Open Trust Protocol (OTrP)

draft-pei-opentrustprotocol-06.txt

Mingliang Pei (*mingliang_pei@symantec.com*)
Hannes Tschofenig (*hannes.tschofenig@arm.com*)
Andrew Atyeo (*andrew.atyeo@intercede.com*)
Nick Cook (*nicholas.cook@arm.com*)
Minho Yoo (*paromix@sola-cia.com*)

IETF 101th, London

OTrP Proposed Design Choices

- **Use asymmetric keys and certificates for device and TAM attestation**
  - Manufacturer-provided keys and trust anchors
  - Enable attestation and establish mutual trust between a TAM and a TEE-device
- **An OTrP Agent in REE relays message exchanges between a TAM and TEE**
- **JSON-based messaging between TAM and TEE**
  - **Other message format: CBOR?**
- **Capable to support different transport**

OTrP Operations and Messages

✓ Remote Device Attestation

Command	Descriptions
GetDeviceState	• Retrieve information of TEE device state including SD and TA associated to a TAM

✓ Security Domain Management

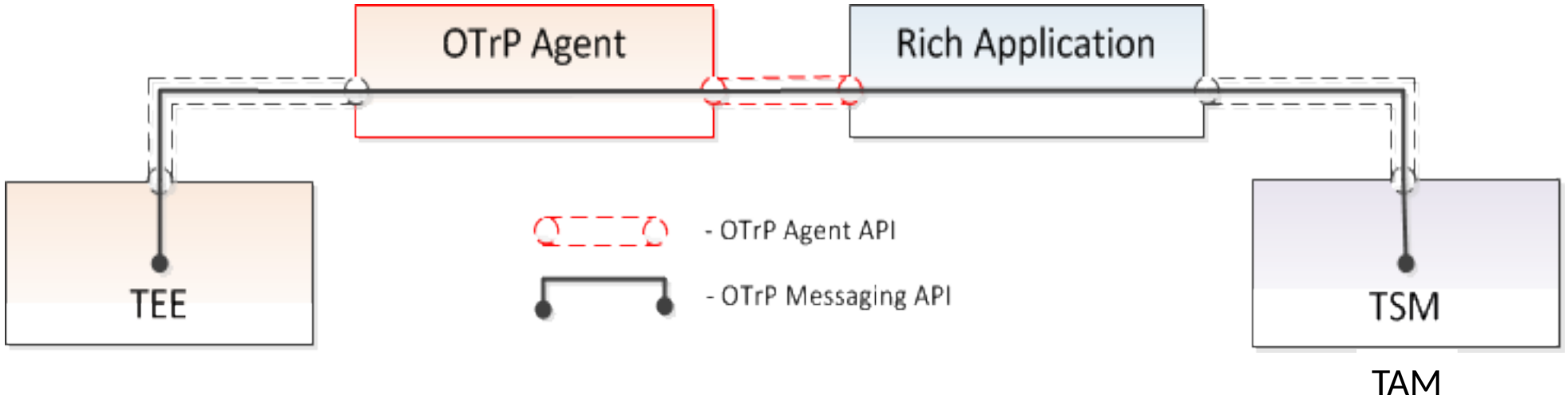
Command	Descriptions
CreateSD	• Create a SD in the TEE associated with a TAM
UpdateSD	• Update a SD or associated SP information
DeleteSD	• Delete a SD or SD related information in the TEE associated with a TAM

✓ Trusted Application Management

Command	Descriptions
InstallTA	• Install a TA in a SD associated with a TAM
UpdateTA	• Update a TA in a SD associated with a TAM
DeleteTA	• Delete a TA in a SD associated with a TAM

OTrP Message Exchange via an OTrP Agent

- An OTrP Agent handles how to interact with a TEE from a REE
- Most commonly developed and distributed by TEE vendor



OTrP Agent Message Relay between TEE and TAM

1. **ProcessOTrPMessage**
   A TEE specific OTrP Agent function that passes OTrP messages between TEE and TAM

   Input:
   An OTrP message from TAM

   Output:
   An OTrP message returned by TEE

2. **GetTAInformation**
   Local query for a TA's information in the device. The response can be verified by a locally stored TEE SP specific anonymous public key.

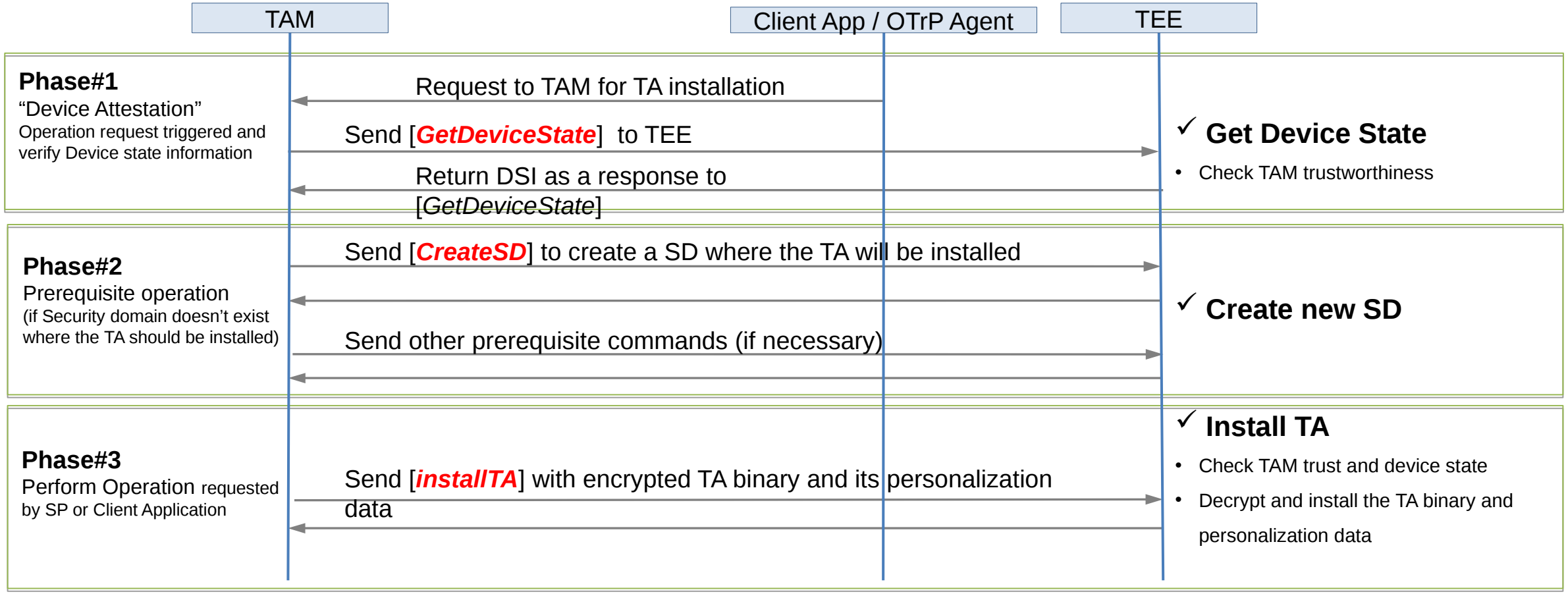
   Input:
   A JSON message with TA identifier, SP Identifer, and a nonce value

   Output:
   An OTrP message received from TEE that describes a TA

Sample Protocol Usage Flow

message sender's Certificate

message sender to check immutability



OTrP JSON Message Format and Convention

```
{
   “<name>[Request | Response]”: {
      "payload": "<payload contents of <name>TBS[Request | Response]>",
      "protected":"<integrity-protected header contents>",
      "header":  <non-integrity-protected header contents>,
      "signature":"<signature contents>"
   }
}
```

For example:

- CreateSDRequest
- CreateSDResponse

Changes from the prior version

- Added transport mandatory support
  - HTTPs as default for now
- Schema small changes to support multiple values
  - GetDeviceStateRequest:
    - Use an array to represent a list of OCSP stapling data (“*ocspdat*”)
    - Use an array to represent a list of support of signing algorithms for algorithm agility instead of comma separate strings (“*supportedsigalgs*”)
  - Use JSON Boolean true | false instead of string “true” and “false”
  - Use “TAM” consistently across the entire document in place of “TSM” (e.g. *tsmid* to *tamid*)
  - Communicated with GP editors (also preferred during discussion with the editors)

Changes from the prior version cont.

- OTrP Agent API changed to be abstract ones
  - Independent of programming languages
- Separated trusted error codes (TEE responded) from the non-trusted error codes (TEE not reachable etc.)
  - E.g. ERR_AGENT_TEE_BUSYERR_AGENT_TEE_FAILERR_AGENT_TEE_UNKNOWN
- Many small editorial updates

Transport Support Consideration

- TEE generally doesn’t have networking capability
- A Rich Application, or Client Application in REE will be doing all networking with TAM
- A Rich App in a device with TEE, which already does PKI cryptography, is most probably capable to do HTTPs, at least on devices with a TEE such as one over TrustZone or SGX today
- Question:
  - Can we start with the protocol with just HTTPs or CoAP must be an mandate for TAM to start with?

Q&A

Thank you!

Message Format Negotiation

- A Client Application may query a device for its preferred message format
- A Client Application triggers TAM to send messages in a preferred format
- Use a default message format