

Script for installing/prepping nginx

Prepping nginx

I use [nginx](#) for all my "in-house" applications, but also for other things.

This script is targeted towards Debian-based systems, as I usually rely on Ubuntu/Debian for server stuff.

This is a mirror of a GitHub Gist: <https://gist.github.com/Decicus/2f09db5d30f4f24e39de3792bba75b72>

The GitHub Gist link should be considered the "master copy". This wiki page is mainly for explaining the scripts and copies of said script/configs may be outdated.

A Git mirror can be found on my [personal Gitea instance](#) of said GitHub Gist, in case it ever goes down.

This prep script does the following:

- Downloads and installs `nginx, openssl, curl`
- Installs [acme.sh](#) for Let's Encrypt certificates
- Creates a directory for SSL certificates (`/srv/ssl`) and sets the correct permissions to prevent other users from snooping.
- Downloads a file for [loading the `acme.sh` environment and Cloudflare API \(DNS\) variables](#), mainly to replace the automatic `acme.sh` loading.
 - Cloudflare API information is used for [utilizing the DNS API](#).
 - Generally I have to manually remove the environment loader that `acme.sh` automatically adds to `bashrc/zshrc`, so it's not "double". Haven't added anything to handle that for me yet.
- Downloads an [nginx config for Let's Encrypt's webroot authentication](#).
 - Points (via alias) all requests to `/.well-known/acme-challenge` to a single directory `/var/www/html/.well-known/acme-challenge`
- Downloads an [nginx config for "good" TLS/SSL settings](#)
 - Settings are based on the "intermediate NGINX settings" from [Mozilla's configuration generator](#)
 - An alternative is to check [cipherli.st](#).
- Downloads an [nginx config for PHP \(7.4\) FPM](#)
 - This isn't used in the base "virtualhost" config by default, as it's commented.
- Downloads a [script to generate dhparams via OpenSSL](#)
 - [At this point I forget why this is good to have, so read this link I found on the first Google result](#)
- Links to a base virtualhost config in the terminal :)

Files below were last updated April 30th, 2021. Check the GitHub Gist linked further up for updated script/configs.

setup.sh

The actual prep script, all links refer to the GitHub gist's "raw" URLs.

```
#!/bin/bash
# Make sure the 'essentials' are installed
sudo apt install -y nginx-full openssl curl

# Get acme.sh for issuing certificates
curl -L https://get.acme.sh/ | sudo bash

GIST="https://gist.github.com/Decicus/2f09db5d30f4f24e39de3792bba75b72/raw"
NGINX="/etc/nginx"
SSL_BASE="/srv/ssl"

# Create preferred base directory for storing SSL certificates
mkdir -p $SSL_BASE
chown -R root:root $SSL_BASE
chmod -R 600 $SSL_BASE

# Now the fun starts

# I have bash scripts that interact with acme.sh
# But I use zsh as the main shell
# Therefore I need a shared "environment file" that loads acme.sh
# And related environment variables
curl -L "$GIST/.acmeenv" > "$HOME/.acmeenv"

# Get the alias config for Let's Encrypt challenges:
curl -L "$GIST/letsencrypt.conf" > "$NGINX/letsencrypt.conf"

# Get the base SSL configuration
curl -L "$GIST/ssl_params.conf" > "$NGINX/ssl_params.conf"

# Get the PHP 7.4 FPM configuration (not enabled by default)
# You also need to install PHP before enabling it.
curl -L "$GIST/phpfpm.conf" > "$NGINX/phpfpm.conf"

# Get the dhparams file generation script, and execute.
curl -L "$GIST/generate-dhparams.sh" | sudo bash

# Add to ZSH/Bash config files
echo '. "$HOME/.acmeenv"' >> "$HOME/.zshrc";
echo '. "$HOME/.acmeenv"' >> "$HOME/.bashrc";
```

echo "Base setup done. Open this link for a base nginx site configuration: [\\$GIST/000-default.conf](#)"

000-default.conf

Base virtualhost config

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    include letsencrypt.conf;

    server_name _;
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name _;
    root /var/www/html;

    ssl_certificate /srv/ssl/default/fullchain.pem;
    ssl_certificate_key /srv/ssl/default/key.pem;

    server_tokens off;

    include ssl_params.conf;
    include letsencrypt.conf;

    add_header X-Frame-Options "SAMEORIGIN";
    add_header X-XSS-Protection "1; mode=block";
    add_header X-Content-Type-Options "nosniff";
    add_header Strict-Transport-Security "max-age=63072000; includeSubDomains; preload";

    index index.nginx-debian.html index.html index.htm;

    charset utf-8;

    location / {
```

```

    try_files $uri $uri/ =404;
}

location /.well-known {
    auth_basic "off";
}

location = /favicon.ico { access_log off; log_not_found off; }
location = /robots.txt { access_log off; log_not_found off; }

# Uncomment for PHP support (check /etc/nginx/phpfpm.conf), assumes PHP 7.2 FPM is installed.
# include phpfpm.conf;

access_log /var/log/nginx/default-access.log combined;
error_log /var/log/nginx/default-error.log error;

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

```

generate-dhparams.sh

```

#!/bin/bash
sudo touch /etc/nginx/dhparams.pem
sudo chmod 700 /etc/nginx/dhparams.pem
# 4096 would also work here:
sudo openssl dhparam -out /etc/nginx/dhparams.pem 2048

```

letsencrypt.conf

```

location /.well-known/acme-challenge {
    alias /var/www/html/.well-known/acme-challenge;
}

```

phpfpm.conf

```

location ~ \.php$ {
    try_files $uri =404;
    fastcgi_split_path_info ^(.+\.php)(/.+)$;
    fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
}

```

```
fastcgi_index index.php;

include fastcgi_params;

fastcgi_param SCRIPT_FILENAME $document_root/$fastcgi_script_name;

}
```

ssl_params.conf

```
ssl_protocols TLSv1.2 TLSv1.3;
ssl_ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384;
ssl_prefer_server_ciphers off;
ssl_dhparam /etc/nginx/dhparams.pem;
ssl_session_cache shared:SSL:10m;
ssl_session_timeout 1d;
ssl_session_tickets off;
```

.acmeenv

```
. "$HOME/.acme.sh/acme.sh.env"

export CF_Account_ID=""
export CF_Token=""
```

Revision #6
Created 13 January 2019 08:13:51 by Alex Thomassen
Updated 30 April 2021 08:28:25 by Alex Thomassen