1 Zdrojový kód souboru wp-config.php.....</i>	15
<i>Obr. 2 Ukázka možnosti aktualizace prostřednictvím administračního rozhraní</i>	17
<i>Obr. 3 Ovládací prvky pluginu Easy Updates Manager</i>	19
<i>Obr. 4 Ukázka možnosti aktivace / deaktivace pluginu</i>	20
<i>Obr. 5 Schéma infrastruktury pro aktualizace.....</i>	23
<i>Obr. 6 Formulář pro vložení agenta.....</i>	25
<i>Obr. 7 Tabulka s výpisem seznamu agentů.....</i>	25
<i>Obr. 8 Tabulka s výpisem logů</i>	27
<i>Obr. 9 Přihlašovací formulář do administračního systému.....</i>	27
<i>Obr. 10 Ukázka struktury Bcrypt hashe.....</i>	29
<i>Obr. 11 Databázové schéma administračního rozhraní.....</i>	32
<i>Obr. 12 Adresářová struktura složky wp-content</i>	34
<i>Obr. 13 Databázové schéma systému WordPress</i>	35
<i>Obr. 14 Schéma činnosti SSH protokolu</i>	37
<i>Obr. 15 Schéma činnosti při generování RSA klíčů.....</i>	38
<i>Obr. 16 Ukázka uživatelského prostředí softwaru MySQL Workbench</i>	42
<i>Obr. 17 Ukázka uživatelského prostředí softwaru phpMyAdmin</i>	43
<i>Obr. 18 Ukázka instalačního průvodce systémem WordPress</i>	46

SEZNAM PŘÍLOH

P I Zdrojové kódy administračního systému

P II Diplomová práce v elektronické podobě (umístěno na CD)