WoT device lifecycle

Elena Reshetova, Intel
Reasoning

• A single view on lifecycle is utmost important to define security correctly (and not just security)
• Helps defining security scope
• Currently described in https://rawgit.com/w3c/wot-security/working/index.html#lifecycle-of-a-wot-device
• Should probably live in the main architecture document
Lifecycle overview

1. Manufactured
2. Installation & Commissioning
3. Security Provisioning
4. Operational state
5. Maintenance & security updates

- Security provisioning done?
- Installation & commissioning step done?
- D. Re-installation required
- A. General maintenance or security-related updates done
- B. Maintenance or security-related updates required
- C. Security re-provisioning needed
- E. Full decommissioning
States and Scope

• Manufactured
• Installation/Commissioning
• Security provisioning
• Operational state
• Maintenance and Security updates

Operational state is the only one in-scope
More on lifecycles

• What about below lifecycles?
  • WoT Script lifecycle (created – provisioned to device – operational – updates – decommissioning from device)
  • WoT Thing & TD lifecycle (created – made available (discovered) – updates - deletion)
    • Do they have the same lifecycle?
    • Can be partly deducted from Scripting API (if created using scripting API)

• How do they fit into overall WoT device lifecycle?
  • i.e. WoT script can be provisioned to the device during
    • WoT device manufacturing stage – flashing the whole stack with script(s)
    • WoT installation & commissioning stage – by installing locally or another reflash?
    • WoT device operational stage – remote and dynamic installation (not in current scope, maybe future)
  • i.e. WoT script can be updated on the device
    • ....