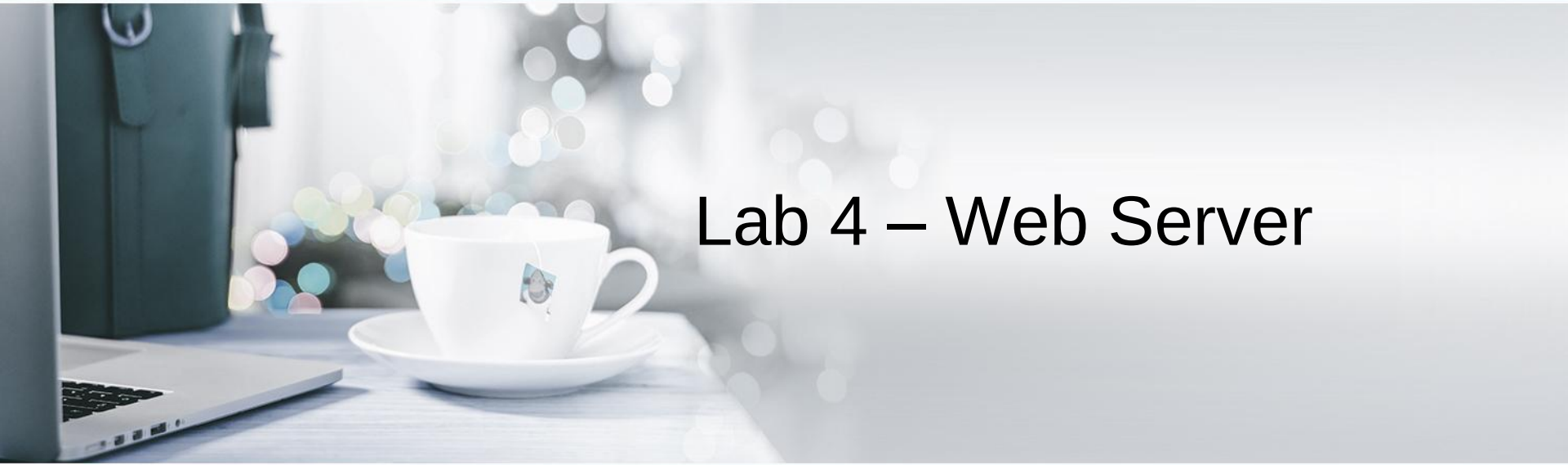
Open Source Software Development

Ung Văn Giàu
Email: giau.ung@eiu.edu.vn



Lab 4 – Web Server

Contents

1. Install web server (LAMP)

- **Introduction**
  - A “LAMP” stack is a group of open source software typically installed together to enable a server to host dynamic websites and web apps
  - LAMP = Linux, Apache, MySQL, PHP
- **Manual**
  - Install Apache (**apache2**)
  - Install MySQL (**mysql-server**)

    Note: Not install mysql_secure_installation
  - Install PHP (**php, libapache2-mod-php, php-mysql**)
  - Config Apache server

    ✓Path: **/etc/apache2/mods-enabled/dir.conf**

    ✓Modify **DirectoryIndex**: index.php index.html index.xhtml index.htm

1. Install web server (LAMP)

- **Manual**
  - Restart the Apache web server (**systemctl restart apache2**)
  - Set up Virtual Hosts:
    - ✓Create the directory for your_domain (**/var/www/your_domain**)
    - ✓Assign ownership ($USER:$USER or www-data:www-data) to the your_domain directory
    - ✓Assign permissions to the your_domain directory (755 or 744)
    - ✓Create an **index.html** file in the your_domain direactory with any content

1. Install web server (LAMP)

- **Manual**
  - Set up Virtual Hosts:

    Create a virtual host file:

    ✓Copy and rename 000-default.conf to your_domain.conf (**/etc/apache2/sites-available/**)

    ✓Modify the file:

      - **ServerName** your_domain
      - **ServerAlias** www.your_domain
      - **DocumentRoot** /var/www/your_domain

    ✓Enable the file with the a2ensite tool (**a2ensite your_domain**)

    ✓Test for configuration errors (**apache2ctl configtest**)

    ✓Reload the configuration (**systemctl reload apache2**)

1. Install web server (LAMP)

- **Manual**
  - Set up domain name:
    - ✓Open hosts file of windows and add the following line:

      Server_IP your_domain
    - ✓Open a browser and browse url: http://your_domain

2. Install web server (LEMP)

- **Introduction**
  - The LEMP software stack is a group of software that can be used to serve dynamic web pages and web applications
  - LEMP = Linux, Nginx (Engine-X), MySQL, PHP
- **Manual**
  - Install the Nginx (**nginx**)
  - If you have the ufw firewall running, allow connections to Nginx
    - ✓Enable this (**ufw allow 'Nginx HTTP'**)
    - ✓verify the change (**ufw status**)

2. Install web server (LEMP)

- **Manual**
  - Install MySQL (**mysql-server**)
  - Install PHP (**php**)
  - Install the php-fpm module along with an additional helper package, php-mysql, which will allow PHP to communicate with your database backend (**php-fpm, php-mysql**)
  - Configure domain (configure Nginx)

    Create a new server block **configuration file** within the /etc/nginx/sites-available/ directory (/etc/nginx/sites-available/**example.com**)

2. Install web server (LEMP)

- **Manual**
  - Configure domain
    - ✓Add the following content

```
server {
        listen 80;
        root /var/www/html;
        index index.php index.html index.htm index.nginx-debian.html;
        server_name example.com;

        location / {
                try_files $uri $uri/ =404;
        }

        location ~ \.php$ {
                include snippets/fastcgi-php.conf;
                fastcgi_pass unix:/var/run/php/php7.2-fpm.sock;
        }

        location ~ /\.ht {
                deny all;
        }
}
```

2. Install web server (LEMP)

- **Manual**
  - Configure domain

    Enable new server block by creating a symbolic link from new server block **configuration file** (in the /etc/nginx/sites-available/ directory) to the /etc/nginx/sites-enabled/ directory
  - Test new configuration file for syntax errors (**nginx –t**)
  - Reload Nginx to make the necessary changes (**systemctl reload nginx**)
  - Create a PHP File to Test Configuration

    ✓Create index.php

    ✓Add the line: <?php phpinfo(); ?>
  - Use existing phpMyadmin:

    Create a soft link form **/usr/share/phpmyadmin/** to **your_domain** directory

3. Install and Secure phpMyAdmin

- **Introduction**
  - Many users need the functionality of a database management system like MySQL, they may not feel comfortable interacting with the system solely from the MySQL prompt
  - phpMyAdmin was created so that users can interact with MySQL through a web interface

- **Prerequisites**
  - Ubuntu Server
  - Completed a LAMP (Linux, Apache, MySQL, and PHP) installation

3. Install and Secure phpMyAdmin

- **Manual (For LAMP)**
  - Install phpMyAdmin (**phpmyadmin, php-mbstring, php-zip, php-gd, php-json, php-curl**)

    Note:

    ✓When the prompt appears, “**apache2**” is highlighted. Hit SPACE, TAB, and then ENTER to select Apache.

    ✓Select **No** when asked whether to use **dbconfig-common** to set up the database
  - Enable the mbstring PHP extension (**phpenmod mbstring**)
  - Restart Apache for changes (**systemctl restart apache2**)
  - Access the web interface by visiting server’s domain name or public IP address followed by /phpmyadmin

3. Install and Secure phpMyAdmin

- **Manual (For LAMP)**
  - Configuring Password Access for the MySQL Root Account
    - ✓Open the MySQL prompt from terminal: sudo mysql
    - ✓Change root password
    - ✓Create a new user account and grant all privileges

3. Install and Secure phpMyAdmin

- **Manual (For LAMP)**
  - **Securing phpMyAdmin Instance (*)**
    - ✓To prevent unauthorized access
    - ✓One of the easiest ways of doing this is to place a gateway in front of the entire application by using Apache's built-in **.htaccess** authentication and authorization functionalities
    - ✓Enable the use of .htaccess file overrides by editing Apache configuration file (**/etc/apache2/conf-available/phpmyadmin.conf**)
    - ✓Add an **AllowOverride All** directive within the **<Directory /usr/share/phpmyadmin>** section of the configuration file
    - ✓Restart Apache

3. Install and Secure phpMyAdmin

- **Manual (For LAMP)**
  - **Securing phpMyAdmin Instance (*)**
    - ✓Create .htaccess within the application directory (**/usr/share/phpmyadmin/.htaccess**)
    - ✓Enter the following information:

      AuthType Basic

      AuthName "Restricted Files"

      AuthUserFile /etc/phpmyadmin/.htpasswd

      Require valid-user
    - ✓Create .htpasswd file and pass it an initial user with the htpasswd utility (**htpasswd -c /etc/phpmyadmin/.htpasswd username**)
    - ✓Select and confirm a password for the user you are creating

3. Install and Secure phpMyAdmin

- **Manual (For LAMP)**
  - **Securing phpMyAdmin Instance (*)**
    - ✓If you want to enter an additional user, you need to do so without the -c flag (**htpasswd /etc/phpmyadmin/.htpasswd additionaluser**)

4. Install vsftpd

- **Introduction**
  - FTP, short for File Transfer Protocol, is a network protocol that was once widely used for moving files between a client and server
  - Optimized for security, performance, and stability, vsftpd offers strong protection against many security problems
- **Manual**
  - Install vsftpd (**vsftpd**)
  - Copy the configuration file so we can start with a blank configuration, saving the original as a backup (**/etc/vsftpd.conf**)

4. Install vsftpd

- **Manual**
  - Open the Firewall
    - ✓check the firewall status to see if it's enabled (**ufw status**)
    - ✓open ports
      - ports 20 and 21 for FTP,
      - port 990 for when we enable TLS,
      - ports 40000-50000 for the range of passive ports we plan to set in the configuration file

```
ufw allow 20/tcp
ufw allow 21/tcp
ufw allow 990/tcp
ufw allow 40000:50000/tcp
ufw status
```

4. Install vsftpd

- **Manual**
  - Create dedicated FTP user
    - ✓Create a new user: **sudo adduser newUser**
    - ✓Create an **ftp** directory in /home/newUser/ directory
    - ✓Set permissions (744 or 764) for the **ftp** directory
    - ✓Create a new file in the **ftp** directory

4. Install vsftpd

- **Manual**
  - Configure FTP Access
    - ✓Open the config file (/etc/vsftpd.conf) to verify that the settings in configuration match those below:

```
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
```

    - ✓Enable the user to upload files by uncommenting the write_enable setting

```
write_enable=YES
```

4. Install vsftpd

- **Manual**
  - Configure FTP Access
    - ✓uncomment the chroot to prevent the FTP-connected user from accessing any files or commands outside the directory tree

```
chroot_local_user=YES
```

    - ✓add a user_sub_token to insert the username in our local_root directory path so our configuration will work for this user and any additional future users. Add these settings anywhere in the file:

```
user_sub_token=$USER
local_root=/home/$USER/ftp
```

4. Install vsftpd

- **Manual**
  - Configure FTP Access

    limit the range of ports that can be used for passive FTP

```
pasv_min_port=40000
pasv_max_port=50000
```

  - Restart the daemon to load the configuration changes (**systemctl restart vsftpd**)
  - Test FTP Access
  - If you see the error: “vsftpd: refusing to run with writable root inside chroot ()”, adding the following line to config file:

    allow_writeable_chroot=YES

  - Secure Transactions (*)

5. Change Apache Port*

Ubuntu 24.04

- **Open Apache Config File**

  sudo nano /etc/apache2/ports.conf

  Change Apache Port Number: Listen 80 → Listen 8080
- **Open Virtual Host Configuration**

  sudo nano /etc/apache2/sites-enabled/myweb.conf

  Change <VirtualHost: *:80> to <VirtualHost: *:8080>
- **Restart Apache Server**

  sudo systemctl restart apache2
- **Test**

  http://192.168.32.134:8080/