EEC172 Prelab 4

5/13/2024

Name:

SID:

Difficulty Rating

Question 1.

(Mix & Match) Match the following terms to their definitions:

(1) Device Shadow

(2) AWS IoT "thing"

(3) AWS IoT Policies

(4) AWS IoT Rules

(5) AWS SNS Topics

(a) a set of permissions that define what actions a user or device can perform on AWS IoT resources

(b) statements that specify one or more actions to perform when a message meets a particular criteria

(c) a virtual representation of a physical device's state

(d) a communication channel to which a notification/message can be sent

(e) a virtual entity that represents a device or sensor

Difficulty Rating

Question 2.

(Multiple Choice) Which HTTP request methods can be used to interact with the AWS IoT Device Shadow?

a. GET
b. HEAD
c. POST
d. PUT
e. DELETE
f. CONNECT

Difficulty Rating

Question 3.

(Free Response) At a high level, briefly describe how an AWS IoT Device Shadow can be useful when the physical device goes offline.

4. Smart Thermostats?

Consider the control flow diagram of an AWS IoT-enable themostat and A/C system on the right. The thermostat and A/C system are designed to do the following:

- The thermostat begins air conditioning (A/C) when it detects the current temperature is outside the user-configured temperature thresholds

- **The A/C always brings the temperature to the relevant threshold in exactly 5 min**

- **After hitting the threshold, the thermostat keeps the A/C on for an additional 1 min, which always brings the temperature to an extra 1 degree above or below the relevant threshold.**

- Synchronization occurs immediately after startup and then at 15 min intervals so long as there is network connection.

- The thermostat will try to update the shadow if it detects a temperature change or changes the A/C status.

- **Assume updates to the shadow occur instantaneously if they are successful**

The thermostat thresholds can be set on the physical device, or remotely via the device shadow, and the thermostat attempts to regularly synchronizes with its shadow using the following process:

1. the thermostat checks for network connection, and if there is none, skips synchronization.

2. the thermostat makes a get request to check if any "desired" properties have been remotely updated.

3. if there are modified "desired" properties, the thermostat updates these properties locally.

4. the thermostat then makes an update request to update the "reported" properties with its local state.

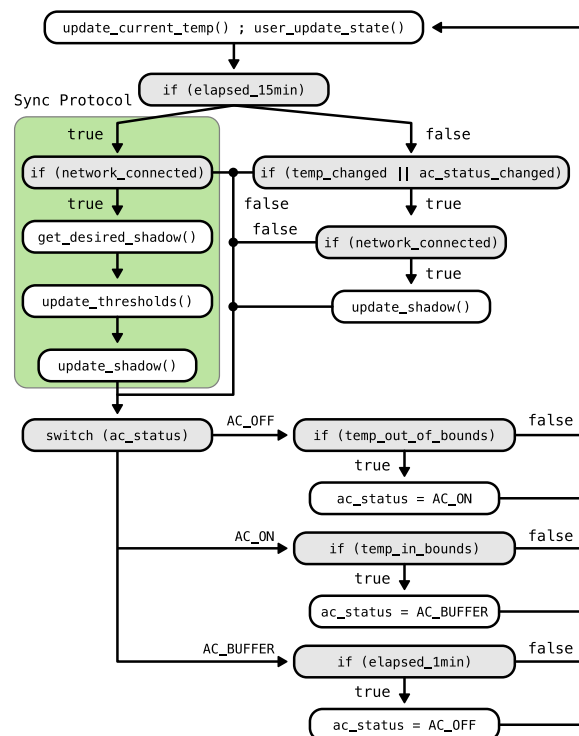5. the thermostat initiates A/C as necessary.

Figure 1: Pseudocode Control Flow Diagram of Themostat Main Loop

Difficulty
Rating

Question 4.

(Free Response) Assume that the thermostat starts out perfectly in sync with its device shadow, in the following state:

```
{
    "state": {
        "desired": {
            "cool-threshold": 75,
            "heat-threshold": 66,
        },
        "reported": {
            "cool-threshold": 75,
            "heat-threshold": 66,
            "current-temp": 68,
            "a/c status": "off"
        }
    }
}
```

The thermostat was last synchronized with its shadow at t=0min. Then, the following sequence of events occur:

1. at t=5min, the thermostat detects a new temperature of 67

2. at t=7min, the network goes down

3. at t=12min, the user successfully makes a remote update request with `{ "desired": { "heat-threshold": 72 }}`

4. at t=17min, the user manually sets the thermostat to `{ "cool-threshold": 78, "heat-threshold": 70 }`

5. at t=25min, the network comes back up

Assuming the thermostat has followed its programming the entire time, and that no forces are acting to change the temperature other than those explicitly stated. **What is the device shadow state at t=25min and t=35min? Explain what has occured to justify your answer.**

Difficulty
Rating

Question 5.

(Free Response) Consider the synchronization protocol of the thermostat from Question 4. Note that it is possible for older state change requests to supercede newer ones if the device has been disconnected for a considerable length of time.

What information could we add to the shadow and how could we modify the synchronization protocol and the device's behavior so that older requests do not supercede newer ones? Write out a new example shadow state in json format, as well as all the step of your new synchronization protocol and any additional changes you would make to the device behavior.