Virtual/Software/Docker Gateway

Seneca now offers a flexible alternative to the purchase of a turnkey hardware gateway. For customers that wish to use existing infrastructure, whether virtualized or bare metal, the setup of the xConnect Gateway software can be installed with a few easy steps.

Minimum Requirements

Virtual Machine

Hardware

- 2 vCPU
- 8GB RAM
- 60GB of Storage
- 2 Network Adapters
    - 1 adapter connected to vSwitch of assets being monitored (Agent Network)
    - 1 adapter connected to vSwitch that has outbound internet access (Internet-enabled Network)

Software

- Any Debian-based Linux Distribution (Recommend Ubuntu Server 18.04 LTS or equivalent)
- Docker.io and Docker-Compose

Bare Metal (Physical Device)

Hardware

- Core i3 or greater
- 8GB RAM
- 60GB of Storage
- 2 Network Adapters
    - 1 adapter connected to Switch of assets being monitored (Agent Network)
    - 1 adapter connected to Switch that has outbound internet access (Internet-enabled Network)

Software

- Any Debian-based Linux Distribution (Recommend Ubuntu Server 18.04 LTS or equivalent)

- Docker.io and Docker-Compose
- Nano or text editor of choice

Installation

1. Open a shell into your Linux installation
2. `apt install docker.io docker-compose` to install the docker container engine
3. `mkdir /etc/xconnect && cd /etc/xconnect`
4. `git clone https://github.com/senecaxconnect/xconnect_gateway_docker`
5. `nano gw.env`
6. Modify gw.env with the provided API and SecretKey provided by the Seneca xConnect Support Team. Replace the placeholders with the provided keys:
   `SELENE_APIKEY=<APIKEYHERE>`
   `SELENE_SECRETKEY=<SECRETKEYHERE>`
7. To start the gateway software, execute the following command:
   `MQTT_PORT={Desired MQTT Input Port} GW_NAME={Desired Gateway Name} docker-compose up -d`
   Replace {Desired MQTT Input Port} with 1883 unless otherwise instructed
   Replace {Desired Gateway Name} with XCGW-*some_identifying_name*

   Example: `MQTT_PORT=1883 GW_NAME=XCGW-DOCK01 docker-compose up -d`