

# REST HTTPS

- [Settings](#)
- [Incoming HTTPS](#)
  - [Profiles](#)
  - [Certificates](#)
- [Outgoing HTTPS](#)

## Settings

By default, non-encrypted REST / HTTP is used for interaction with the web server. In this case you can easily monitor HTTP-traffic flowing between WCS and the web server.

WCS Manager listens HTTP on the port 9091, and HTTPS on the port 8888.

Settings of the manager module of WCS related to HTTPS are listed in [wcs-manager.properties](#):

Setting	Default value
manager.enable_https	true
manager.https_port	8888
manager.keystore	wss.jks
manager.keystore_password	password
spring.profiles.active	dev
rest_template.allow_self_signed	true

## Incoming HTTPS

### Profiles

Incoming HTTPS requests are used only for 'sendData' callbacks. See more in the [Web Call Server - Call Flow](#) documentation.

These requests go from the web server to WCS, and WCS takes the role of an HTTPS server here.

To switch between HTTP and HTTPS modes the following profiles are used:

- dev
- production

These profiles can be specified in the WCS Manager settings, in the [wcs-manager.properties](#) file.

Example:

```
spring.profiles.active = dev
```

Even if the production profile is on, HTTPS can be turned off using the `manager.enable_https=false` setting. In this case the HTTPS port will not be listened by the WCS server.

### Certificates

By default, Web Call Server uses the same keystore for REST / HTTPS certificates as it uses for Websockets - `wss.jks`, specified in the `manager.keystore` parameter.

Managing this keystore is thoroughly described in the [Websocket SSL](#) section.

## Outgoing HTTPS

When querying the web server, the WCS server inspects the URL of the request and initiates an HTTPS query if that URL starts with `https`. For example, `https://localhost/ConnectionStatusEvent`. To query the web server that uses a self-signed certificate, use the `rest_template.allow_self_signed=true` setting.

If this parameter is set to `false` queries will not pass through, and you will need to configure your web server to work with actual SSL certificates issued by the certification center. To do this, please refer to the documentation of your web server.