Pull Model from the Implementation Standpoint

Jack Visoky and Joakim Wiberg
Rockwell Automation and HMS

October 10, 2018
Pull Model Background

- Pull Model was added to CIP Security (Volume 8) in May of 2018
- Allows for a device to automatically request a certificate
  - Discovers and EST Server using DNS-SD
  - Uses EST to request a certificate
Volume 8 describes precisely how a device should behave to discover and subsequently request a certificate.

Standard technologies are used, mechanisms given to disable this, etc…

However, no information is given on how the user should set up the EST server:
  - Under what conditions should a device be granted a certificate
  - What type of authentications are necessary/possible
  - Are there any common options for this?

As this is so open-ended, it doesn’t make sense to put this information within the CIP Specification:
  - However, it is still likely to be quite useful to users
Various Models for Granting a Certificate

• This paper discusses a few models which are likely to be useful for a large amount of users implementing the Pull Model
• Models discussed here can be combined and modified
  – These are simple examples that seem to be generally applicable
• There is no “one-size-fits-all” solution here
  – This information is meant to provide guidance
  – Combined with a proper threat model a user can apply the appropriate level of security
Vendor Certificate Base Approval

- When a device connects to an EST server, it must do so over a TLS connection
- The device will use its default certificate to do so (as it has not yet been provisioned)
  - In many cases this default certificate is signed by an ODVA member company (referred to as a Vendor Certificate)
- EST server could be set up such that any device which presents a valid Vendor Certificate from a list of trusted vendors is granted a certificate

![Diagram of CIP Security Device, EST Request, EST Server with CA, IF [Signed by Trusted Vendor], THEN Grant Certificate, ELSE Reject]
Administrator Approval Based on Notification

- When a device contacts the EST server the server could pend on approval from an administrator
- The EST server would send a notification (potentially email, SMS, etc…) to the admin and allow for remote approval
  - Included in this could be information about the request, including IP address, serial number, etc…
- Approval here is somewhat manual so scalability would be a concern
  - Also if remote approval is allowed then authenticating that communication is also important
Approval Based on a List of Serial Numbers

- Administrator could pre-configure the EST server with a list of serial numbers of devices in inventory
- Based on the initial request via the default certificate the device could be granted a certificate
  - Default Certificates have Vendor ID and Serial Number, this could be matched to the list
  - Note that security is lessened significantly if this is a self-signed certificate; Vendor Certificate works much better
Username and Password (Software Only)

- EST has a provision to allow for a username and password along with the certificate request.
- For software that is a CIP endpoint, or any device with a human interface, a username and password could be requested.
- Mechanism for entering this information is of course not standardized.
- Need to ensure passwords are managed properly.

Diagram:

- User
- CIP Security Device with Human Interface
- EST Request
  (Username and Password also transmitted according to EST protocol)
- EST Server with CA
Approval via a Provisioning Certificate

- A provisioning certificate could be “pre-loaded” into a device
  - Possibly through a removable media channel utilizing a smart card
    - Protection of private key and certificate is paramount
- Mechanism is not standardized, would necessarily be vendor specific
- Risks around losing control of the removable media with private key and certificate
Global Grant

• You ask for a certificate, you get a certificate!
• Obvious security issues…
  – Could be useful for small systems with a hardened network, or lab environments for testing
    • Just get the system up a running without worrying about security (?)
Conclusions

• Several models explored
• Combinations possible
• Ultimate implementation should be driven by a threat model
  – Tradeoffs can be made between security and usability
  – Other countermeasures (e.g. Intrusion Detection Systems) can compensate for potential security drawbacks of a given model
THANK YOU