SSL
for 44net devices

Mark Phillips, NI2O/G7LTT 11/2025 V1.0

Contents:

Abstract:

With the HTTPS emerging as the preferred protocol for serving web pages and other sensitive data it behoves 44net users to install SSL certificates to service this protocol. This document aims to demonstrate how a reader would install an SSL certificate onto their web server. Installing such a certificate will allow secure browsing transactions as well as ensure the authenticity of the web site or data connection in use.

Assumptions:

Familiarity with the Debian command line (Ubuntu and others also OK)
Familiarity with firewall applications

Requirements:

A functioning server connected to the 44net with correct routing and static IP addresses.
FQDN for this server is populated in the 44net DNS.
Familiarity with Debian based Linux systems (may work on MacOS too).
Apache based web server.

Stop the web server:

We must stop the apache web server from running in order to update its files and add out certificates.

```
sudo systemctl apache stop
```

Assuming this went well you will be returned to the command line prompt.

Enable Apache SSL module:

The Apache SSL module must be enabled before Apache will serve web pages over the SSL connection.

```
sudo a2enmod ssl
```

Install required software:

We'll be using the LetsEncrypt SSL service to generate our certificates. LetsEncrypt offer free SSL certificates for not-for-profit use and have their own open source software tools to both install and update the certificates. All LetsEncrypt certificates are valid for 3 months and will require to be refreshed near their expiration time.

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

Generate and install the SSL certificate:
We must run a command and modify its parameters in order to connect to LetsEncrypt and have them sign our certificate.

```
certbot --authenticator standalone --installer apache -d
your.server.fqdn
```

Where "your.server.fqdn" is the fully qualified domain name of your server. The name must already be registered in the DNS that managed the domain.

The certbot application will reach out from your server to LetsEncrypt and request that they sign a certificate that it will then download and install in the relevant file areas required by the Apache web server software.

If successful and you were paying attention you'll also see that the certbot application not only installed the relevant SSL key files within the Apache file structure but it also scheduled a job for 3 months from now that will renew the SSL key. If all goes well you should have to do nothing more than keep the server online.

```
Requesting a certificate for wx.ni2o.ampr.org

Successfully received certificate.
Certificate is saved at:
/etc/letsencrypt/live/wx.ni2o.ampr.org/fullchain.pem
```

```
Key is saved at:
/etc/letsencrypt/live/wx.ni2o.ampr.org/privkey.pem
This certificate expires on 2026-01-28.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew
this certificate in the background.

Deploying certificate
```

Reboot the server:

Yes, you could simply restart the Apache service but a reboot will ensure that the SSL service comes up following a reboot.

```
sudo reboot now
```