draft-moran-suit-manifest-05

Brendan Moran
Overview of manifest problem statement
Problem Statement (1/6)

- IoT devices need to be updated
  - Security patches
  - Functional bugs
  - Feature deployment
  - Time to market
Problem Statement (2/6)

• IoT updates should be easy:
  • An author has firmware
  • A device needs firmware
  • Put the firmware on the device

• IoT updates are not easy because:
  • Many possible actors
  • Diversity of update model
  • Diversity of memory model
  • Multiple functional units
  • Security concerns
Problem Statement (3/6)

• Diversity of update model
  • Active Partition (discard on update)
  • Active Partition (swap)
  • Multi-partition (execute in-place)

• Diversity of memory model
  • XIP
  • Run-from-RAM
  • OS + single application
  • OS + multiple applications
Problem Statement (4/6)

• Update may have many actors with varying concerns and privileges
  • Device OEMs
  • Firmware/Software vendors
  • Device operators
  • Network operators
  • Device Owners
  • Users
Problem Statement (5/6)

• Devices may be composed of multiple functional units:
  • One or more host processors
  • Intelligent I/O
    • Radio modules
    • I/O controllers
  • Intelligent peripherals
    • Sensors with dedicated controllers
    • Actuators with dedicated controllers
Problem Statement (6/6)

• Security Considerations
  • Devices need to make decisions based on trust
  • Trust must be established for any code or configuration either
    • At time of installation
    • At time of use
    • Or, both
  • Multiple actors adds complexity
    • Different trust levels
    • Trust for different operations
  • Multiple functional units adds complexity
    • Actors x Functional units x Operations
Behavioural Manifest
Summary
Observations about updates

• We can’t have many similar formats, there needs to be just one.
• Simple parsers need few unique structures and low nesting levels.
• Update use cases all use the same operations in varying orders.
• An update consumer does not care what an update is, just what it should do.

• Maybe a sequence of update-relevant commands?
Behavioural manifests

Composed of several parts:

• External information:
  • Structure Version
  • Sequence number

• Common information
  • Dependencies
  • Components identifiers
  • Common sequence

• Command Sequences
  • 6 sequences

• Text / text reference

• Layout of the structure:

  `{  
  1 : version  
  2 : sequence number  
  3 : common  
  7 : dependency resolution  
  8 : payload fetch  
  9 : install  
  10 : validate  
  11 : load  
  12 : run  
  13 : text  
  }`
Summary of changes from 04 (1/?)

• Common elements moved into nested in bstr
  • Reduce parsing complexity
  • Improve consistency in format

• Changed encoding of command sequences:
  • Was: [ + { SUIT_Command } ]
  • Now: [ + SUIT_Command ]
    • SUIT_Command is pairs of integer, argument
Summary of changes from 04 (2/?)

• Changed handling of optional sequences:
  • Was: conditional sequences, no explicit structure
  • Now: “try-each” list of conditional sequences. One must pass.

• Added encrypted manifest support
  • Cose Encrypt and detached payload added in outer wrapper as:
    • 3 : COSE_Encrypt
    • 4 : Manifest Ciphertext

• Added “swap” directive

• Defined SUIT-specific digest identifiers

• Editorial changes