NGINX + SSL

Version:	1.0.0
Created by:	cloudimg

Table of Contents

1.) Overview

This document is provided as a user guide for the NGINX + SSL product offering on the AWS Marketplace. Please reach out to support@cloudimg.co.uk if any issues are encountered following this user guide for the chosen product offering.

2.) Access & Security

Please update the security group of the target instance to allow the below ports and protocols for access and connectivity.

Protocol	Type	Port	Description
SSH	TCP	22	SSH connectivity
HTTP	HTTP	80	NGINX Front End
HTTPS	HTTS	443	NGINX Front End (SSL enabled)

3.) System Requirements

The minimum system requirements for the chosen product offering can be found below.

Minimum CPU	Minimum RAM	Required Disk Space
1	1 GB	20 GB

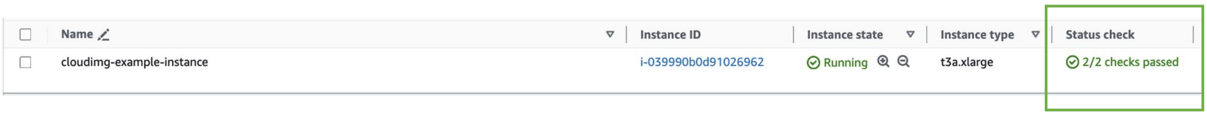
4.) Connecting to the Instance

Once launched in the Amazon EC2 Service, please connect to the instance via an SSH client using the **ec2-user** with the key pair associated at launch. Once connected as the **ec2-user** user, you will be able to sudo to the **root** user by issuing the below command.

Switch to the root user.

```
sudo su -
```

NOTE: Please allow the EC2 Instance to reach 2/2 successful status checks to ensure you will be able to connect successfully with the ec2-key pair assigned at launch. Upon attempting to SSH to early you may receive errors such as below, this is expected with an early SSH connection. Allow the EC2 instance to reach 2/2 status checks and you will be able successfully connect with the ec2-key pair assigned at launch as the ec2-user.



□	Name	Instance ID	Instance state	Instance type	Status check
□	cloudimg-example-instance	i-039990b0d91026962	Running	t3a.xlarge	2/2 checks passed

Example errors you may receive with an early SSH connection.

```
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).

ec2-user@your-instance-ip's password:
```

5.) On Startup

An OS package update script has been configured to run on boot to ensure the image is fully up to date at first use. You can disable this feature by removing the script from /stage/scripts/ and deleting the entry in crontab for the root user.

Disable the OS update script from running on reboot.

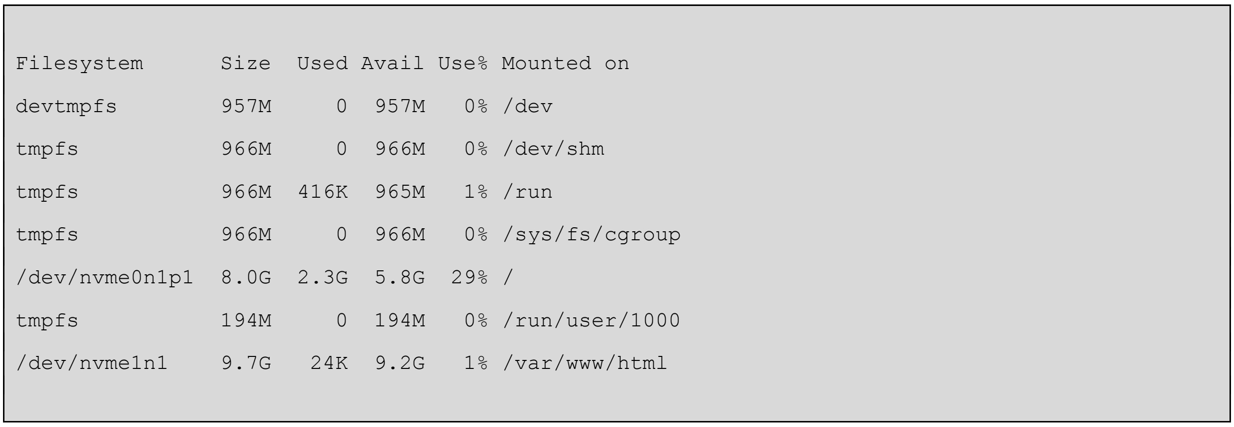
```
rm -f /stage/scripts/initial_boot_update.sh

crontab -e

#DELETE THE BELOW LINE. SAVE AND EXIT THE FILE.
@reboot /stage/scripts/initial_boot_update.sh
```

6.) Filesystem Configuration

Please see below for a screenshot of the server disk configuration and specific mount point mappings for software locations.



```
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        957M     0  957M   0% /dev
tmpfs           966M     0  966M   0% /dev/shm
tmpfs           966M  416K  965M   1% /run
tmpfs           966M     0  966M   0% /sys/fs/cgroup
/dev/nvme0n1p1  8.0G  2.3G  5.8G  29% /
tmpfs           194M     0  194M   0% /run/user/1000
/dev/nvme1n1    9.7G   24K  9.2G   1% /var/www/html
```

Mount Point	Description
/boot	Operating System Kernel files
/var/www/html	NGINX Web Server Root

7.) Server Components

Please see below for a list of installed server components and their respective installation paths. The below versions are subject to change on initial boot based on the initial_boot_update.sh script finding new versions of the software in the systems package repositories.

Component	Software Home
NGINX	/etc/nginx

8.) Scripts and Log Files

The below table provides a breakdown of any scripts & log files created to enhance the useability of the chosen offering.

Script/Log	Path	Description
Initial_boot_update.sh	/stage/scripts	Update the Operating System with the latest updates available.
Initial_boot_update.log	/stage/scripts	Provides output for initial_boot_update.sh

9.) Using System Components

Instructions can be found below for using each component of the server build mentioned in section 7 of this user guide document.

NGINX

The NGINX Server has been configured to start on boot, please use the below commands to start, stop and check the status of the service.

```
#Check the NGINX Server is running
systemctl status nginx

#Stop the NGINX Server
systemctl stop nginx

#Start the NGINX Server
systemctl start nginx
```

Configure SSL via Certbot

Before issuing a certificate, Let’s Encrypt validates ownership of your domain. The Let’s Encrypt client, running on your host, creates a temporary file (a token) with the required information in it. The Let’s Encrypt validation server then makes an HTTP request to retrieve the file and validates the token, which verifies that the DNS record for your domain resolves to the server running the Let’s Encrypt client.

Prerequisites

- Own or control the registered domain name for the certificate. If you don’t have a registered domain name, you can use a domain name registrar, such as GoDaddy or dnsexit.

- Create a DNS record that associates your domain name and your server’s public IP address.

An example nginx configuration file has been created under the below directory /etc/nginx/conf.d/www.example.com.conf

Please edit the above files contents and name to suit your needs.

Run – Check the contents for the www.example.com/conf file.

```
cat /etc/nginx/conf.d/www.example.com.conf
```

Edit the values in **RED** to suit your needs for a simple configuration.

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;
    root /var/www/html;
    server_name example.com www.example.com;
}
```

Run – Please run the below commands to verify the configuration file syntax is correct.

```
systemctl restart nginx

nginx -t && nginx -s reload
```

Run – Issue the below command to generate the required SSL certificated. Changing the values in **RED** to match that of your own domain.

```
certbot --nginx -d example.com -d www.example.com
```

EXPECTED OUTPUT – Change values in **RED** to suit your needs.

```
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator nginx, Installer nginx
Enter email address (used for urgent renewal and security notices)
 (Enter 'c' to cancel): example-sysadmin@cloudimg.co.uk

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.3-September-21-2022.pdf. You must
agree in order to register with the ACME server. Do you agree?
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
(Y)es/(N)o: Y

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Would you be willing, once your first certificate is successfully issued, 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
Account registered.

Congratulations! Your certificate and chain have been saved at:
/etc/letsencrypt/live/example.com/fullchain.pem
Your key file has been saved at:
/etc/letsencrypt/live/example.com//privkey.pem
Your cert will expire on 2017-12-12.
```

If you encounter any issues with issuing the required SSL certificate, please reach out to support@cloudimg.co.uk for support.

Automating Certificate Renewal

It is advised to configure a cron job to daily check if the certificate requires renewal within the next 30 days, if found to be, the latest certificate will be downloaded and applied. Below is an example to configure the required cron job to run daily at noon.

Run – Open crontab as the **root** user.

```
crontab -e

0 12 * * * /usr/bin/certbot renew --quiet
```