Table of Contents

Intro

This document describes a range of procedures and settings for the EpiSensor Gateway that can be used to increase security when deploying EpiSensor systems in critical environments. This is not intended to be a comprehensive guide to Linux systems administration, but rather covers security considerations for EpiSensor's application software and its communications requirements. For additional information, please contact support@episensor.com

Related Documents

Related installation and configuration documents are listed in the following table:

Document	Reference No.
EpiSensor ZGW-10 Datasheet	EPI-213-00
EpiSensor NGR-30-5 Datasheet	EPI-077-00
Gateway API User Guide	ESE-009-08

Applicable Hardware & Operating Systems

The following table lists the EpiSensor Gateway SKU's that this guide applies to:

EpiSensor Gateway	Operating System	Based on Hardware
NGR-30-5	Ubuntu Core 16	Dell Edge Gateway 3000 Series
ZGW-10	Ubuntu Core 20	Raspberry Pi Compute Module 4

Required ports and services

The following table lists the default protocols, ports and services the Gateway uses to communicate. Some may not be required, depending on the use case and data export configuration, and additional restrictions may be applied, for example to only permit access to specific hostnames on the public Internet.

Direction	Protocol	Port	Service	Description
Inbound	TCP	22	SSH	System maintenance and configuration
Outbound	UDP	123	NTP	For NTP Time synchronization
Inbound	TCP	8081	HTTP(S)	Web Interface / API for Gateway software. 8081 is the default port, if a non-default port is configured, it should also be allowed on the firewall.
Inbound	TCP	443	HTTPS	Web Interface / API for Gateway software. 443 is the default port; if a non-default port is configured, it should also be allowed on the firewall.
Inbound	TCP	8082	HTTP(S)	Web Interface / API for Edge software
Outbound	TCP	53	DNS	DNS lookups
Outbound	TCP	1883	MQTT	For non-secure (MQTT) communications. 1883 is the default port; if a non-default port is configured, it should be allowed on the firewall.
Outbound	TCP	8883	MQTTS	For secure (MQTTS) communications. 8883 is the default port; if a non-default port is configured, it should be allowed on the firewall.
Inbound	TCP	49152:65535	FTP(S)	For active-mode FTP data export

Automatic Software Updates

This section describes how to control automatic software updates on Ubuntu Core. Ubuntu Core, by default, provides a limited set of options to control when snap packages update. More information here on the standard options available.

For some use cases, particularly Gateways using cellular connections with limited data plans, or systems deployed in business critical environments, it's necessary to disable automatic updates. This application note describes three approaches that can be taken that will disable automatic updates on Gateways running Ubuntu Core.

Local Hosts File Entry

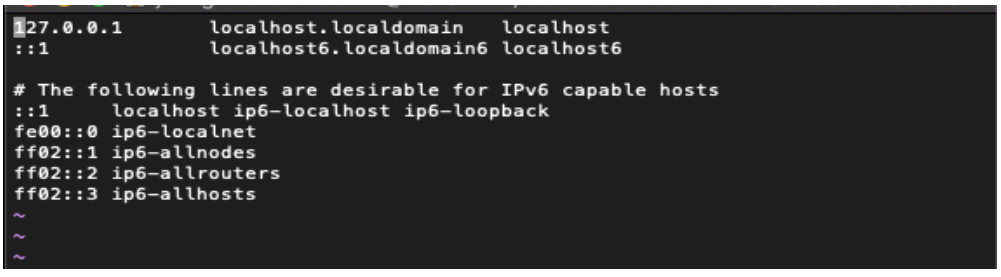
Adding an entry to the hosts file on the Gateway will cause automatic updates to fail by redirecting network requests for Ubuntu related domain names to localhost.

Log on to your EpiSensor Gateway running Ubuntu Core via SSH using the default credentials, and run the following command to edit the hosts file:

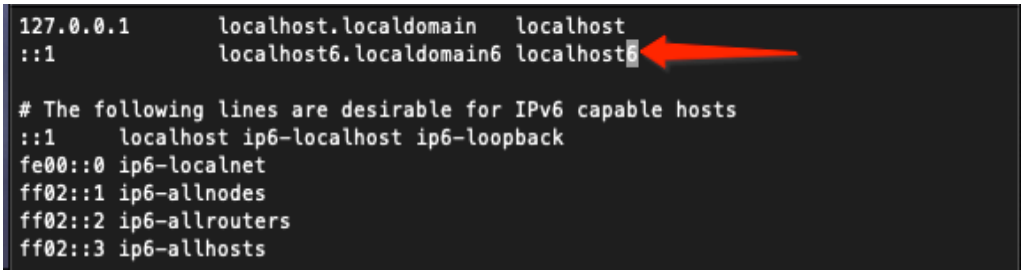
```
sudo vi /etc/hosts
```

Example:

The default file will appear as follows:



Move the cursor to the following position and type "o" to add a new line below the cursor position

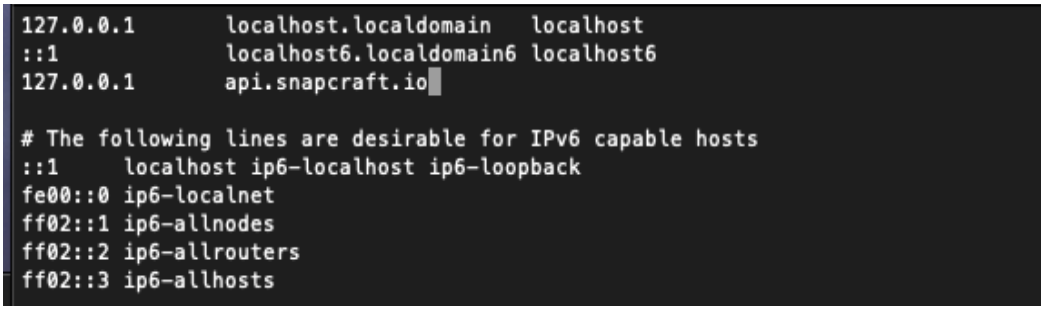


Add the following line (when the IP address has been entered, press tab to move over to URL):

```
127.0.0.1  api.snapcraft.io
```

Press “esc” in order to complete the line and move on, then type “o” to add the next line.

Example:



The following lines should be added to the hosts file:

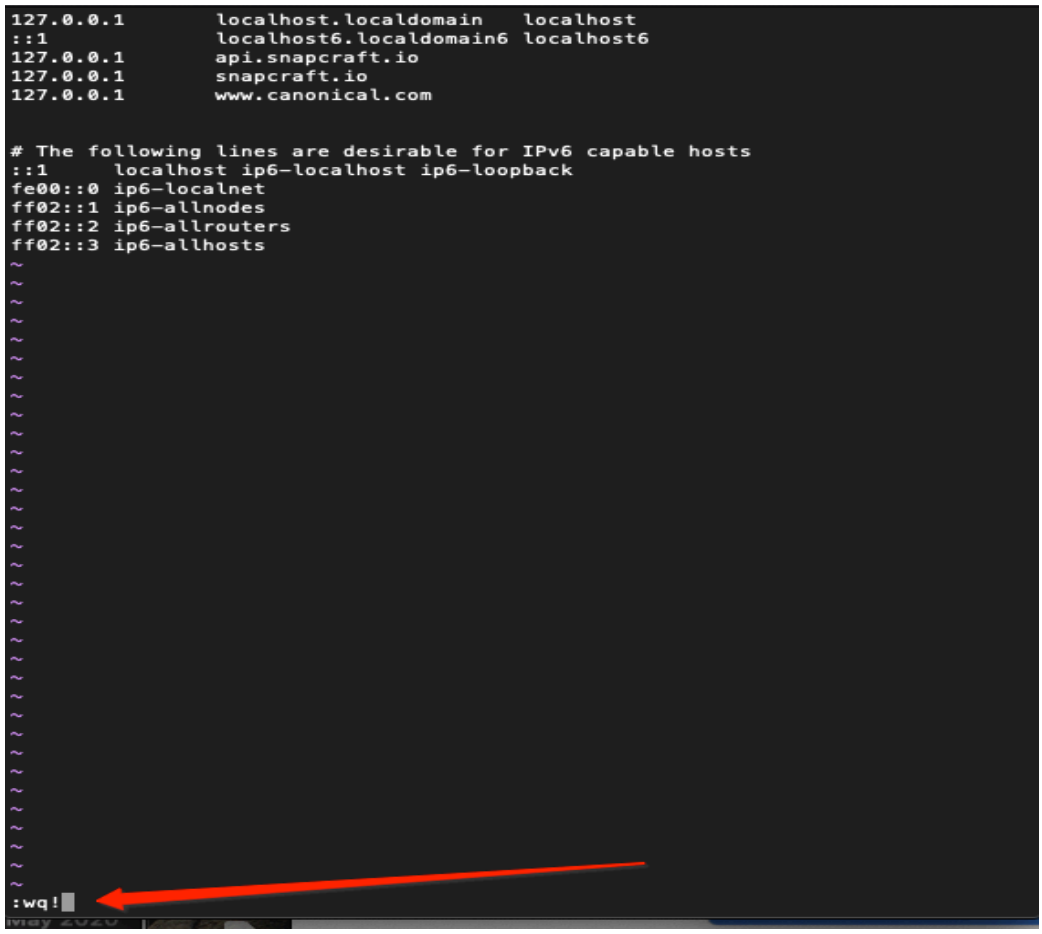
```
127.0.0.1  api.snapcraft.io

127.0.0.1  snapcraft.io

127.0.0.1  www.canonical.com
```

Once complete, press “esc” and type the following to save the changes:

```
:wq!
```

If you wish to exit without saving the changes, type:

`:q!`



To verify that the changes have been applied, run the following command:

`ping api.snapcraft.io`

You should receive a ping reply from “localhost.localdomain” rather than the external snapcraft.io server. It is not necessary to restart the Gateway for the changes to be applied.

```
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=1 ttl=64 time=0.059 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=2 ttl=64 time=0.072 ms
```

```
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=3 ttl=64 time=0.086 ms
```

Network Level Firewall Rules

For Gateways deployed on private cellular networks, or corporate networks - depending on the functionality available from the service provider - it may be possible to restrict outbound access to Ubuntu-related domain names that will prevent Gateways from checking for and downloading software updates. If this is supported by your provider, request that TCP traffic on all ports is disabled for the following three domains:

```
api.snapcraft.io
snapcraft.io
www.canonical.com
```

Dell Managed Refresh

For Dell Edge Gateway 3000 series hardware running Ubuntu Core, Dell has developed a 'snap' that when installed, will take control of the automatic software update process, and only update the system with manual user intervention. To install the Dell Managed Refresh snap, ensure that your Gateway has Internet access, and run the following commands:

```
snap install dell-managed-refresh
sudo snap set system refresh.timer="managed"
```

To verify that the refresh.timer setting has been applied, run:

```
snap get system refresh
```

and you should see the following output:

```
Key Value
refresh.timer managed
```

Then run the following command to verify that the Dell Managed Refresh snap has been installed:

```
snap list --all | grep "dell-managed-refresh"
```

and you should see the following output:

```
dell-managed-refresh  1.0                2     latest/stable  dell-inc
```

Firewall Configuration

A software firewall can be configured on an EpiSensor Gateway using “ufw” (Uncomplicated Firewall). The ufw snap is available for Ubuntu Core, which provides a user-friendly interface on top of standard iptables firewall rules. This section describes how to install and configure the firewall. The configuration should be adapted based on the services that are relevant for a particular customer site based on the “Required Ports and Services” section of this document.

Installation

Install the latest version of the ufw snap from the Ubuntu store as follows:

```
ubuntu@ubuntu:~$ snap install ufw
```

After the installation of the snap, the firewall is disabled (unless it was previously installed and enabled). You can check the status of the firewall as follows:

```
ubuntu@ubuntu:~$ sudo ufw status

Status: inactive
```

Configuration

The default settings for the firewall can be viewed in the file `/etc/default/ufw`. It is not recommended to edit this file to configure the firewall, but to use the ufw command line interface instead.

Block All Incoming

```
ubuntu@ubuntu:~$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
```

Allow All Outgoing

```
ubuntu@ubuntu:~$ sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
```

Allow SSH

```
ubuntu@ubuntu:~$ sudo ufw allow SSH
Rules updated
Rules updated (v6)
```

Alternatively, ssh may also be allowed by specifying the port number as follows: `sudo ufw allow 22/tcp`

Allow HTTP access on port 8081

```
ubuntu@ubuntu:~$ sudo ufw allow 8081/tcp
Rules updated
Rules updated (v6)
```

Allow FTP access on active port range 49152 – 65535

```
ubuntu@ubuntu:~$ sudo ufw allow 49152:65535/tcp
Rules updated
```

```
Rules updated (v6)
```

List the added rules

```
ubuntu@ubuntu:~$ sudo ufw show added

Added user rules (see 'ufw status' for running firewall):

ufw allow OpenSSH

ufw allow 49152:65535/tcp

ufw allow 8081
```

Enabling/Disabling

It is crucial that the firewall is configured correctly before it is enabled. If not, you risk being prevented from logging in remotely to fix the configuration. For example, installing and enabling the firewall without allowing ssh access will create such a scenario.

Checking the status of the firewall (disabled)

```
ubuntu@ubuntu:~$ sudo ufw status

Status: inactive
```

Enable the firewall

```
ubuntu@ubuntu:~$ sudo ufw enable

Command may disrupt existing ssh connections. Proceed with operation (y|n)? y

Firewall is active and enabled on system startup
```

Checking the status of the firewall (enabled)

```
ubuntu@ubuntu:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
SSH                        ALLOW       Anywhere
49152:65535/tcp            ALLOW       Anywhere
8081/tcp                   ALLOW       Anywhere
SSH (v6)                   ALLOW       Anywhere (v6)
49152:65535/tcp (v6)       ALLOW       Anywhere (v6)
8081/tcp (v6)              ALLOW       Anywhere (v6)
```

Resetting the firewall

```
ubuntu@ubuntu:~$ sudo ufw disable
Firewall stopped and disabled on system startup
ubuntu@ubuntu:~$ sudo ufw reset
Resetting all rules to installed defaults. This may disrupt existing ssh
connections. Proceed with operation (y|n)? y
Backing up 'user.rules' to '/etc/ufw/user.rules.20211020_103705'
Backing up 'before.rules' to '/etc/ufw/before.rules.20211020_103705'
```

```
Backing up 'after.rules' to '/etc/ufw/after.rules.20211020_103705'

Backing up 'user6.rules' to '/etc/ufw/user6.rules.20211020_103705'

Backing up 'before6.rules' to '/etc/ufw/before6.rules.20211020_103705'

Backing up 'after6.rules' to '/etc/ufw/after6.rules.20211020_103705'

ubuntu@ubuntu:~$ sudo ufw show added

Added user rules (see 'ufw status' for running firewall):

(None)
```

Unix User Accounts

This section describes the default user accounts available on an EpiSensor Gateway, the default sudo configuration, how to change the default user account password and how to add user accounts. Standard unix user account management features are available, and should be referred to for detailed user management requirements.

Default admin account

By default, there is a single “admin” user account configured for SSH access. The factory default password for the admin user is as follows: ITEpi*&1

The admin user is part of the sudoers group. To change the password for the admin account (when logged in as that user) run the following command and follow the instructions:

```
passwd
```

Adding users

To add a new user account, run the following command and follow the instructions (replacing ‘test’ with the required username):

```
sudo adduser --extrausers test
```

Sudo Configuration

The sudo package allows a regular user to run commands in an elevated context. This means a regular user can run commands normally restricted to the root account. The configuration file for Sudo is in /etc/sudoers, however it can only be edited by using the “visudo” command. By default, on NGR-30-5 Gateways, sudo can be used by members of the 'admin' and 'sudo' user groups.

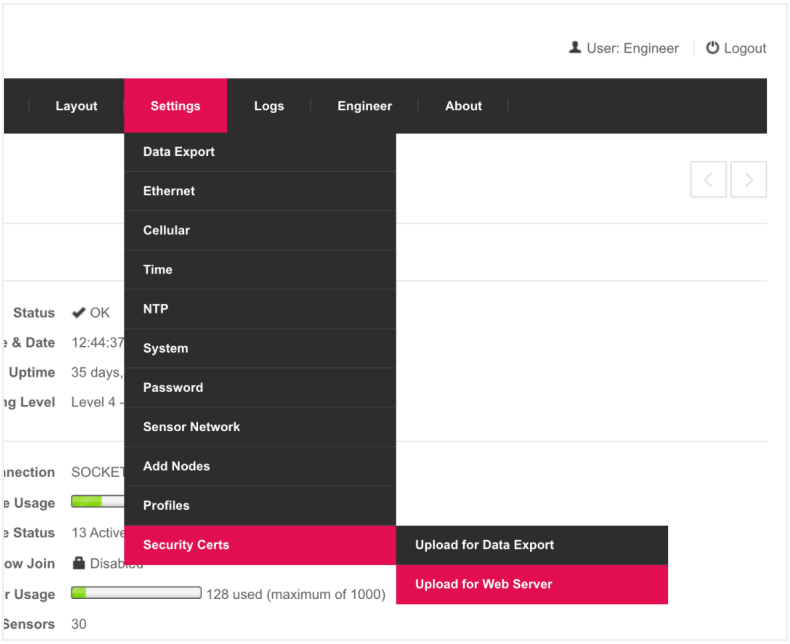
Securing Web Interfaces and APIs

The Gateway Web Interface and API, and the Edge web interface and API may be configured to use HTTPS instead of the HTTP (the default configuration). This section explains how to enable HTTPS for both applications, and also any considerations for communications between both applications after enabling HTTPS.

Enabling HTTPS for epi-gateway

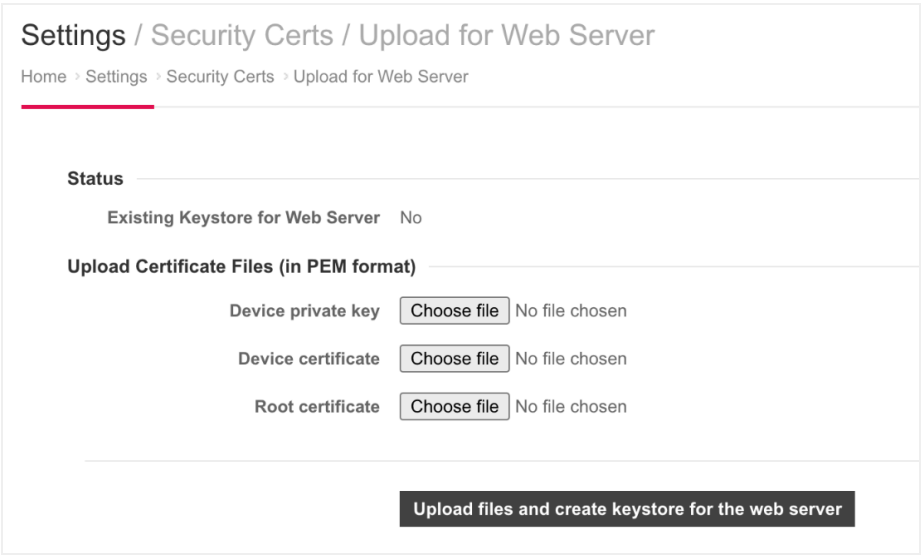
Before enabling this configuration, it is necessary to firstly upload the security certificates for HTTPS to the gateway.

Security certificates may be uploaded using the Settings → Security Certs → Upload for Web Server page as show in the following screenshot:



The “Upload for Data Export” tab should be used only when uploading Security Certificates for MQTT Data Export, as required.

The “Upload for Web Server” page is shown in the following screenshot:



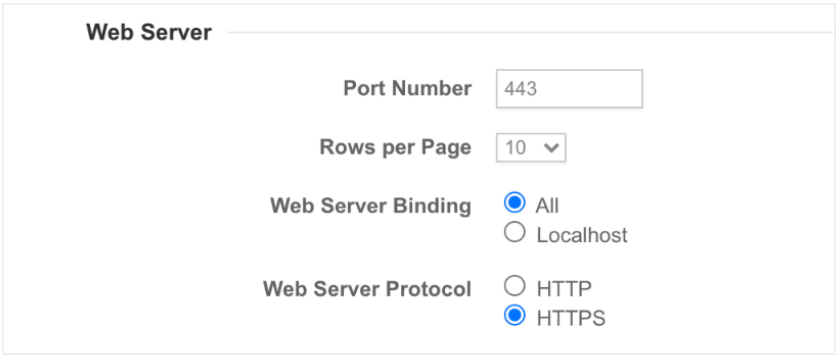
When Security Certificates are uploaded to the Gateway, a keystore will be generated which will be used for HTTPS authentication by the Web Server. It will be indicated on this page whether or not Security Certificates have already been uploaded for the Gateway.

If there is an existing keystore on the Gateway and you proceed to upload a new set of Security Certificates, the existing keystore will be overwritten. The existing keystore may also be deleted from the Gateway using the Action dropdown list on Settings → System page.

The Security Certificates uploaded to the Gateway for the Web Server should be PEM-encoded X.509 format. Three files are required as follows:

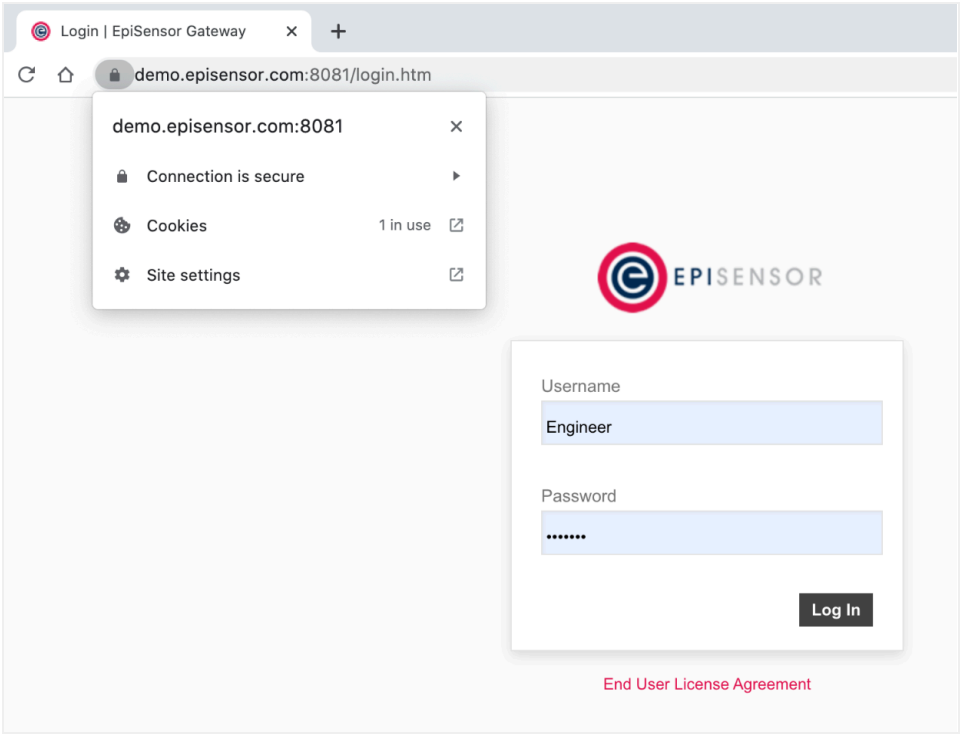
- Device Private Key: The private key file for this device.
- Device Certificate: The certificate file for this device which verifies the private key for this device.
- Root Certificate: The root certificate is the certificate issued by the trusted certificate authority (CA).

When the certificates have been uploaded successfully, HTTPS may be enabled. This can be done from the Engineer Page (only available to users with Engineer privileges) as follows:



The Web Server protocol should be changed from HTTP to HTTPS. In addition, you may wish to change the port number. The default port number for HTTPS is 443. Note: If anything other than port 443 is specified, the webserver will be accessible on the specified non-default port number and on port 443.

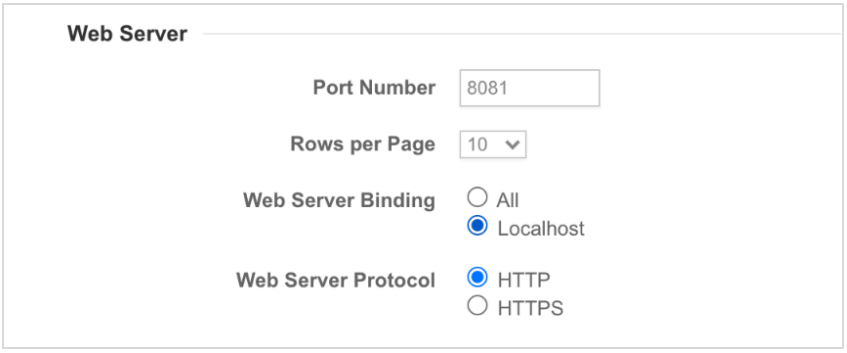
The Gateway must restart for these settings to take effect, after which the Gateway's web interface and API will only be accessed using HTTPS.



Note: in the above example, HTTPS has been enabled on port 8081 and 443.

As an extra security precaution, the Web Interface (and API) may be configured to be bound to localhost. In other words, the Web Interface and API would only be accessible from the localhost IP address.

This setting is also available on the 'Engineer' page as shown in the following screenshot:



Enabling HTTPS for epi-edge

To enable TLS on the Edge API and web interface, open the settings.js file in a text editor with the following command:

```
sudo vi /var/snap/epi-edge/current/settings/settings.js
```

And add the following to the end of the file, referencing the correct key and cert file names and locations:

```
module.exports.https = {
    key: fs.readFileSync(`${SNAP_DATA}/certs/ssl/privkey.pem`),
    cert: fs.readFileSync(`${SNAP_DATA}/certs/ssl/fullchain.pem`)
};
```

Note: the key and cert files should be relative to "/var/snap/epi-edge/current/" (which is the directory that the SNAP_DATA environment variable resolves to).

Restart Edge with "snap restart epi-edge" and the web server and API should now have HTTPS enabled.

Considerations for communications between apps

To enable

Gateway User Accounts

Additional user accounts may be added using the Gateway Web Interface. If no additional user accounts are created the Gateway will have two default users, Administrator and Engineer.

Note: If any additional user accounts are created the default users (Administrator and Engineer) will be deleted from the system.

It will be possible to add/remove additional named users accounts in an upcoming Gateway software release.

A hash of the password for each user account is stored on a Java keystore on the Gateway's filesystem. The password is encrypted with a key that's randomly generated during software installation for each system.

Note: if a non-default Gateway password for the admin account is set (or if the admin account is disabled/removed), then Edge will need to be configured with the updated password for HTTP API access.

Edge User Accounts

Additional user accounts may also be added using the Edge application. If no additional user accounts are created, Edge will have one default user account (admin) and the default password will be set to: A1

```
"adminAuth": {
   "type": "credentials",
   "users": [{
       "username": "admin",
       "password": "$xxxxxx/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx/xxxxxx/xxxxxx",
       "permissions": "*"
   }]
```

The screenshot above shows the "adminAuth" section of a settings JSON file for Edge that's stored in the following location on the Gateway:

`/var/snap/epi-edge/current/settings/settings.json`

When adding a new user, the 'users' array in the JSON document above should be extended to include the new username and a secure hash of the password for that user, which can be generated using the bcrypt library. Permissions for all Edge users should be set to *

BIOS Password

Run the following commands to set a BIOS password. When interacting with the BIOS after a password has been set, pass in "--ValSetupPwd=" as an argument with the current BIOS password.

Set the BIOS password:

```
sudo dcc.cctk --setuppwd=<new-password>
```

Change the BIOS password:

```
sudo dcc.cctk --setuppwd=<new-password> --ValSetupPwd=<old-password>
```

Wireless Interfaces

Depending on the version of EpiSensor Gateway hardware used, there may be onboard Cellular, Wi-Fi and Bluetooth radios. On the NGR-30-5 (Based on Dell Edge Gateway 3000 series hardware), the wireless radios can be enabled or disabled in the BIOS.

WWAN

To check the status of the Cellular radio, run:

```
sudo dcc.cctk --WirelessWwan
```

To enable the Cellular radio, run:

```
sudo dcc.cctk --WirelessWwan=Enabled
```

To disable the Cellular radio, run:

```
sudo dcc.cctk --WirelessWwan=Disabled
```

WLAN

To check the status of the Wi-Fi radio, run:

```
sudo dcc.cctk --WirelessLan
```

To enable the Wi-Fi radio, run:

```
sudo dcc.cctk --WirelessLan=Enabled
```

To disable the Wi-Fi radio, run:

```
sudo dcc.cctk --WirelessLan=Disabled
```

Log Monitoring

Some environments may require log monitoring to be enabled, so logs can be streamed to a remote server where anomalies can be detected. This section describes how this can be achieved by installing and configuring an additional EpiSensor snap called “epi-logstream”.

Installation

Create a directory in /home/admin/INSTALL if one does not already exist. Copy the epi-logstream .snap and .md5 files from your local computer to the Gateway (using for example Filezilla or SSH). Also copy over the install script. The files should all be copied into the INSTALL directory.

```
sh episensor@172.31.255.1  (replace with relevant IP address as appropriate)

cd INSTALL

./install_epi-logstream.sh -m clean
```

For an upgrade to an existing deployment run the install script in mode upgrade:

```
./install-epi-logstream.sh -m upgrade
```

Configuration

The following parameters may be configured in EpiSensor LogStream :

- Syslog Host (default value of localhost)
- Syslog Port (default value of 514)
- Syslog Protocol (default value of UDP, SSL and TCP also supported)
- Syslog Format (default value of bsd, RFC5424 also supported)
- Journalctl Format used as parameter to --output= (default value of short)
- Journalctl Show parameter (default value of -f --no-hostname)
- Log4J Builder Log Level (default value of WARN, DEBUG will provide more details)

The daemon may be reconfigured using snap set as shown in the following examples:

```
snap set epi-logstream syslog.host=192.168.8.8

snap set epi-logstream syslog.port=5314

snap set epi-logstream syslog.protocol=SSL

snap set epi-logstream syslog.format=bsd

snap set epi-logstream journal.format=json

snap set epi-logstream journal.show=-f

snap set epi-logstream log.level=DEBUG
```

The configuration of the security certs for SSL is To Be Defined.