XVII. Monit

Monit is a utility for managing and monitoring processes, programs, files, directories and filesystems on a Unix system.

Monit conducts automatic maintenance and repair and can execute meaningful causal actions in error situations.

Monit service can recovers a crashed or dead service automatically. A webservice like Apache or nginx may crash/stop under various circumstances. General approach is that we setup a monitoring service like Nagios to track the status of a service.

In case if service is dead a notification would be sent by such monitoring tool and the next action would be to start the crashed service by logging into the system. This process can be automated by monit.

Every process in Linux will have a PID, PID of processes are stored in a text file.
For example, by default nginx service pid file location is /var/run/nginx.pid. If the process dies, the PID file gets deleted.

```
root@imrandevops:/var/run# pwd
/var/run

root@imrandevops:/var/run# service nginx start

root@imrandevops:/var/run# ls -ltr nginx.pid
rwrr1 root root 5 May 28 00:19 nginx.pid

root@imrandevops:/var/run# cat nginx.pid
1856

root@imrandevops:/var/run# service nginx stop

root@imrandevops:/var/run# ls ltr
nginx.pid
ls: cannot access nginx.pid: No such file or directory
```

Monit can monitor PID file of a program, if the PID file is absent it will try to start that particular service.

1. Setup monit and configure it to monitor nginx service.

```
root@imrandevops:~# apt update
root@imrandevops:~# apt install nginx
root@imrandevops:~# service nginx start
root@imrandevops:~# apt install monit
```

Monit comes with a basic web interface through which all of the processes can be set up. We will setup first the web interface for monit.

Monit web UI runs on port 2812. Open monitrc and uncomment the section that begins with set httpd port 2812.

```
root@imrandevops:~# vi /etc/monit/monitrc
set httpd port 2812
use address 192.168.1.8 # only accept connection from localhost
allow 0.0.0.0/0.0.0.0 # allow localhost to connect to the server and
allow admin:monit # require user 'admin' with password 'monit'
```

Reload monit configuration

```
root@imrandevops:~# monit reload
```

Now you should be able to access the web interface of monit by going to “http://monitIP:2812”

2. Monitoring setup for nginx service.

Now you can setup services that you want to monitor in "/etc/monit/monitrc file".
Open monitrc and add below mentioned content.

```
check process nginx with pidfile /var/run/nginx.pid
start program = "/etc/init.d/nginx start"
stop program = "/etc/init.d/nginx stop"
```

Reload monit

```
root@imrandevops:~# monit reload
```

We will stop nginx service to test monit.

```
root@imrandevops:~# service nginx stop

root@imrandevops:~# service nginx status
* nginx is not running

root@imrandevops:~# tailf /var/log/monit.log
[UTC May 28 00:41:30] info : monit HTTP server started
[UTC May 28 00:41:30] info : 'imrandevops'
Monit reloaded
[UTC May 28 00:44:30] error : 'nginx' process is not running
[UTC May 28 00:44:30] info : 'nginx' trying to restart
[UTC May 28 00:44:30] info : 'nginx' start: /etc/init.d/nginx

root@imrandevops:~# service nginx status
* nginx is running
Monit will check the pid file for every 120 seconds, this can be changed by
editing "set daemon" value in monitrc file.
Please refer to monit documentation to learn monit in detail
```

Check monit in detail.
https://www.mmonit.com/monit/documentation/monit.html#NAME

XVIII. SSL Self-signed Certificate for AWS Elastic Load Balancer

DOCUMENTED BY SANTOSH BABU. K.

1. What is SSL?

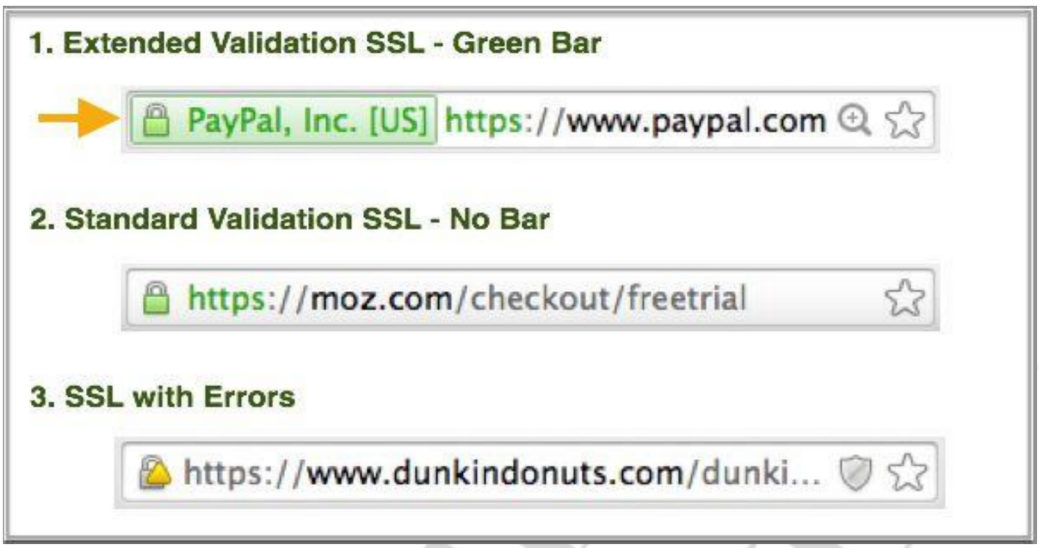
SSL (Secure Sockets Layer) is the standard security technology for establishing an encrypted link between a web server and a browser. This link ensures that all data passed between the web server and browsers remain private and integral. SSL is an industry standard and is used by millions of websites in the protection of their online transactions with their customers.

Typically, an SSL Certificate will contain your domain name, your company name, your address, your city, your state and your country. It will also contain the expiration date of the Certificate and details of the Certification Authority responsible for the issuance of the Certificate. When a browser connects to a secure site it will retrieve the site's SSL Certificate and check that it has not expired, it has been issued by a Certification Authority the browser trusts, and that it is being used by the website for which it has been issued. If it fails on any one of these checks the browser will display a warning to the end user letting them know that the site is not secured by SSL.

SSL Certificate having two types of certificate:

1.Standard Validation SSL – (Standard level of validation. Typically cost between $0-$100.Lower price, quicker issuance time)
Standard SSL certificate meaning that it's a valid certificate issued by a trusted Certificate Authority, but there was no extended validation of the owner of the domain/site. This could mean that the certificate claims to be from Foo Inc. but the CA did not check that the person/entity applying for the certificate was indeed Foo Inc. when they issued the certificate.

2.Extended Validation SSL – (Offers the highest level of validation and often costs between $100-500.5-10 days issuance time)
Extended Validation SSL Certificate. This type of certificate does extended validation in that the CA verifies the physical address and other details of Foo Inc. before issuing the certificate. In the end this just serves to give more assurance to the end user that the owner of the site/domain is indeed the company Foo Inc.



2. List of Popular SSL Certificate Authorities(CA):

1.Comodo
2.Symantec
3.GoDaddy
4.GlobalSign
5.Digicert…..etc,

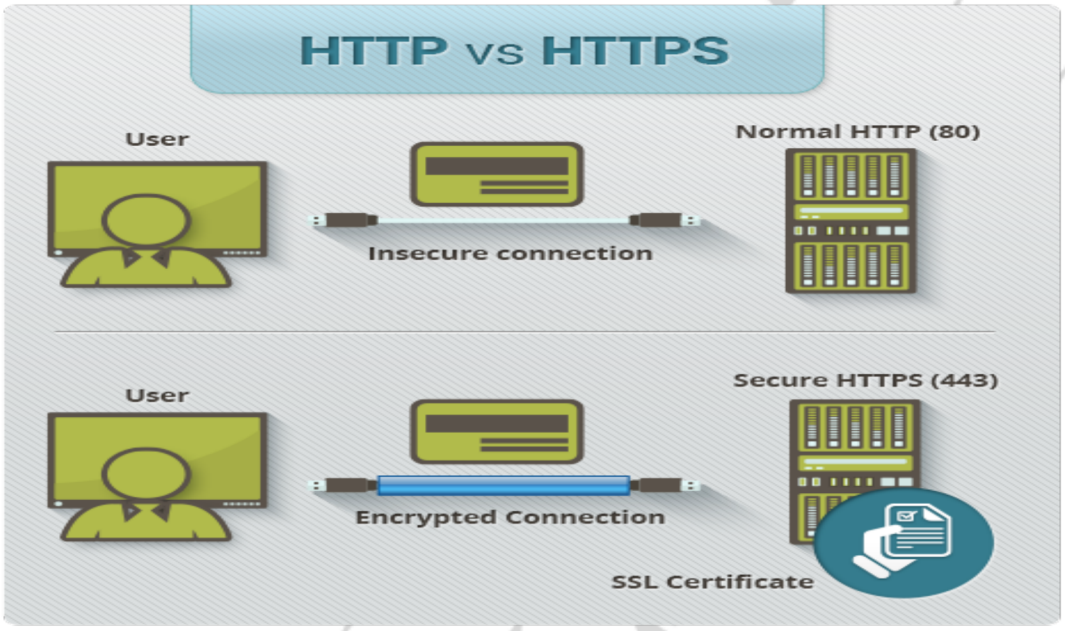
3. What is Self-Signed Certificate?

In cryptography and computer security, a **self-signed certificate** is an identity certificate that is signed by the same entity whose identity it certifies. This term has nothing to do with the identity of the person or organization that actually performed the signing procedure. In technical terms, a self-signed certificate is one signed with its own private key.

In typical public key infrastructure (PKI) arrangements, a digital signature from a certificate authority (CA) attests that a particular public key certificate is valid (i.e., contains correct information). Each CA has one or more root keys; and the certificates associated with those public keys are "trust anchors" that use a special type of self-signed certificates. Establishing trust of the CA root certificate is dependent upon procedures beyond checking its digital signature.

Self-signed certificates can be created for free using a wide variety of tools including OpenSSL, Java's keytool, Adobe Reader, and Apple's Keychain.

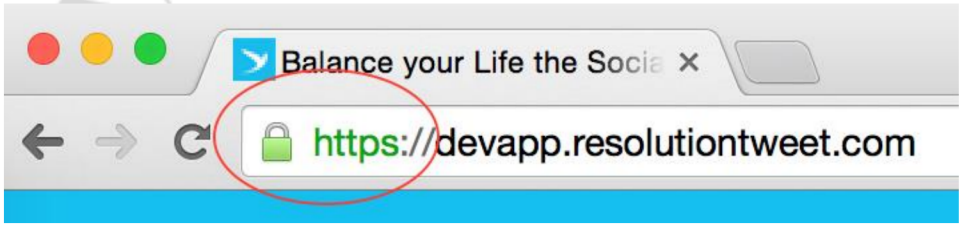
4. How it Works?



5.What is HTTPS Protocol?

When you request a HTTPS connection to a webpage, the website will initially send its SSL certificate to your browser. This certificate contains the public key needed to begin the secure session. Based on this initial exchange, your browser and the website then initiate the 'SSL handshake'. The SSL handshake involves the generation of shared secrets to establish a uniquely secure connection between yourself and the website.

When a trusted SSL Digital Certificate is used during a HTTPS connection, users will see a padlock icon in the browser address bar. When an Extended Validation Certificate is installed on a web site, the address bar will turn green.



Prerequisite's:

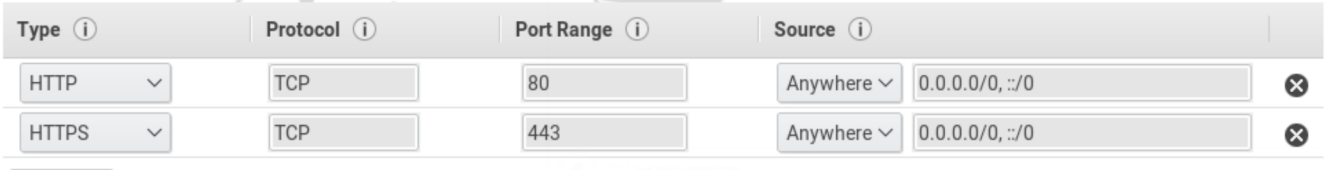
1.EC2 Instance (Install a web server, such as Apache or Internet Information Services (IIS), on each instance, enter its Public IP into the address field of an Internet-connected web browser, and verify that the browser displays the default page of the server.)

Security Group:

Web-server:

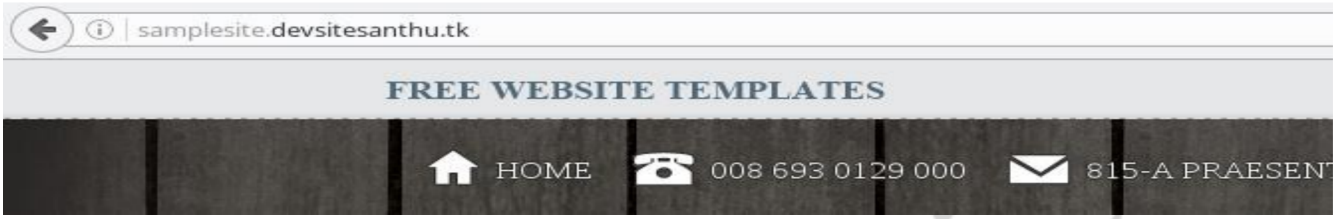
2.ElasticLoadBalancer**(ClassicLoadBalancer)**-(Attach the EC2 instance webserver to ELB,After at least one of your EC2 instances is in service, you can test your load balancer. Copy the string from **DNS name** (for example, my-load-balancer-1234567890.us-west-2.elb.amazonaws.com) and paste it into the address field of an Internet-connected web browser. If your load balancer is working, you see the default page of your server.

Security Group:



Web-server:

3.Register a Domain and set the DNS record for ELB(Using freenom.com to register domain)



6. How to Generate Self-Signed Certificate using OpenSSL? (in Centos)

Note: keys are must be in pem format because AWS understands. pem(Privacy Enhanced Mail) keys

What is OpenSSL?

OpenSSL is an open source software package that implements SSL and TLS protocols for conducting secure communication over digital networks.

Step1: Login into Website Server instance and Install mod_ssl package

- sudo -i
- yum install mod_ssl

```
[centos@ip-172-31-27-189 ~]$ sudo -i
[root@ip-172-31-27-189 ~]# yum install mod_ssl
```

Step2: Generate a 2048 bit RSA Private key

```
[root@ip-172-31-27-189 ~]# openssl genrsa 2048 > privatekey.pem
Generating RSA private key, 2048 bit long modulus
.................................+++
.........................+++
e is 65537 (0x10001)
```

genrsa: You can generate a public-private key pair with the genrsa
2048: key length in bits

Step3: Generate Self-Signed Certificate using req flag
Command:
openssl req -new -x509 -nodes -sha256 -days 365 -key privatekey.pem -outform PEM -out mycertificate.pem

```
[root@ip-172-31-27-189 ~]# openssl req -new -x509 -nodes -sha256 -days 365 -key privatekey.pem -outform PEM -out
 mycertificate.pem
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:IN
State or Province Name (full name) []:ANDHRAPRADESH
Locality Name (eg, city) [Default City]:AMARAVATHI
Organization Name (eg, company) [Default Company Ltd]:VISUALPATH
Organizational Unit Name (eg, section) []:IT
Common Name (eg, your name or your server's hostname) []:*.devsitesanthu.tk
Email Address []:sample.email@gmail.com
```

req:certificate request and certificate generating utility.
new: new request
x509: this option outputs a self signed certificate instead of a certificate request. This is typically used to generate a test certificate or a self signed root CA
nodes: if this option is specified then if a private key is created it will not be encrypted.
sha256:SHA256 hash algorithm does not intervene in the encryption / authentication process but tools (browsers, email clients, servers...) must be able to read / decipher this kind of hash during the connection / authentication process.
days: when the **-x509** option is being used this specifies the number of days to certify the certificate for. The default is 30 days.
key: private key format - default PEM
outform: output format (DER or PEM)
out: This specifies the output filename to write to or standard output by default.
These generate keys are saved in current working Directory

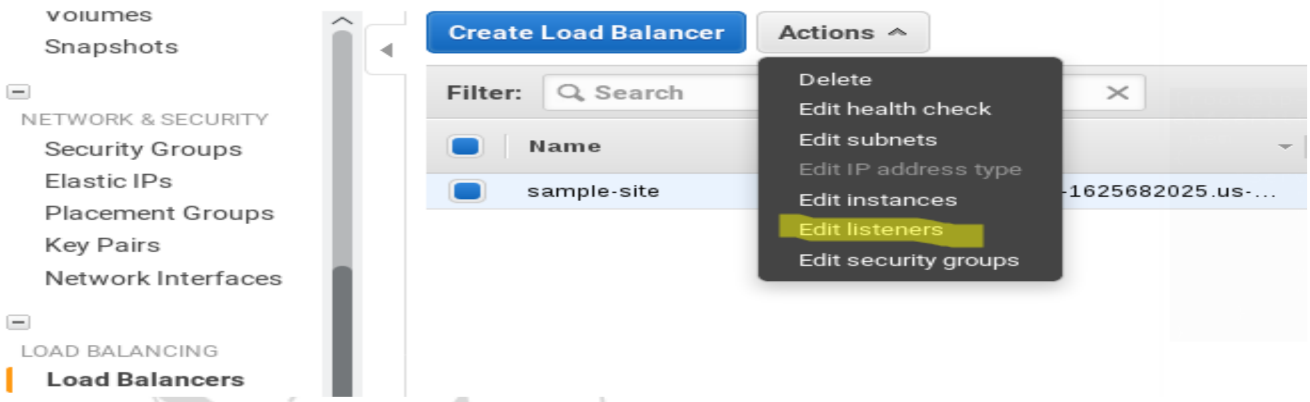
```
[root@ip-172-31-27-189 ~]# ls
get-pip.py  install.log  install.log.syslog  mycertificate.pem  mycert.pem  privatekey.pem  privky.pem  test
```

Step4: How to enable SSL on domain :

7.Upload keys into AWS

Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.

1. On the navigation pane, under **LOAD BALANCING**, choose **Load Balancers**.
2. Select your load balancer.
3. Goto Actions → Click on Edit listeners



4. Add HTTPS Protocol for now We can see **SSL Certificate**, choose **Change**.