

Cloud Server Project – YOURLS URL Shortener

Ethan Brownsdon (35574325)

ICT171: Introduction to Server Environments and Architectures

June 9th 2025

Site available at <https://ict171.com>

GitHub Repository available at <https://github.com/aeonr6/ict171-assignment-2>

IP: 3.27.183.110

Domain: <https://ict171.com>

Setup Guide

This project was setup using Amazon AWS on a Linux Machine, and will assume you are doing the same. The steps can also be repeated in a Virtual Machine.

This project will assume you have created a Ubuntu EC2 instance that allows SSH, HTTP, and HTTPS traffic.

Passwords for the service have been provided in the .pdf in submitted to LMS

Linking IP and DNS:

For this project, I purchased an IP directly from Amazon AWS Route 53. When done this way, AWS automatically creates a hosted zone. This document will assume you do not have a hosted zone already created.

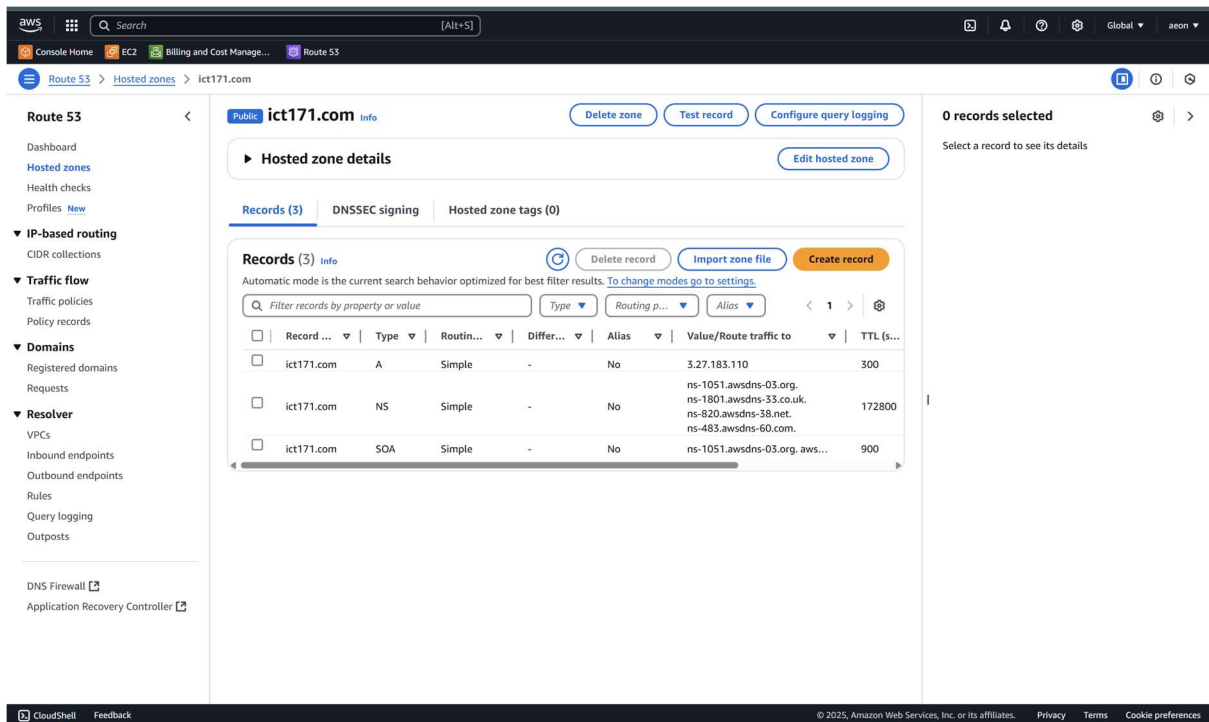
Start by navigating to Route 53 on the AWS website. Then, on the left hand side of the screen click “Hosted Zones”

In the Hosted Zones section, click “Create Hosted Zone” and enter your Domain Name into the box provided. Ensure it is set to public and click “Create Hosted Zone”

Then, click create record. This will likely create 2 records: one labelled NS and one labelled SOA. If you did not buy your Domain from AWS Route 53, edit the NS record so that it contains the name servers provided to you. You do not have to do this is you purchased a record from AWS

Click “Create Record” and select “A – IPV4”. Then, in the Value section, enter your EC2 instances public facing IP Address. Click “Create Record”.

It will then look something like this:



Hosting and Server

Begin by connecting to the cloud server via SSH

```
sudo ssh yourkeypair.pem ubuntu@ec2ipaddress
```

Ensure all apps are updated and upgraded using the command

```
sudo apt update && sudo apt upgrade -y
```

The -y automatically confirms the code

Then, download apache2

```
sudo apt install apache2 -y
```

Install php and its dependencies

```
Sudo apt install php libapache2-mod-php php-mysql php-cli php-curl php-mbstring  
php-xml unzip curl -y
```

This will automatically download and unzip the required dependencies for php

Then install mysql using the command

```
sudo apt install mysql-server -y
```

You will then allow mysql to install a necessary server deployment

```
sudo mysql_secure_installation
```

This will then ask you a few setup questions that allow mysql to deploy safely

You will then configure the mysql settings by logging into the mysql server and creating a database

```
sudo mysql -u root -p
```

```
CREATE DATABASE yourls;
```

```
CREATE USER 'yourlsuser'@'localhost' IDENTIFIED BY 'Password';
```

```
GRANT ALL PRIVILEGES ON yourls.* TO 'yourlsuser'@'localhost';
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

```
ubuntu@ip-172-31-9-240:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.42-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE yourls;
Query OK, 1 row affected (0.01 sec)

mysql> CREATE USER 'yourlsuser'@'localhost' IDENTIFIED BY XXXXXXXXXX;
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT ALL PRIVILEGES ON yourls.* TO 'yourlsuser'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> EXIT;
Bye
```

Then, exit the virtual machine and install yourls directly from the source to the computer you are working on. This is available at <https://github.com/YOURLS/YOURLS/releases> .

After installing yourls, transfer the file over to the EC2 virtual machine and unzip it

```
scp -i yourkeypair.pem YOURLS-1.10.1.zip ubuntu@ec2ipaddress:/home/ubuntu/
```

```
ssh -i yourkeypair.pem ubuntu@ec2ipaddress
```

```
unzip YOURLS-1.10.1.zip
```

Then, move the YOURLS file to /var/www/html. You will also need to move the .htaccess file as it is a hidden file. On some distributions, the .htaccess file will not exist. The code “2>/dev/null” hides the error in this case.

```
sudo mv YOURLS-1.10.1/* /var/www/html
```

```
sudo mv YOURLS-1.10.1/.htaccess /var/www/html 2>/dev/null
```

Then, Apache must be given access to the files so that it can read and write them. Restart Apache after this to ensure settings have loaded correctly

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

```
sudo systemctl restart apache2
```

You then need to allow rewrite rules for Apache and edit config settings to allow changes.

```
sudo a2enmod rewrite
```

```
sudo nano /etc/apache2/apache2.conf
```

```
<Directory /var/www/>
```

```
Options Indexes FollowSymLinks
```

AllowOverride All

Require all granted

</Directory>

sudo systemctl restart apache2

Then, you must create and edit the YOURLS config file.

cd /var/www/html/user

sudo cp config-sample.php config.php

sudo nano config.php

Ensure the above file contains the following on the required lines.

```
define( 'YOURLS_DB_USER', 'yourlsuser' );  
  
define( 'YOURLS_DB_PASS', 'Password' );  
  
define( 'YOURLS_DB_NAME', 'yourls' );  
  
define( 'YOURLS_DB_HOST', 'localhost' );  
  
define( 'YOURLS_SITE', 'https://ict171.com' );  
  
define( 'YOURLS_PRIVATE', true );  
  
define( 'YOURLS_COOKIEKEY', 'A long random string' );  
  
$yourls_user_passwords = array(  
  
    'admin' => 'Password'  
  
);
```

This will look like this:

```

GNU nano 7.2 config.php
define('YOURLS_DB_USER', 'yourlsuser' );
// MySQL database password #
define('YOURLS_DB_PASS', 'ICT171Assignment2' );
// The name of the database for YOURLS
// Use lower case letters [a-z], digits [0-9] and underscores [_] only #
define('YOURLS_DB_NAME', 'yourls' );
// MySQL hostname
// If using a non standard port, specify it like 'hostname:port', e.g. 'localhost:9999' or '127.0.0.1:666' #
define('YOURLS_DB_HOST', 'localhost' );
// MySQL tables prefix
// YOURLS will create tables using this prefix (eg 'yourls_url', 'yourls_options', ...)
// Use lower case letters [a-z], digits [0-9] and underscores [_] only #
define('YOURLS_DB_PREFIX', 'yourls_');
//
// Site options
//
// YOURLS installation URL
// All lowercase, no trailing slash at the end.
// If you define it to "http://sho.rt", don't use "http://www.sho.rt" in your browser (and vice-versa)
// To use an IDN domain (eg http://h4m6.com), write its ascii form here (eg http://xn--hh-bjab.com) #
define('YOURLS_SITE', 'https://ict171.com' );
// YOURLS language
// Change this setting to use a translation file for your language, instead of the default English.
// That translation file (a .mo file) must be installed in the user/language directory.
// See http://yourls.org/translatiions for more information #
define('YOURLS_LANG', '' );
// Allow multiple short URLs for a same long URL
// Set to true to have only one pair of shortURL/longURL (default YOURLS behavior)
// Set to false to allow multiple short URLs pointing to the same long URL (bit.ly behavior) #
define('YOURLS_UNIQUE_URLS', true );
// Private means the Admin area will be protected with login/pass as defined below.
// Set to false for public usage (eg on a restricted intranet or for test setups)
// See http://yourls.org/privatepublic for more details if you're unsure #
define('YOURLS_PRIVATE', true );
// A random secret hash used to encrypt cookies. You don't have to remember it, make it long and complicated
// Hint: copy from http://yourls.org/cookie #
define('YOURLS_COOKIEKEY', 'MLWS-n-J)(HEZc8cCHBzz6kY-OATb807trXSjR2' );
// Username(s) and password(s) allowed to access the site. Passwords either in plain text or as encrypted hashes
// YOURLS will auto encrypt plain text passwords in this file

```

The cookie key required can be found at <https://api.yourls.org/services/cookiekey/1.0/> as mentioned in the config document

```

GNU nano 7.2 yourls-script.sh
#!/bin/bash

# Config
API_URL="https://ict171.com/yourls-api.php"
USERNAME="admin"
PASSWORD="35574325Admin"

# URL to shorten
LONG_URL="https://aws.amazon.com"
CUSTOM_KEY="aws"
Title="AWS"

# Perform API request
curl "${API_URL}?username=${USERNAME}&password=${PASSWORD}&action=shorturl&format=simple&url=${LONG_URL}&keyword=${CUSTOM_KEY}&title=${TITLE}"

```

The .htaccess file must then be edited and configured. In this, I have included an optional step to redirect <https://ict171.com> to <https://ict171.com/admin>

```
sudo nano /var/www/html/.htaccess
```

```
<IfModule mod_rewrite.c>
```

```
RewriteEngine On
```

```
#OPTIONAL STEP
```

RewriteBase /

RewriteCond %{REQUEST_URI} ^/

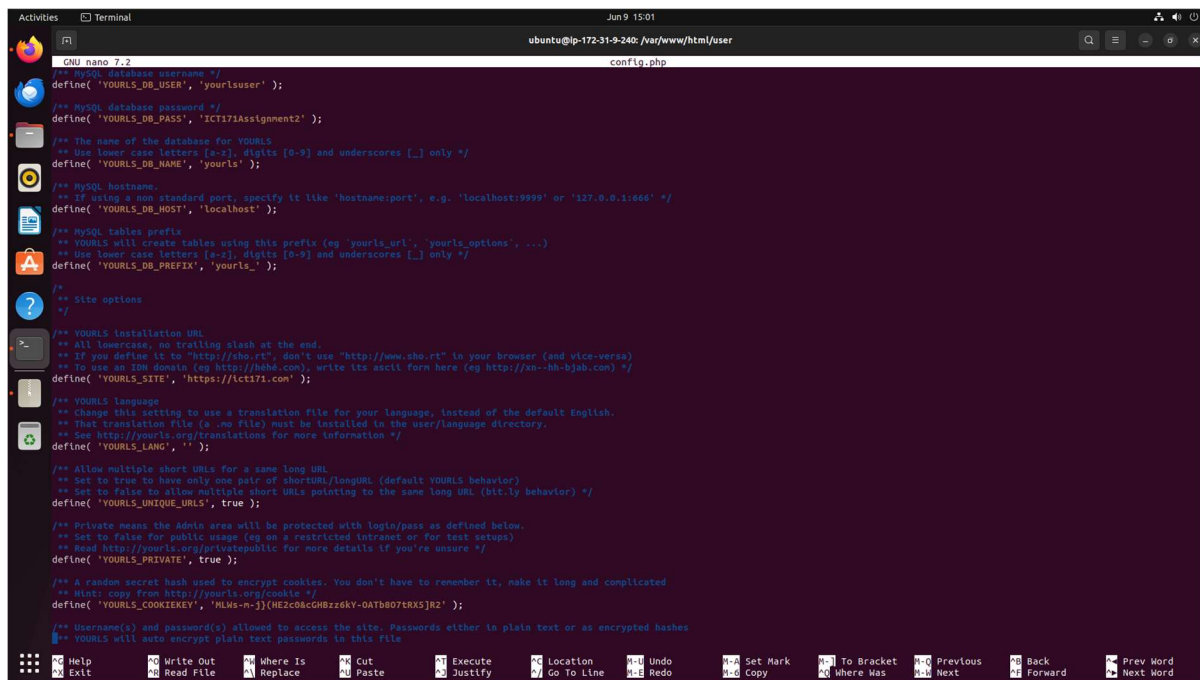
RewriteRule ^\$ /admin [R=302,L]

#END OPTIONAL STEP

RewriteCond %{REQUEST_FILENAME} !-f

RewriteCond %{REQUEST_FILENAME} !-d

RewriteRule ^.*\$ yourls-loader.php [L]



```
GNU nano 2.2 config.php
/*-- MySQL database username */
define( 'YOURLS_DB_USER', 'yoursuser' );

/*-- MySQL database password */
define( 'YOURLS_DB_PASS', 'lct171Assignment2' );

/*-- The name of the database for YOURLS
** Use lower case letters [a-z], digits [0-9] and underscores [_] only */
define( 'YOURLS_DB_NAME', 'yours' );

/*-- MySQL hostname.
** If using a non standard port, specify it like 'hostname:port', e.g. 'localhost:9999' or '127.0.0.1:666' */
define( 'YOURLS_DB_HOST', 'localhost' );

/*-- MySQL tables prefix
** YOURLS will create tables using this prefix (eg 'yours_url', 'yours_options', ...)
** Use lower case letters [a-z], digits [0-9] and underscores [_] only */
define( 'YOURLS_DB_PREFIX', 'yours_' );

/*-- Site options
*/

/*-- YOURLS installation URL
** All lowercase, no trailing slash at the end.
** If you define it to "http://sho.rt", don't use "http://www.sho.rt" in your browser (and vice-versa)
** To use an IDN domain (eg http://hh4.com), write its ascii form here (eg http://xn--hh-bj9b.com) */
define( 'YOURLS_SITE', 'https://lct171.com' );

/*-- YOURLS language
** Change this setting to use a translation file for your language, instead of the default English.
** That translation file (a .mo file) must be installed in the user/language directory.
** See http://yours.org/translations for more information */
define( 'YOURLS_LANG', '' );

/*-- Allow multiple short URLs for a same long URL
** Set to true to have only one pair of shortURL/longURL (default YOURLS behavior)
** Set to false to allow multiple short URLs pointing to the same long URL (bit.ly behavior) */
define( 'YOURLS_UNIQUE_URLS', true );

/*-- Private means the Admin area will be protected with login/pass as defined below.
** Set to false for public usage (eg on a restricted intranet or for test setups)
** Read http://yours.org/privatepublic for more details if you're unsure */
define( 'YOURLS_PRIVATE', true );

/*-- A random secret hash used to encrypt cookies. You don't have to remember it, make it long and complicated
** Hint: copy from http://yours.org/cookie */
define( 'YOURLS_COOKIEKEY', 'MLMs-n-3}{HE2c08cGHBz6ky-QATB807ERX5]R2' );

/*-- User(s) and password(s) allowed to access the site. Passwords either in plain text or as encrypted hashes
** YOURLS will auto encrypt plain text passwords in this file
```

</IfModule>


```
GNU nano 7.2
# BEGIN YOURLS
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_URI} ^/$
RewriteRule ^$ /admin [R=302,L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^.*$ /yourls-loader.php [L]
</IfModule>
# END YOURLS
```

Security

Now that the server has been created, a certificate must be applied to the website. This will be done using certbot, which first must be installed.

```
sudo apt install certbot python3-certbot-apache -y
```

Then, you use certbot to create and apply a certificate

```
sudo certbot - -apache2
```

It will then ask you a number of questions, including the address you would like to link the certificate to.

After this, your server will be running, and all functions of YOURLS will be available to anyone with the administrator password.

Script

This service can now be used with any web browser. However, on systems without access to a web browser, a bash script can be used to create new links. This is done using the following script:

```
#!/bin/bash
```

```
API_URL="https://ict171.com/yourls-api.php"
```

```
USERNAME="admin"
```

```
PASSWORD="Password"
```

```
LONG_URL="https://example.com"
```

```
CUSTOM_KEY="testlink"
```

```
TITLE="My Test Link"
```

```
curl
```

```
"${API_URL}?username=${USERNAME}&password=${PASSWORD}&action=shorturl  
&format=simple&url=${LONG_URL}&keyword=${CUSTOM_KEY}&title=${TITLE}"
```

The above script uses the YOURLS API and its markers to shorten a URL.

The LONG_URL is the marker for the website URL you would like to redirect to

The CUSTOM_KEY is the marker for the shortened URL. For example, if you made the custom key testlink, the website would redirect to ict171.com/testlink

The TITLE is the marker for what the link is called on the admin dashboard of YOURLS.

Once this script has been edited to fit your desired link and ran, logging in to the YOURLS admin page will display this link on the table of created links.

e.g.:

```
GNU nano 7.2                                yourls-script.sh
#!/bin/bash

# Config
API_URL="https://ict171.com/yourls-api.php"
USERNAME=
PASSWORD=

# URL to shorten
LONG_URL="https://aws.amazon.com"
CUSTOM_KEY="aws"
Title="AWS"

# Perform API request
curl "${API_URL}?username=${USERNAME}&password=${PASSWORD}&action=shorturl&format=simple&url=${LONG_URL}&keyword=${CUSTOM_KEY}&title=${TITLE}"
```

Links

```
GNU nano 7.2
# BEGIN YOURLS
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_URI} ^/$
RewriteRule ^$ /admin [R=302,L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^.*$ /yourls-loader.php [L]
</IfModule>
# END YOURLS
```

The original download of Y

OURLS is available at <https://yourls.org/docs/guide/install> . Some of the steps used in the creation of this project are available at this link, as it is the YOURLS installation guide.

The website is live at <https://ict171.com> .

This will redirect to <https://ict171.com/admin> .

The GitHub repository is available at <https://github.com/aeonr6/ict171-assignment-2> .

An example working link is <https://ict171.com/r3> which leads to the Yamaha YZF-R3 information page