

Obsahová poznámka

Nasledujúce strany boli vypracované na 200%+ pôvodného zadania pre neskoršiu pomoc ostatným študentom predmetu ZPS (Základy počítačových systémov) s kontrolou a doplnením vlastných riešení.

@iairu // Ondrej Špánik

2020-03-29

Ondrej Špánik

Zadanie 6 – HTTP Server

Realizovanie úloh v GNU/Linux (Debian)

Úvod – Inštalácia Apache

Základný Apache HTTP server (zatiaľ bez PHP, MySQL, ...) sa inštaluje cez

```
sudo apt install apache2
```

Všetky konfiguračné súbory sú v `/etc/apache2`, tie podstatné k úlohám budem postupne vypisovať nižšie.

1. Vypnutie auto-spúšťania pri štarte

Linux, resp. Debian používa **systemctl** pre manažment služieb ako Apache a iné. Pomocou tohto príkazu sa dá mimo bežného **reštartovania/štartu/stopu** (`sudo systemctl restart/start/stop apache2`) aj **vypnúť/zapnúť automatické spúšťanie** (`sudo systemctl disable/enable apache2`), čiže:

```
sudo systemctl disable apache2
```

2. Zmena adresára stránok HTTP servera

Adresáre pre jednotlivé porty sa definujú v súbore:

```
/etc/apache2/sites-enabled/000-default.conf
```

(názov `.conf` súboru môže byť pri niektorých inštaláciách apache rozdielny).

Predvolene sa jedná len o port 80 (po pridaní ďalších portov by bolo treba upraviť aj `ports.conf`)

```
GNU nano 3.2 /etc/apache2/sites-enabled/000-default.conf
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/mojHTTPserver
```

Zmeniť sa mimo zložky pre stránky (`DocumentRoot`) dajú aj zložky pre logovanie, pri ich prípadnej zmene treba zaručiť aby neboli verejne prístupné, pretože obsahujú osobné údaje (IP adresy a časy návštev stránok a pod.)

3. Sprístupnenie webstránok

Predvolene stránky fungujú len na localhoste, po zmene adresára pravdepodobne ani to nie, prístup sa zvykne meniť v hlavnom konfiguračnom súbore `/etc/apache2/apache2.conf`

Podstatné je mať `Directory`, ktorý smeruje na našu zložku a povoliť v ňom prístup všetkým pomocou:

`Require all granted`

Potom som už len vytvoril indexy v zložkách `web1` a `web2` a pridal do nich nejaký text.

```
sudo mkdir /var/www/mojHTTPserver &&
cd /var/www/mojHTTPserver && sudo mkdir web1 web2
echo "<h1>Vitaj na prvom webe</h1>" | sudo tee web1/index.html
echo "<h1>Vitaj na druhom webe</h1>" | sudo tee web2/index.html
```

```
<Directory />
  Options FollowSymLinks
  AllowOverride None
  Require all denied
</Directory>

<Directory /usr/share>
  AllowOverride None
  Require all granted
</Directory>

<Directory /var/www/mojHTTPserver>
  Options Indexes FollowSymLinks
  AllowOverride All
  Require all granted
</Directory>
```

Predvolene sú blokovaný všetci, aj localhost

Pod `/var/www/mojHTTPserver` som povolil prístup všetkým

4. Inštalácia a konfigurácia CMS Wordpress (+ veci k tomu)

Stiahnutie Wordpressu do správnej zložky som spravil v jednom „chainovanom“ príkaze, predtým som získal odkaz na wordpress .zip zo stránky wordpress.org.

```
cd /var/www/mojHTTPserver && sudo mkdir wordpress && cd wordpress
&& sudo wget wordpress.org/latest.zip && sudo unzip latest.zip &&
sudo rm latest.zip
```

Pred konfiguráciou je vhodné zaručiť, že je nainštalovaný aj mysql alebo mariadb server, pretože wordpress potrebuje databázy :) (samozrejmosťou je aj php a podpora php pre apache)

Inštalácia MariaDB (+ postinštalačný skript pre zabezpečenie):

```
sudo apt install mariadb-server && mysql_secure_installation
```

Inštalácia PHP:

```
sudo apt install php libapache2-mod-php php-mysql
```

Vhodné je tiež nastaviť `dir.conf` do poradia, kde je php ako prvá priorita pri prístupu na web ak existuje viacero indexov s rôznymi formátmi:

```
sudo nano /etc/apache2/mods-enabled/dir.conf
```

```
<IfModule mod_dir.c>
  DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
```

Po týchto úpravách sa apache reštartuje `sudo systemctl restart apache2`

Prístup k databázam sa zvykne v praxi riešiť pomocou webového rozhrania PhpMyAdmin, jednoduchšie v tomto bode je použiť len mariaDB shell (príkaz `sudo mariadb`) a vytvoriť databázový účet pre wordpress.

(meno db: wordpressdb, účet: wordpressdbadmin heslo:supersilneheslo)

```
CREATE DATABASE wordpressdb;
```

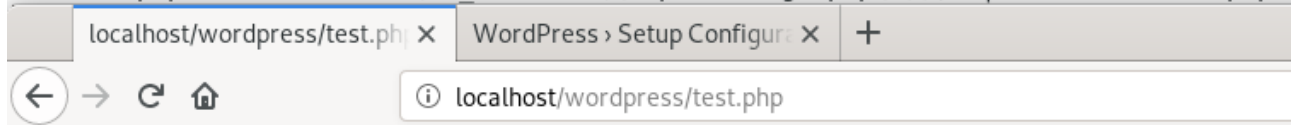
```
GRANT ALL PRIVILEGES ON wordpressdb.* TO wordpressdbadmin@localhost IDENTIFIED BY "supersilneheslo";
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

Kontrola, či PHP dokáže do existujúcich súborov zapisovať zlyháva,

```
echo "<?php echo serialize(is_writable(\"wp-settings.php\"));" | sudo tee test.php
```



b:0;

takže treba podľa bodu 5 prestaviť vlastníctvo v apache envvars alebo dať chown a chmod na obsahy zložiek na základe user/group z envvars

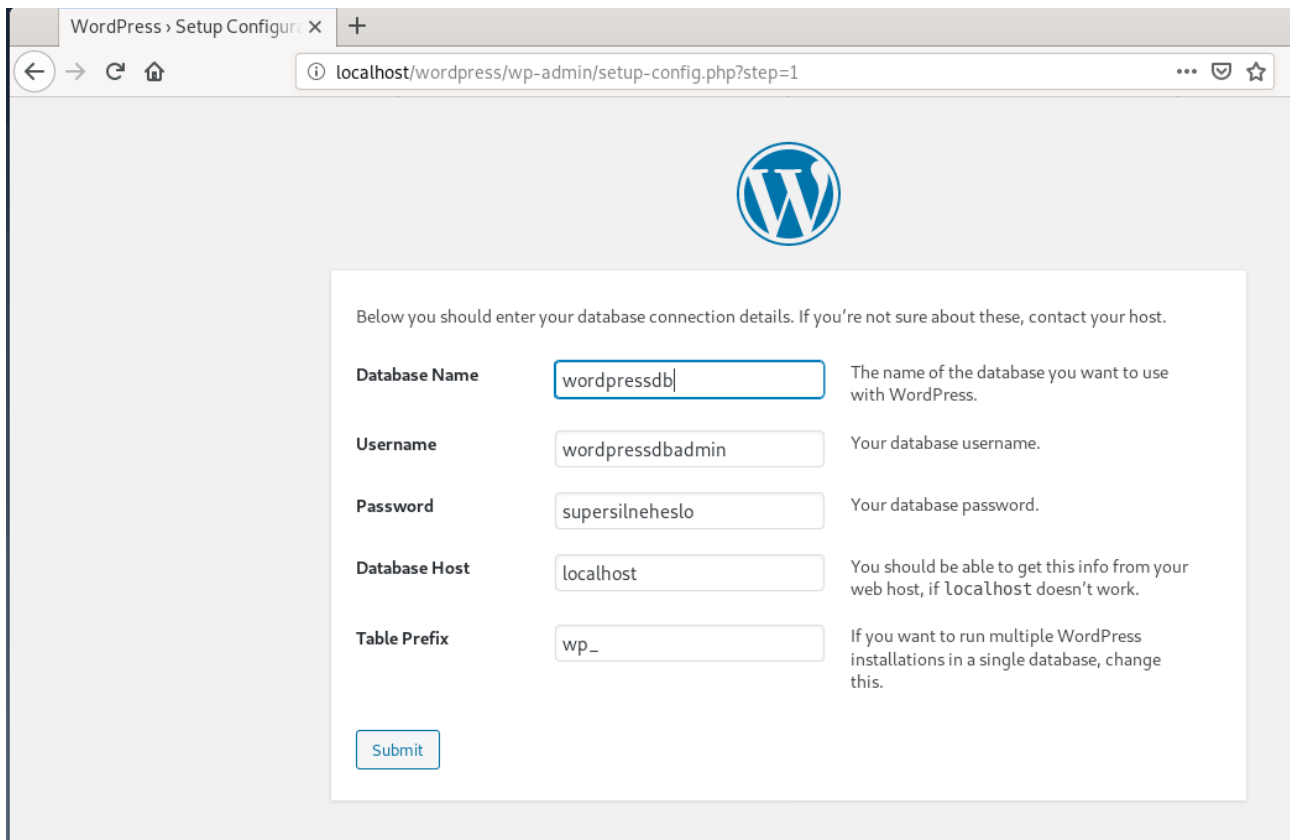
u mňa bol predvolený user/group v envvars www-data, používateľ www-data už podľa `sudo adduser www-data` existuje, takže stačí len vlastníctvo:

```
sudo chown -R www-data:www-data /var/www/mojHTTPserver/*
```

 pre vlastníctvo, kontrola či sú práva 755 na zložkách a 644 na súboroch pomocou `ls -l` a ide sa na vec...

```
ondrej@COVID19:/var/www/mojHTTPserver/wordpress$ ls -l
total 208
-rw-r--r--  1 www-data www-data  420 Nov 30  2017 index.php
-rw-r--r--  1 www-data www-data 19935 Jan  1  2019 license.txt
-rw-r--r--  1 www-data www-data  7368 Sep  2  2019 readme.html
-rw-r--r--  1 www-data www-data  6939 Sep  2  2019 wp-activate.php
drwxr-xr-x  9 www-data www-data 4096 Dec 18 17:16 wp-admin
-rw-r--r--  1 www-data www-data   369 Nov 30  2017 wp-blog-header.php
-rw-r--r--  1 www-data www-data  2283 Jan 20  2019 wp-comments-post.php
-rw-r--r--  1 www-data www-data  2898 Jan  7  2019 wp-config-sample.php
drwxr-xr-x  4 www-data www-data 4096 Dec 18 17:16 wp-content
-rw-r--r--  1 www-data www-data  3955 Oct 10 18:52 wp-cron.php
drwxr-xr-x 20 www-data www-data 12288 Dec 18 17:16 wp-includes
-rw-r--r--  1 www-data www-data  2504 Sep  2  2019 wp-links-opml.php
-rw-r--r--  1 www-data www-data  3326 Sep  2  2019 wp-load.php
-rw-r--r--  1 www-data www-data 47597 Dec  9 08:30 wp-login.php
-rw-r--r--  1 www-data www-data  8483 Sep  2  2019 wp-mail.php
-rw-r--r--  1 www-data www-data 19120 Oct 15 11:37 wp-settings.php
-rw-r--r--  1 www-data www-data 31112 Sep  2  2019 wp-signup.php
-rw-r--r--  1 www-data www-data  4764 Nov 30  2017 wp-trackback.php
-rw-r--r--  1 www-data www-data  3150 Jul  1  2019 xmlrpc.php
```

Konfigurácia Wordpressu započne navigáciou na <http://localhost/wordpress>



The screenshot shows a web browser window with the address bar displaying "localhost/wordpress/wp-admin/setup-config.php?step=1". The page features the WordPress logo at the top center. Below the logo, a text block 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 descriptive note to its right:

- Database Name:** Input field contains "wordpressdb". Note: "The name of the database you want to use with WordPress."
- Username:** Input field contains "wordpressdbadmin". Note: "Your database username."
- Password:** Input field contains "supersilneheslo". Note: "Your database password."
- Database Host:** Input field contains "localhost". Note: "You should be able to get this info from your web host, if localhost doesn't work."
- Table Prefix:** Input field contains "wp_". Note: "If you want to run multiple WordPress installations in a single database, change this."

A "Submit" button is located at the bottom left of the form area.

po všetkej príprave, úspech!

All right, sparky! You've made it through this part of the installation. WordPress can now communicate with your database. If you are ready, time now to...

[Run the installation](#)

5. Nastavenie vlastníkov/práv

Pri tejto inštalácii už apache malo svoj vlastný účet www-data, stačil teda len tretí krok, keby ale bolo na root účte, bolo by nutné editovať envvars (druhý krok) a vytvárať účet.

Predvolené nastavenia apache môžu fungovať pod root účtom a root skupinou, s tým, že pokiaľ zmením práva jednotlivých vlastných súborov tak PHP a iné služby pod apache stratia možnosť do týchto súborov zapisovať (vychádzam z osobného problému s ktorým som sa stretol mimo ZPS).

Ak avšak nezmením práva, najväčší problém to robí pri vzdialenom prístupe (napr. cez FTP), kedy by bol používateľ nútený prihlasovať sa ako root, čo je veľmi zlý nápad z bezpečnostného hľadiska.

Vhodné vlastníctvo je podľa mňa:

1. vytvoriť používateľa „www“ alebo podobne pomenovaného

- pokiaľ mám viacero nezávislých ľudí, ktorí majú na mojom serveri weby, uvažoval by som nad viacerými používateľmi

- tento používateľ bude čiste len na prístup pre službu apache a vzdialený prístup, čiže nepotrebuje home zložku (teda netreba useradd -m ...)

```
sudo useradd www && passwd www
```

2. zmeniť používateľa pod ktorým beží apache editáciou súboru s „environment variables“

```
sudo nano /etc/apache2/envvars
```

```
GNU nano 3.2 /etc/apache2/envvars Modified
# envvars - default environment variables for apache2ctl
# this won't be correct after changing uid
unset HOME
# for supporting multiple apache2 instances
if [ "${APACHE_CONFDIR##/etc/apache2-}" != "${APACHE_CONFDIR}" ] ; then
    SUFFIX="-${APACHE_CONFDIR##/etc/apache2-}"
else
    SUFFIX=
fi
# Since there is no sane way to get the parsed apache2 config in scripts, some
# settings are defined via environment variables and then used in apache2ctl,
# /etc/init.d/apache2, /etc/logrotate.d/apache2, etc.
export APACHE_RUN_USER=www
export APACHE_RUN_GROUP=www
# temporary state file location. This might be changed to /run in Wheezy+1
export APACHE_PID_FILE=/var/run/apache2${SUFFIX}/apache2.pid
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

3. potom treba pozmeniť vlastníctvo k súborom vo webovej zložke (s rekurziou (-R) opatrne)

```
sudo chown www:www -R /var/www/mojHTTPserver
```

Vhodné práva k súborom:

Všeobecne zaužívané ideálne práva pre apache server sú (oktálne písané):

755 pre zložky (t.j. drwxr-xr-x), príkazom: (zapisovať teda môže len vlastník zložky, ostatní iba čítať)
`sudo chmod u+rwx g-w+rx o-w+rx [názov_zložky]`

alebo `sudo chmod 755 [názov_zložky]`

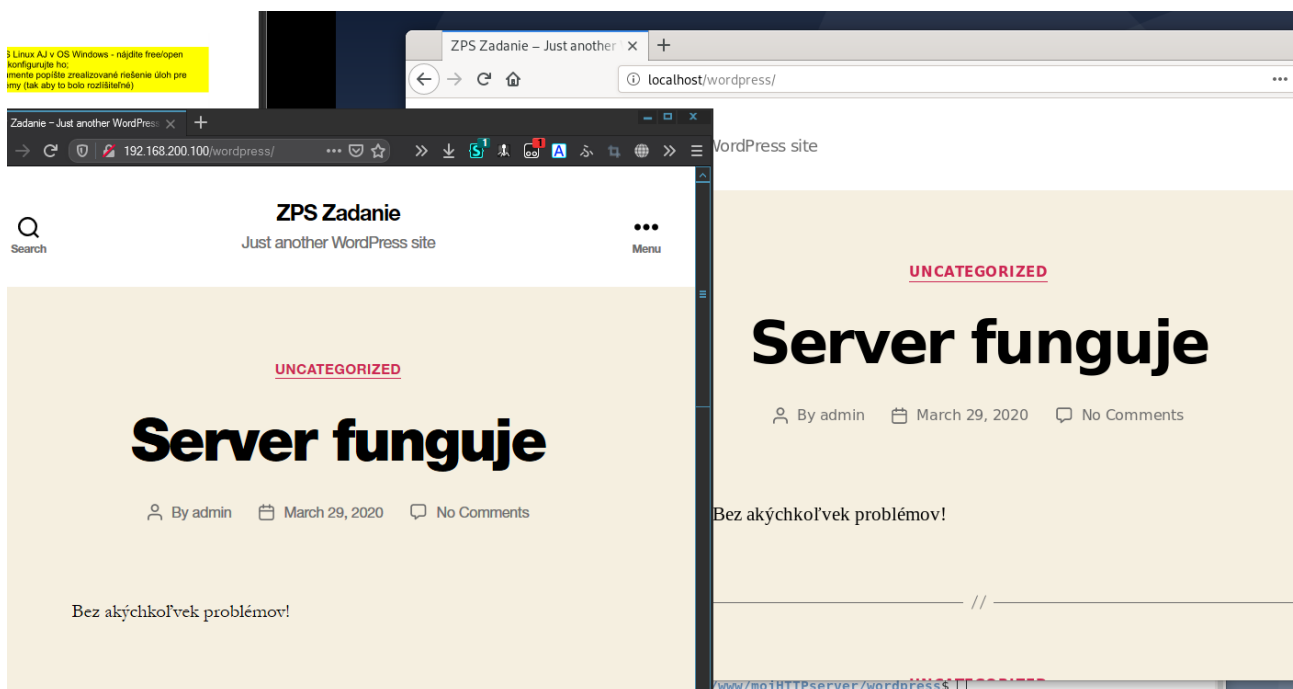
644 pre súbory (t.j. -rw-r--r--), príkazom: (zapisovať teda môže len vlastník zložky, ostatní iba čítať)
`sudo chmod u+rwx g-wx o-wx [názov_súboru]`

alebo `sudo chmod 644 [názov_súboru]`

.htaccess a iné skryté súbory sa zobrazia flagom -a cez `ls -la`, to sa týka aj terajšej/iných vyšších zložiek -> **varovanie z osobnej skúsenosti: určite nepoužívať** `chown -R www:www *` pretože to rekurzívne aplikuje aj na vyššie zložky a rozbije množstvo oprávnení, v minulosti som si takto rozbil celú inštaláciu linuxu :)

```
drwxr-xr-x  3 www  www  4096 Mar 29 13:03 .
drwxrwxr-x 14 root  root  4096 Mar 29 13:01 ..
-rw-r--r--  1 www  www    0 Mar 29 13:02 .htaccess
-rw-r--r--  1 www  www    0 Mar 29 13:02 index.html
-rw-r--r--  1 www  www    0 Mar 29 13:02 is_writable.php
drwxr-xr-x  2 www  www  4096 Mar 29 13:03 nejaka_zlozka
-rw-r--r--  1 www  www    0 Mar 29 13:02 phpinfo.php
-rw-r--r--  1 www  www    0 Mar 29 13:02 podstranka.html
-rw-r--r--  1 www  www    0 Mar 29 13:02 style.css
```

6. Overenie funkčnosti



Chýbajúce CSS pri prístupe mimo localhost sa dá získať pozmenením Site Address a Wordpress Address na IP adresu v nastaveniach Wordpressu.

Realizovanie úloh vo Windowse

Použil som WAMP Server (<http://www.wampserver.com/en/>).

1. Vypnutie auto-spúšťania pri štarte

Predvolene sa nespúšťa pri štarte Windows, keby sa spúšťal tak buď:

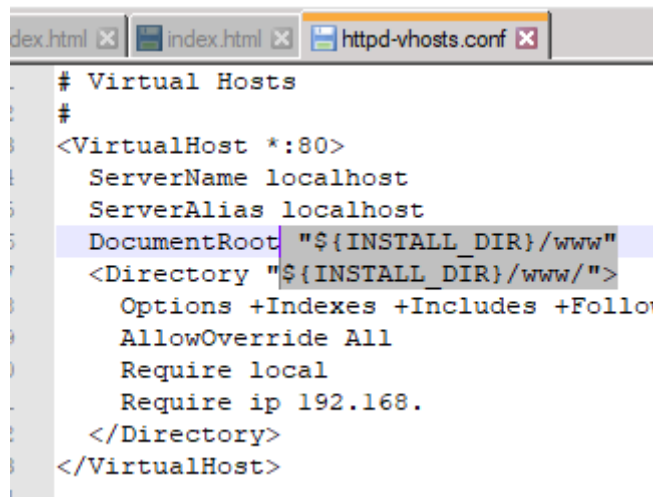
Task Manager > Startup > Disable

alebo vyhodnením z Startup zložky v Start menu, alebo pomocou už inštalovaného SysInternals Autoruns z ostatných ZPS úloh

2. Zmena adresára stránok HTTP servera

httpd-vhosts.conf (cez gui wampu alebo c:\wamp64\bin\apache\apache...\conf\extra)

DocumentRoot a Directory



```
# Virtual Hosts
#
<VirtualHost *:80>
    ServerName localhost
    ServerAlias localhost
    DocumentRoot "${INSTALL_DIR}/www"
    <Directory "${INSTALL_DIR}/www/">
        Options +Indexes +Includes +FollowSymLinks
        AllowOverride All
        Require local
        Require ip 192.168.0.0/24
    </Directory>
</VirtualHost>
```

3. Sprístupnenie webstránok

Tiež httpd-vhosts.conf

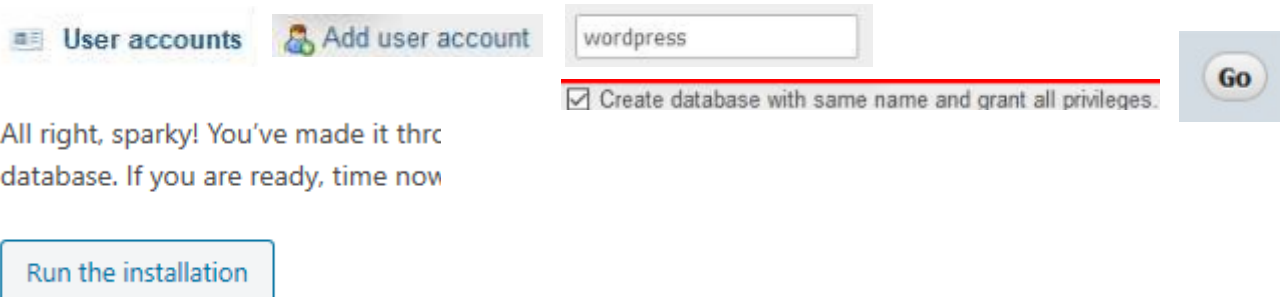
`Require all granted` pre všetkých alebo limitovanie na lokálne adresy ako je na obrázku.

Ak sú stránky na inom porte ako 80, listen porty sa nastavujú v hlavnom httpd.conf

4. Inštalácia a konfigurácia CMS Wordpress

Nový databázový účet a databáza pre wordpress sa vytvorí cez <http://localhost/phpmyadmin>

Ak by sa server zverejňoval zmysel by dávalo vytvoriť heslo pre root, pretože predvolene je v tejto edícii WAMPu bez hesla.



User accounts [Add user account](#)

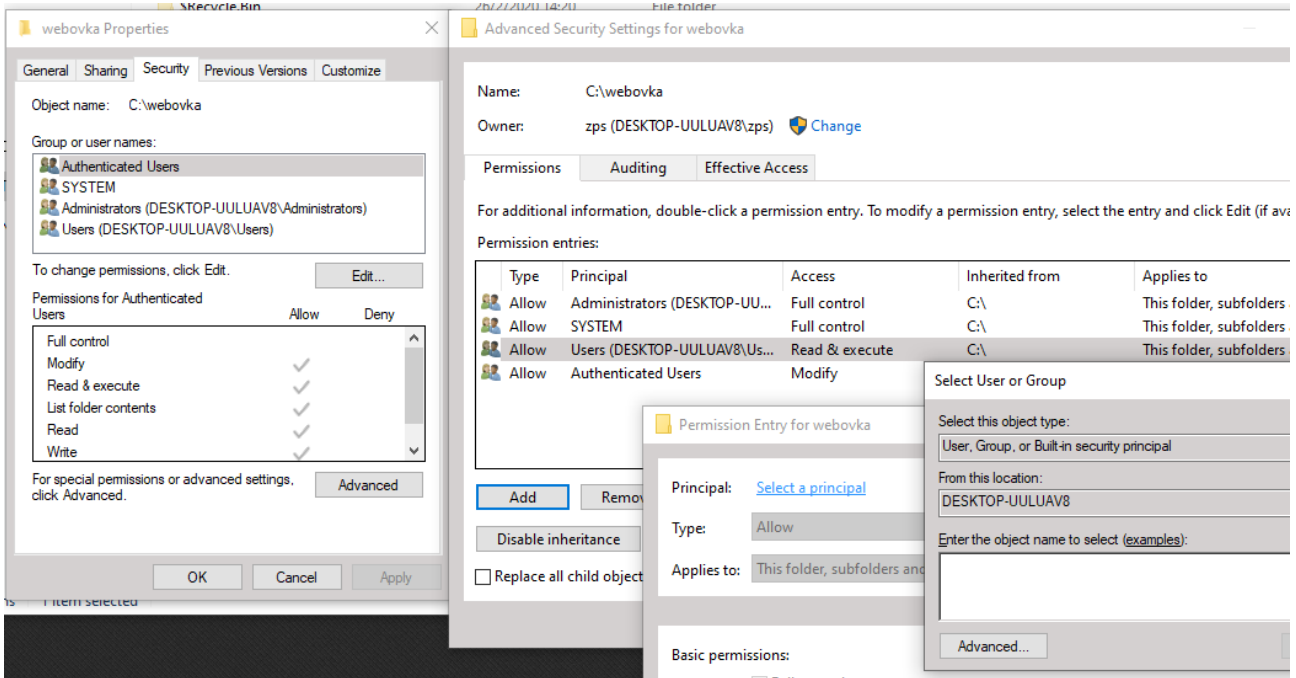
Create database with same name and grant all privileges. [Go](#)

All right, sparky! You've made it thru database. If you are ready, time now

[Run the installation](#)

5. Nastavenie vlastníkov/práv

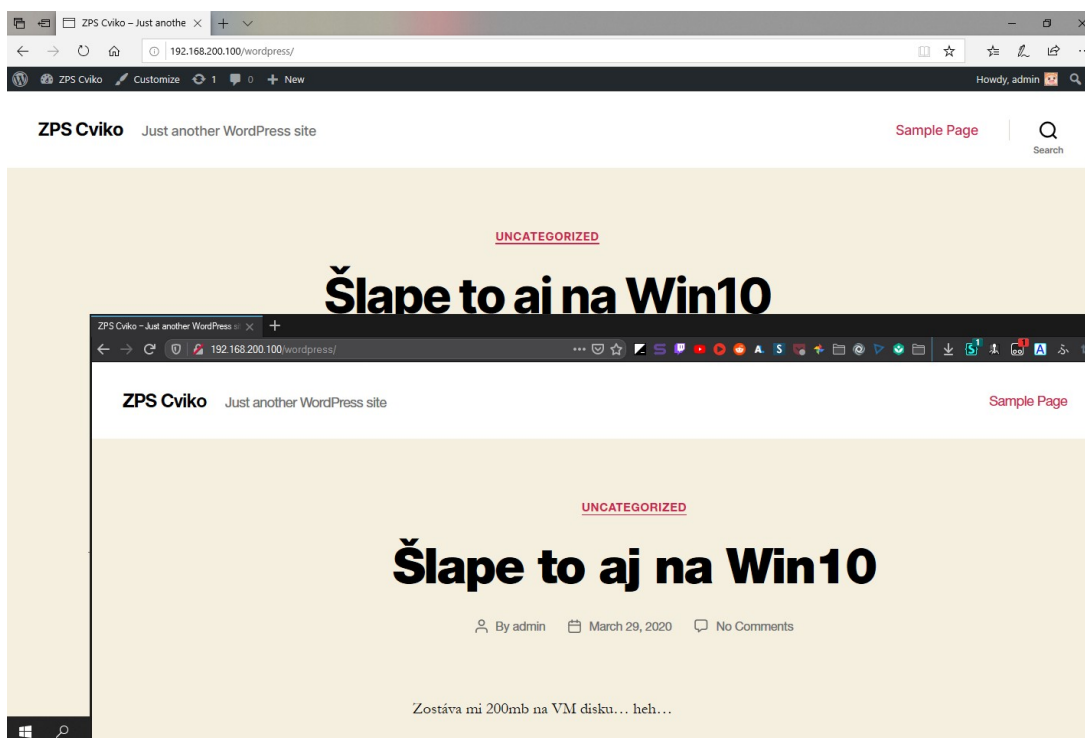
Vo Windowse je z toho väčší bordel ako na Linuxe.



Okrem GUI sa to ešte dá v Powershelli:

```
icacls C:\wamp64\www\mojHTTPserver\* /setowner Ondrej /C  
icacls C:\wamp64\www\mojHTTPserver\* /grant Ondrej:RW /T
```

6. Overenie funkčnosti



Odpoveď na otázku o štandardnom porte

Štandardný port je 80, zmeniť sa dá v konfigurácii, na Linuxe (Apache2) predvolene:

`/etc/sites-enabled/000-default.conf`,

alebo vo Windowse `httpd-vhosts.conf`.

Po zmene/pridaní portu naň treba zaviesť aj počúvanie úpravou

`/etc/apache2/ports.conf` (Listen PORT) a samozrejme reštartovať Apache.

Pokiaľ sa používa iný port ako 80, tak ho treba v prehliadači **uviesť za dvojbodkou v adrese** (napr. localhost:3000, linux.local:82, example.com:8080, 192.168.1.14:3200)