Akraino ELIOT Blue Print

Akraino Pods

Akraino Shared Community Lab provides ThunderX2 Pod 2 and Intel Pod as a stable base platform for EdgeX validation.

ThunderX2 Pod 2 (ARM)

The ThunderX2 Pod 2 consists of 3 Arm®v8 based Gigabyte R281-T91 servers and 1 Ampere HR330A jumphost server. Refer to Akraino ThunderX2 Pod 2 Page for more details.

Server Name	Public Network Address	OS Installed
gigabyte4	10.11.4.14	CentOS 7.6
gigabyte5	10.11.4.15	Ubuntu 18.04
gigabyte6	10.11.4.16	Ubuntu 18.04
gigabyte-jumphost2	10.11.4.18	Ubuntu 18.04

Intel Pod (X86)

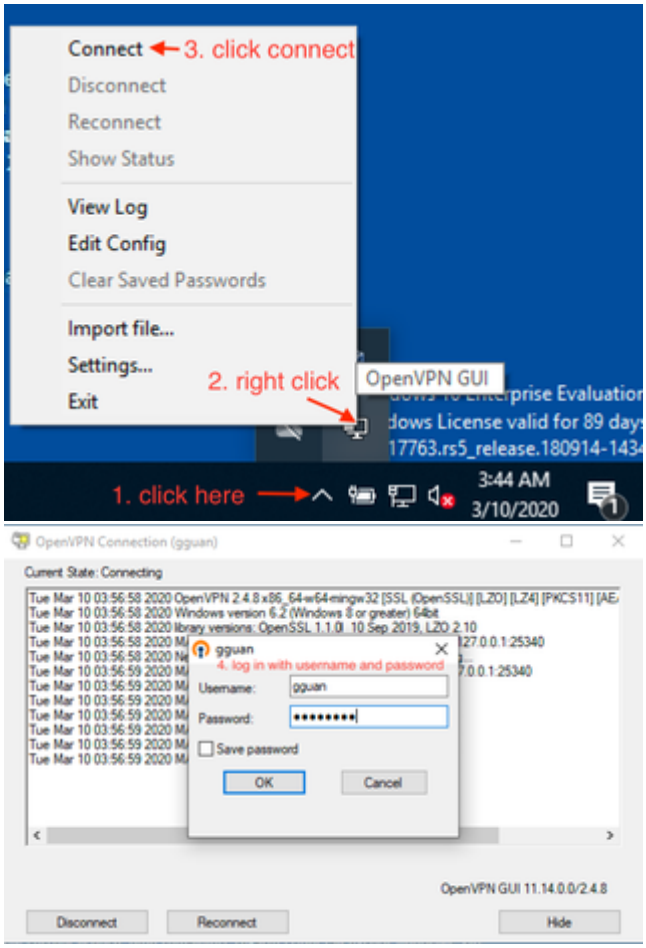
The Intel Pod consist of 4x Intel LWF2208IR540605, 2x Intel LWF2208IR540606, 2x Intel NUC8i3CYSM, 2x NUCi7BEK. Refer to Akraino Intel Pod Page for more details.

Server Name	Public Network Address	OS Installed	Machine Type
intel1	10.11.8.11	CentOS 8	Intel LWF2208IR540605
intel2	10.11.8.12	CentOS 8	Intel LWF2208IR540605
intel3	10.11.8.13	CentOS 8	Intel LWF2208IR540605
intel4	10.11.8.14	CentOS 8	Intel LWF2208IR540605
intel5	10.11.8.15	TBD	Intel LWF2208IR540606
intel6	10.11.8.16	TBD	Intel LWF2208IR540606
intel-nuc1	10.11.8.17	Windows 10	NUC8i7BEK
intel-nuc2	10.11.8.18	Windows 10	NUC817BEK
intel-nuc3	10.11.8.19	Windows 10	NUC8i3CYSM
intel-nuc4	10.11.8.20	Windows 10	NUC8i3CYSM

Accessing the Servers

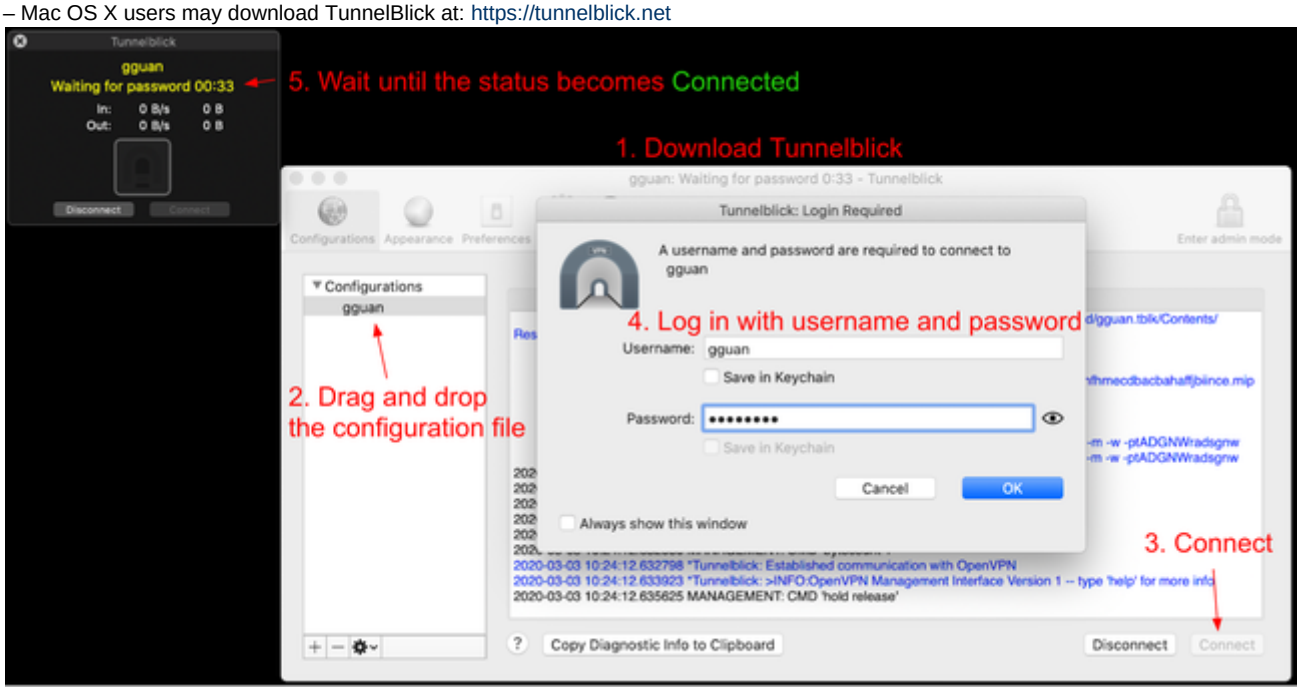
1. Send an email to akraino-lab@iol.unh.edu to request an account. The email should contain your full name, the Akraino Blueprint you are working on, the Pod you would like access to, and your public ssh key. Update the Users section on EdgeX on ELIOT Blueprint Page wo uld be appreciated.
2. Use the username, password and openvpn client configuration provided in the email from akraino-lab@iol.unh.edu to connect to the

VPN.
– The Windows openvpn installer can be installed from: https://openvpn.net/index.php/open-source/downloads.html
put the configuration file in the config directory, usually **c:\program files (x86)\openvpn\config**, and then right click the tray icon to connect



– Linux users can refer to: OPENVPN: Connecting To Access Server With Linux

– Mac OS X users may download TunnelBlick at: https://tunnelblick.net



3. [The first time Log in] Change the password from https://resolute.akr.iol.unh.edu/ipa/ui once the VPN is connected.
4. Access to the servers in the Pods via ssh as root. For example, access gigabyte4:

```
$ ssh root@10.11.4.14
```

Setting Environment

To run EdgeX blackbox test, the following packages are required:

- docker
- docker-compose
- git

For CentOS operating system(gigabyte4, intel1, intel2, intel3, intel4), YUM and DNF(the next-generation replacement for YUM) are the preferred tools for installing software.

** Ensure `firewalld` is **NOT** active (running). Use the following steps to check the status of `firewalld` and stop/disable it.

```
# Check firewall status
$ systemctl status firewalld
```

```
# Stop firewall
$ systemctl stop firewalld
# OR
$ service firewalld stop
```

```
# To permanently disable firewall even after system reboot
$ systemctl disable firewalld
```

Install Docker

Install Docker Engine - Community using the repository. More information see: Get Docker Engine - Community for CentOS and Get Docker Engine - Community for Fedora.

1. Uninstall old versions

```
$ dnf remove docker \
              docker-client \
              docker-client-latest \
              docker-common \
              docker-latest \
              docker-latest-logrotate \
              docker-logrotate \
              docker-selinux \
              docker-engine-selinux \
              docker-engine
```

2. Set up the Docker repository

```
$ dnf -y install dnf-plugins-core
$ dnf config-manager \
    --add-repo \
    https://download.docker.com/linux/centos/docker-ce.repo
```

3. Install the *latest version* of Docker Engine - Community and containerd.

```
$ dnf install docker-ce docker-ce-cli containerd.io
```

** If prompted to accept the GPG key, verify that the fingerprint matches`060A 61C5 1B55 8A7F 742B 77AA C52F EB6B 621E 9F35`, and if so, accept it.

4. Start Docker

```
$ systemctl start docker
```

Install Docker-Compose

Install docker-compose using `pip3`. Use a virtualenv is recommended because many operating systems have python system packages that conflict with docker-compose dependencies.

1. Install pip3

```
$ dnf install python3-pip
```

2. Install and set up virtualenv

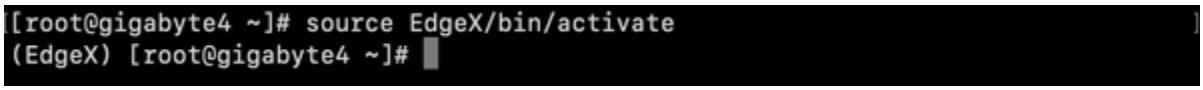
```
$ pip3 install virtualenv
$ virtualenv EdgeX
```

** virtualenv creates a folder named `EdgeX` under the current working directory.

3. Begin using the virtual environment

```
$ source EdgeX/bin/activate
```

The name of the current virtual environment will now appear on the left of the prompt (e.g. `(EdgeX)[root@gigabyte4 ~]#`) to let you know that it's active.



4. Install docker-compose

```
$ pip3 install docker-compose
```

Install Git

```
$ dnf install git
```

Running Blackbox Testing

1. Clone the repo from `https://github.com/edgexfoundry/blackbox-testing.git` and use `fuji` branch

```
$ git clone https://github.com/edgexfoundry/blackbox-testing.git
$ cd blackbox-testing
$ git checkout fuji
```

2. Execute docker-compose command must use the virtual environment

```
$ source ~/EdgeX/bin/activate
```

3. Run Blackbox Testing

```
$ source bin/env.sh # env.sh for x86 and arm64_env.sh for arm
$ bash deploy-edgeX.sh
$ bash ./bin/run.sh -all # run all tests
```

[Optional] Getting Allure Report

Allure Docker Service allows you to see up to date reports simply mounting your allure-results directory in the container. Refer to allure-docker-service. The directory of the XML reports is `blackbox-testing/bin/testResult`.

Using Docker Compose

```
version: '3.4'
services:
  allure:
    image: "frankescobar/allure-docker-service"
    container_name: allure
    environment:
      CHECK_RESULTS_EVERY_SECONDS: 1
      KEEP_HISTORY: "TRUE"
    ports:
      - "4040:4040"
      - "5050:5050"
    volumes:
      - /blackbox-testing/bin/testResult:/app/allure-results
```

** The `/app/allure-results` directory is inside of the container. You MUST NOT change this directory, otherwise, the container won't detect the new changes.

See the report, gigabyte4 for example, at: http://10.11.4.14:4040 or http://10.11.4.14:5050.