VIRTUALIZED LEARNING

Server Configuration and Software Installation Instructions

Prepared By:

Kaylee Cox

George Matthew

Remy Mezebish

Kyle Nguyen

Neehar Peri

Daniel Rowe

Charlie Shin

Kaitlyn Won

Distributed May, 2021

Prepared For:

Dr. Rodolphe Gentili -

Associate Professor of Kinesiology at the University of Maryland, College Park

Dr. Garrett Katz -

Assistant Professor of Electrical Engineering and Computer Science at Syracuse University

This document discusses how to configure VLEARN on a Linux (Debian) operating system. All commands performed below are based upon installation procedures that were performed on the voluble.cs.umd.edu server.

Prerequisites

- Debian version 10 OS installation.
- A fully registered domain name (this is needed for the https/SSL configuration e.g voluble.cs.umd.edu).
- A non-root user with sudo privileges configured on your server.

Apache Installation

Debian's default software repositories include installation binaries for Apache. To install Apache, perform the following:

First, update the local package index to reflect the latest upstream changes:

```
sudo apt update
```

To install Apache, run the following command after capturing the latest upstream changes.

```
sudo apt install apache2
```

Managing Apache Processes

To stop Apache, use the following command.:

```
sudo systemctl stop apache2
```

To start Apache after, use the following command:

```
sudo systemctl start apache2
```

To restart, type the following command:

```
sudo systemctl restart apache2
```

If it is necessary to just make configuration changes, Apache can often reload updated configurations without dropping connections using the following command:

```
sudo systemctl reload apache2
```

To enable the Apache service to start up at boot, type:

```
sudo systemctl enable apache2
```

Apache should now start automatically when the server boots again.

Certbot Installation

Certbot is a tool which can automate certificate issuance and installation for SSL purposes with no downtime. Certbot is not available from the Debian software repositories. To download Cerbot using the apt command, it is necessary to add the backports repository to the sources.list file where apt looks for package sources. To add the backports repository, open (or create) the sources.list file in the sources.list file in your /etc/apt/ directory:

```
sudo vi /etc/apt/sources.list.d/sources.list
```

At the bottom of the file, add the following line:

```
deb http://ftp.debian.org/debian stretch-backports main
```

Save and close the file and update the package lists:

```
sudo apt update
```

Use the following command to install Certbot.

```
sudo apt install python-certbot-apache apache2
```

Apache now has to be configured for Apache by typing the following command and answering the question as listed below:

```
sudo certbot --apache
```

```
Saving debug log to /var/log/letsencrypt/letsencrypt.log Plugins selected:
Authenticator apache, Installer apache Enter email address (used for
urgent renewal and security notices) (Enter  'c' to
cancel):
```

```
[YOUR EMAIL]
```

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You
must agree in order to register with the ACME server at
https://acme-v02.api.letsencrypt.org/directory
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
(A)gree/(C)ancel:
```

```
A
```

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Would you be willing to share your email address with the Electronic
Frontier Foundation, a founding partner of the Let's Encrypt project and
```

```
the non profit organization that develops Certbot? We'd like to send you
email about our work encrypting the web, EFF news, campaigns, and ways to
support digital freedom.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
(Y)es/(N)o:

Y

No names were found in your configuration files. Please enter in your
domain name(s) (comma and/or space separated) (Enter 'c' to cancel):

[YOUR DOMAIN NAME HERE]

Obtaining a new certificate
Performing the following challenges:
http-01 challenge for voluble.cs.umd.edu
Enabled Apache rewrite module
Waiting for verification...
Cleaning up challenges
Created an SSL vhost at /etc/apache2/sites-available/000-default-le
ssl.conf
Enabled Apache socache_shmcb module
Enabled Apache ssl module
Deploying Certificate to VirtualHost /etc/apache2/sites-available/000-
default-le-ssl.conf
Enabling available site: /etc/apache2/sites-available/000-default-le
ssl.conf

Please choose whether or not to redirect HTTP traffic to HTTPS, removing
HTTP access.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1: No redirect - Make no further changes to the webserver configuration.
2: Redirect - Make all requests redirect to secure HTTPS access. Choose
this for
new sites, or if you're confident your site works on HTTPS. You can undo
this
change by editing your web server's configuration.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Select the appropriate number [1-2] then [enter] (press 'c' to cancel):

2

Enabled Apache rewrite module
Redirecting vhost in /etc/apache2/sites-enabled/000-default.conf to ssl
vhost in /etc/apache2/sites-available/000-default-le-ssl.conf
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
Congratulations! You have successfully enabled https://voluble.cs.umd.edu
You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=voluble.cs.umd.edu - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

IMPORTANT NOTES:
- Congratulations! Your certificate and chain have been saved
at:/etc/letsencrypt/live/voluble.cs.umd.edu/fullchain.pem Your key file
has been saved at:
/etc/letsencrypt/live/voluble.cs.umd.edu/privkey.pem
Your cert will expire on 2021-03-08. To obtain a new or tweaked  version
of this certificate in the future, simply run certbot again  with the
"certonly" option. To non-interactively renew *all* of
your certificates, run "certbot renew"
- Your account credentials have been saved in your Certbot  configuration
directory at /etc/letsencrypt. You should make a  secure backup of this
folder now. This configuration directory will  also contain certificates
and private keys obtained by Certbot so  making regular backups of this
folder is ideal.
- If you like Certbot, please consider supporting our work by:
Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate  Donating
to EFF: https://eff.org/donate-le
- We were unable to subscribe you the EFF mailing list because your e-mail
address appears to be invalid. You can try again later by visiting
https://act.eff.org.
```

Verify that a `VirtualHost` block for your domain exists in the `/etc/apache2/sites-available/000-default-le-ssl.conf.` with the `ServerName` directive already set appropriately.

To check, open the virtual host file for your domain::

```
sudo vi /etc/apache2/sites-available/000-default-le-ssl.conf
```

Find the existing `ServerName` line. It should resemble the following:

```
ServerName <host name>;
```

If the host name does not exist, update the `ServerName` directive to point to your domain name. Save the file and verify the syntax of your configuration edits:

```
sudo apache2ctl configtest
```

You should see this output (if not, check for typos in the .conf file):

```
Output

Syntax OK
```

Reload Apache to load the new configuration:

```
sudo systemctl reload apache2
```

Certbot should now be able to find the correct VirtualHost block and update it.

Obtain and configure SSL certificate

The Certbot/Apache plugin reconfigures Apache and reloads the configuration whenever necessary. To use this plugin, type the following:

```
sudo certbot --apache -d <host name>
```

The above command runs `certbot` with the `--apache` plugin, using `-d` to specify the host name for the SSL certificate.

When running the above command for the first time, you will be prompted to enter an email address and agree to the terms of service. After doing so, `certbot` will communicate with the Let's Encrypt server and `Certbot` will ask how you'd like to configure your HTTPS settings:

```
Output

Please choose whether or not to redirect HTTP traffic to HTTPS, removing
HTTP access.

-------------------------------------------------------------------------------
-------------------------------------------------------------------------------

1: No redirect - Make no further changes to the webserver configuration.
2: Redirect - Make all requests redirect to secure HTTPS access. Choose
this for

new sites, or if you're confident your site works on HTTPS. You can undo
this

change by editing your web server's configuration.

-------------------------------------------------------------------------------
-------------------------------------------------------------------------------

Select the appropriate number [1-2] then [enter] (press 'c' to cancel):
```

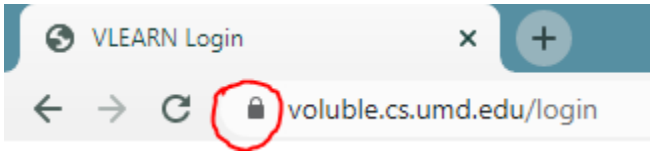
Select your choice then hit ENTER. The configuration will be updated, and Apache will reload to pick up the new settings. `certbot` will wrap up with a message telling you the process was successful and where your certificates are stored:

```
Output

IMPORTANT NOTES:
- Congratulations! Your certificate and chain have been saved at:
/etc/letsencrypt/live/<host name>/fullchain.pem
Your key file has been saved at:
/etc/letsencrypt/live/<dns host name>/privkey.pem
```

```
Your cert will expire on 2021-03-04. To obtain a new or tweaked  version
of this certificate in the future, simply run certbot again  with the
"certonly" option. To non-interactively renew *all* of  your certificates,
run "certbot renew"
- Your account credentials have been saved in your Certbot  configuration
directory at /etc/letsencrypt. You should make a  secure backup of this
folder now. This configuration directory will  also contain certificates
and private keys obtained by Certbot so  making regular backups of this
folder is ideal.
- If you like Certbot, please consider supporting our work by:
Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate  Donating
to EFF: https://eff.org/donate-le
```

At this point, the certificates should be downloaded, installed, and loaded. If you reload your website using https://<host name and notice your browser's security indicator. It should indicate that the site is properly secured, usually with a lock icon..



VLEARN Software Installation

The following apache proxy modules need to be enabled for VLEARN. Run the following commands to enable the reverse proxy for the VLEARN software:

```
sudo a2enmod proxy

sudo a2enmod proxy_http

sudo a2enmod proxy_balancer

sudo a2enmod lbmethod_byrequests

sudo a2enmod proxy_wstunnel
```

****IMPORTANT**:** After running the above commands, the

/etc/apache2/sites-enabled/000-default-le-ssl.conf

file should be updated to include the following for node.js integration (see the following link for more guidance: https://httpd.apache.org/docs/current/mod/mod_proxy.html)

```
ProxyPreserveHost On

ProxyPass / http://127.0.0.1:3000/

RewriteEngine on
```

```
RewriteCond %{HTTP:Upgrade} websocket [NC]

RewriteCond %{HTTP:Connection} upgrade [NC]

RewriteRule ^/?(.*) "ws://127.0.0.1:3000/$1" [P,L]

ProxyPassReverse / http://127.0.0.1:3000/
```

Restart the server using the following command:

```
sudo systemctl restart apache2
```

Create a ~/home/vlearn_webapp directory using:

```
mkdir home

cd home

mkdir vlearn_webapp
```

It is necessary to install node.js modules for VLEARN. To install node.js, run the following command from the /home/vlearn_webapp directory:

```
sudo apt-get install nodejs npm
```

You can check that both Node and npm have been installed by running the command:

```
node -v

npm -v
```

Next is to install subversion on the server. To install subversion, run the following command:

```
sudo apt-get install subversion
```

The VLEARN software can be retrieved from the following SVN repository:

https://vis.cs.umd.edu/svn/projects/vlearn2/Webapp/

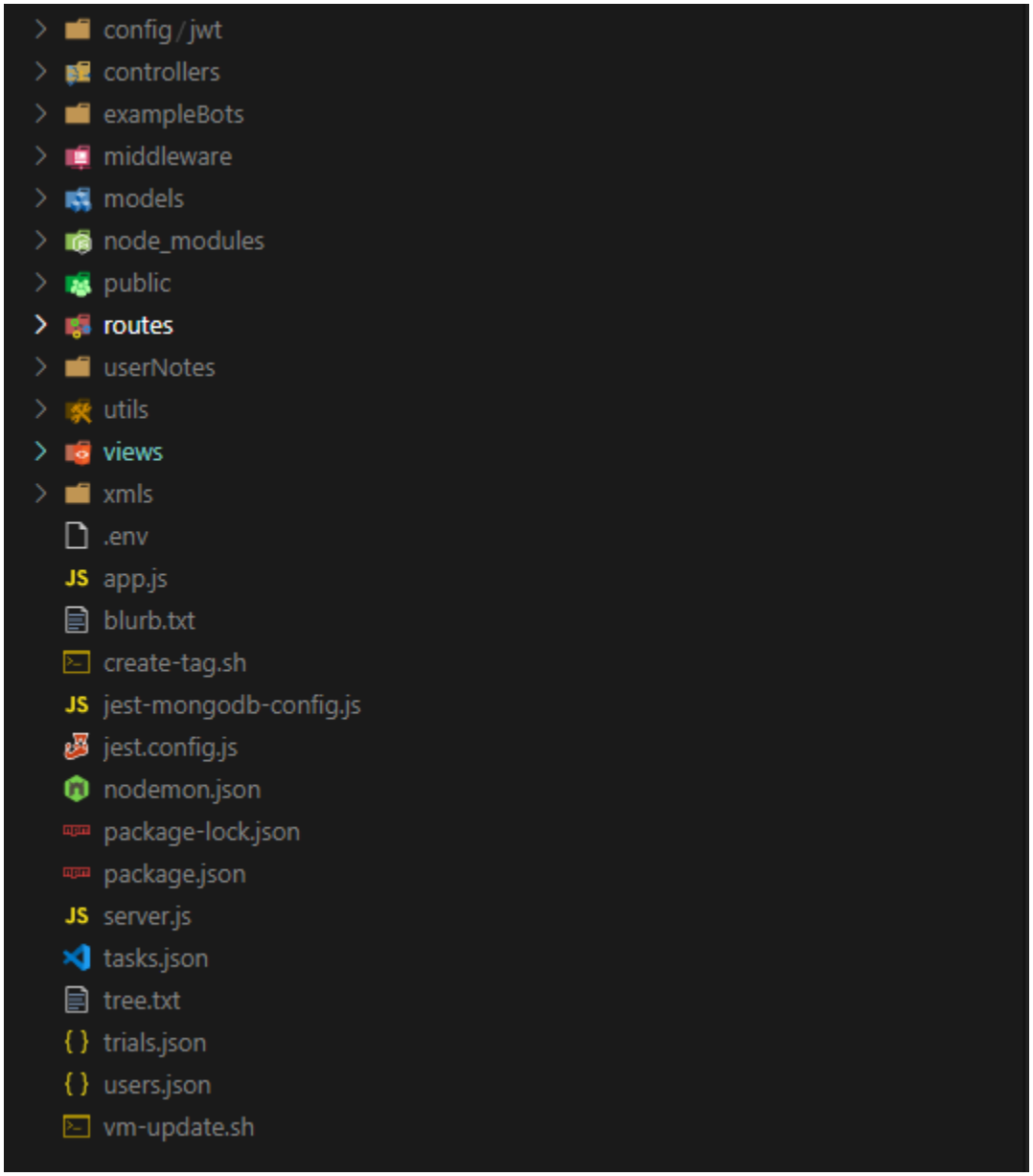
For more information on SVN and our branching, please see the SVN Documentation Document

We will be checking out the code from the most recent tag, from the /home/ directory, run the following command

```
svn checkout
```

`https://vis.cs.umd.edu/svn/projects/vlearn2/Webapp/tags/2.1.2/vlearn_webapp/`

VLEARN code should now be installed with the following directory hierarchy relative to the /home/vlearn_webapp/ directory on the deployment server:



Once the VLEARN code as been installed as per the directory structure above, you can install all dependencies required by VLEARN by running the following command from the home/vlearn_webapp/ directory:

```
npm install
```

VLEARN uses JSON Web Tokens for authentication tracking. The secret key used to sign all JSON Web Tokens, as well as the port number for the webapp should be stored in a file with the name ".env" in the home/vlearn_webapp/ directory. The file should have the following contents:

```
PORT=3000

JWT_KEY='[YOUR_SECRET_KEY_HERE]'

MONGO_USER='[MONGODB USER]'

MONGO_PASS='[MONGODB PASSWORD]'
```

Because the ".env" file is within the repository already, it is up to you if you would like to generate a new secret key or keep the current key.

After deploying the web artifacts as per the directory structure above, you must run the command:

```
node app.js
```

The VLEARN software should then be able to be accessed at the following link: https://<hostame>/

The current VLEARN deployment can be accessed here:

https://voluble.cs.umd.edu/

Running VLEARN via Node Process Manager (Recommended)

Although the VLEARN webapp can be run with the above command, the webapp will not remain up if the user running the command logs out. It is therefore recommended to run through the Node Process Manager (PM2). PM2 was NOT installed along with the other dependencies above, so you will need to install it via the following command:

```
sudo npm install -g pm2
```

Check that PM2 has been installed using:

```
pm2 -v
```

After installing PM2, you can run the following command in the home/vlearn_webapp/ directory to start VLEARN as a background process.

```
sudo pm2 start "npm start" --name "vlearn"
```

If you need to restart VLEARN at any point you can run the command:

```
sudo pm2 restart vlearn
```

If you need to stop VLEARN, use the command:

```
sudo pm2 stop vlearn
```

If you would like to check the status and log files for the VLEARN webapp, you can also use the commands below.

```
sudo pm2 status
```

```
sudo pm2 logs
```