RATT MQTT SPECIFICATION

Version 0.1 - December 2, 2018

New Hampshire's First and Largest Makerspace
http://www.makeitlabs.com/

Base Topic

The base topic is the leading part of the topic name that is used for all RATT-related MQTT messaging.

> Base topic path, default is `ratt/`

The base path is configurable via the .ini file via MQTT.BaseTopic

The remainder of this document will use '`ratt/`' as the base path for simplicity. Exact topic path will vary depending on site configurations.

Broadcast Topics - Control

MQTT messages sent from a client and meant to be received by all RATT nodes. This is useful for initiating site-wide control such as forcing an ACL update across all nodes, for example.

> Broadcast topic path is `ratt/broadcast/control/[subtopic]/...`

Broadcast Subtopics

- `acl/update` – Requests that the node perform an ACL update immediately. This might be issued by a web/database backend if changes to permissions have been made.
  - `No payload`

Targeted Topics - Control

MQTT messages sent from a client and meant to be received by a single RATT node. Thus, they are addressed specifically to that node via a node ID.

> Targeted topic path is `ratt/control/node/[node_id]/[subtopic]/...`
>
> Where *node_id* is the primary MAC address of the node, all lowercase and no colons, e.g. *b827ebe669a7*

Testing

Mosquitto has a command line tool that can be used for testing to publish messages to RATT nodes. The tool will need access to valid SSL keys, see https://github.com/makeitlabs/ratt/blob/master/deployment/certs/mqtt/README.md

Make sure a client_test cert has been created per instructions above, and run these commands from `~/ratt/deployment/certs/mqtt`

To send a targeted message with a payload, e.g:

```
mosquitto_pub --cafile ssl/certs/ca.crt --cert
ssl/certs/client_test.crt --key ssl/private/client_test.key -h
devel -t ratt/control/node/b827ebe669a7/personality/lock -m
"Locked out for a test."
```

To send a targeted message with no payload, e.g.:

```
mosquitto_pub --cafile ssl/certs/ca.crt --cert
ssl/certs/client_test.crt --key ssl/private/client_test.key -h
devel -t ratt/control/node/b827ebe669a7/personality/unlock -n
```

Targeted Subtopics - Control

- `acl/update` – Requests that the node perform an ACL update immediately. This might be issued by a web/database backend if changes to permissions have been made.
  - `No payload`
- `personality/lock` – requests that the node go into lockout mode when next idle. *MQTT message retention is suggested for this topic, so if a RATT node goes offline and comes back online, it will receive the lockout state when it reconnects.*
  - `[optional lock reason text]` - If supplied, the lock reason text will be displayed on the RATT screen while the tool is locked out.
- `personality/unlock` - requests that the node go out of lockout mode and back to normal operation.

Targeted Topics - Status

MQTT messages sent from a single RATT node, meant to convey status about that node.
Thus, they are addressed specifically from that node via a node ID.

> Targeted topic path is `ratt/status/node/[node_id]/[subtopic]/...`
>
> Where *node_id* is the primary MAC address of the node, all lowercase and no colons, e.g. *b827ebe669a7*

Testing

Mosquitto has a command line tool that can be used for testing to see status messages from
RATT nodes. The tool will need access to valid SSL keys, see
https://github.com/makeitlabs/ratt/blob/master/deployment/certs/mqtt/README.md

Make sure a client_test cert has been created per instructions above, and run these commands
from `~/ratt/deployment/certs/mqtt`

To subscribe to status from all nodes:

```
mosquitto_sub --cafile ssl/certs/ca.crt --cert
ssl/certs/client_test.crt --key ssl/private/client_test.key -t
ratt/status/node/# -v -h devel
```

To subscribe to status from a specific node:

```
mosquitto_sub --cafile ssl/certs/ca.crt --cert
ssl/certs/client_test.crt --key ssl/private/client_test.key -t
ratt/status/node/[node_id]/# -v -h devel
```

Targeted Subtopics - Status

- `wifi/*` - WiFi subsystem status. These messages are published together, periodically. *(every 15 seconds currently, likely to be extended... perhaps with some kind of collected stats for the time period (e.g. max/min level & quality))*
  - `wifi/essid`
    - `[Current WiFi ESSID]`
  - `wifi/freq`
    - `[Current WiFi radio frequency]`
  - `wifi/level`
    - `[Current WiFi signal level]`
  - `wifi/quality`
    - `[Current WiFi signal quality}`
- `personality/*` - Personality events
  - `personality/state`
    - `[statename.phasename]` - personality state machine has changed states. State names vary depending on personality, phases are one of ENTER, ACTIVE or EXIT.
    - Published only when state/phase changes
  - `personality/login`
    - `[error <reason> | denied [account.name] | allowed [account.name]]`
      - error indicates RFID reader error, optional reason string follows
      - denied indicates no permission for user [account.name]
      - allowed indicates valid permission for user [account.name]
    - Published only upon read of RFID
  - `personality/error`
    - `[safety [account.name]]` - indicates safety check failure
      - Published only upon safety failure event
  - `personality/activity`
    - `[[active | inactive] [account.name]]`
      - active indicates tool is actively being used
      - Inactive indicates tool is not actively being used
    - Published each time the tool activity state changes
  - `personality/logout`
    - `[explicit | timeout]` - indicates a reason for the logout. Explicit logout means the user pressed the exit key. Timeout means the tool was inactive for too long. *May be extended to include 'tool lost power' reason.*
    - `[account.name]`
    - `[enabled seconds]` - how long the session lasted, in seconds
    - `[active seconds]` - how long the tool was active during session, in seconds

- - - - ■ `[idle seconds]` - how long the tool was idle during session, in seconds
        - ■ Published once only when the user is logged out.
    - ○ `personality/lockout`
        - ■ `pending` - indicates a lockout condition is pending but has not yet taken effect
        - ■ `locked` - indicates a lockout condition has taken effect
        - ■ `unlocked` - indicates a lockout condition is no longer in effect
- ● `system/*` - System events
    - ○ `system/power`
        - ■ `lost`
        - ■ `restored`
        - ■ `shutdown`
    - ○ `system/issue`
        - ■ `[account.name] [description of issue]`