|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **Title\*:** | **End-to-end security use cases and considerations for ETSI M2M Release 2** | | |
| Submitted by: | Francois Ennesser ([francois.ennesser@gemalto.com](mailto:francois.ennesser@gemalto.com)) | | |
|  |  | | |
| from **Source**\*: | Gemalto | | |
| Submitted **To**\***:** | ETSI TC M2M WG4 | | |
|  | | | |
| Relevant WI(s), or deliverable(s): | Release 2 requirements and architecture | | |
| Agenda Item: |  | | |
| Submission date**\***: | 2012-08-10 | | |
|  | | | |
| **Document for\*:** | Decision |  | 🡨 a decision on is formally requested from the addressed (sub-)committee |
| Only one "**X**" | Discussion | **X** | 🡨 the contribution is expected to be presented and discussed, but no decision is formally requested |
|  | Information |  | 🡨 the contribution does not require discussion |
|  | | | |

**Decision/action requested** (Mandatory if Decision box is checked, optional otherwise)

*The present contribution provides use cases where M2M applications require End-to-end security and where the bootstrapping of security credentials causes difficulties that could be solved if the TC M2M Service bootstrapping architecture was extended to enable M2M applications to leverage on such capabilities (as proposed in M2MWG4(12)000007 / M2M(12)21\_018).*

**Background information:**

Discussions at the August 7 TC M2M WG4 teleconference around draft CR to TS 102 690 in M2MWG4(12)00007 concluded that related use cases would be useful to refine the technical proposal to bootstrap security credentials to be used by M2M applications for End-to-End security, and to assess the suitability of proposed solutions. For this purpose, the present contribution describes use cases from different verticals and analyzes their implications in terms of requirements for End-to-end security.

**Use Case 1: Smart Metering – Privacy preservation and critical infrastructure protection**

This use cases typically involves:

1. A utility distribution company that operates distribution services for a given commodity (water, gas, heat or electricity) for a set of consumers (e.g. a town or region): “Distributor”. A consumer with a smart meter may switch between distributors, e.g. to benefit from better tariffs.
2. An end consumer of the commodity (individual household or business, e.g. a hospital): “Consumer”.
3. An Access Network operator

Depending on national regulations, the Smart Meter measuring the flow of commodity may belong to the distributor or the consumer. The distributor typically operates local gateways that relay the communications between the smart meters connected to it via a local communication network (e.g. Power Line Communication) and the Metering Head-end system in charge of controlling the infrastructure. Communication between Gateways and the Head-end system may go through an Access Network operated by an access network operator. The Head End System may be implemented as an ETSI M2M compliant “NA” to rely on ETSI M2M compliant services offered by an M2M Service Provider. That is, the distributor takes a role of an M2M Application Service Provider in the ETSI M2M ecosystem.

**Data collected by smart meters are considered as sensitive personal data of the end consumers by the European Commission,** because access to such data enables tracking of consumer’s activities, (e.g. appliances she is using) in the intimacy of his/her home. As such, access to this data shall be strictly restricted to actors with legitimate purposes, such as billing (in case of the Distributor) and grid optimization (in case of an energy transmission company), or providing enhanced services regarding consumption optimization, if explicitly requested by the consumer. There are also **national regulations, preventing the exposure of such data to unauthorized third parties**.

Even though most access networks provide encryption over certain hops, the access network operators can access the encrypted data at the end of their pipe, since they control the encryption credentials. Some utilities are reluctant to rely on third party data communication services in their Advanced Metering Infrastructures deployments due to this concern.

The current ETSI M2M platform enables secure data transmission at service layer from the Gateway GSCL to the NSCL, based on credentials bootstrapped by the M2M Service Provider. Providing security at service layer removes the concerns regarding exposure of sensitive data to Access Network Operator, but in turn brings concerns regarding exposure to the M2M Service Provider (who now have access to service layer encryption credentials). Thus the **current M2M bootstrapping infrastructure (adding encryption at service layer in addition to encryption at transport layer) simply adds to the complexity without adequately addressing the** **privacy concerns of such M2M applications**.

* The privacy issue could be avoided if an authorized actor, such as an energy distributor, would act as an M2M Service Provider, but in this case the main advantage of the ETSI M2M approach, i.e. **horizontality, would be lost**!
* However the technical solution specified today for bootstrapping in the ETSI M2M **specification could be extended** to resolve the above problem, for example by **enabling the credential distribution (for application security) to be performed by a Third Party** , possibly independent from the M2M Service Provider. Though some M2M application owners may have the means to deploy their own proprietary end-to-end bootstrapping solution, an interoperable infrastructure operated by a trusted third party, dividing the cost among multiple users, is needed to avoid entry barriers against small-scale deployments. Therefore it makes sense to consider such solutions in the context of M2M Service platforms.

Beyond the privacy aspect, Smart meters create a multitude of **entry points to the Information System of a critical infrastructure.** For example at least in some countries, smart meters will be equipped with a remotely actionable switch enabling to remotely order connection/disconnection to the commodity, e.g. in case of unpaid bills. This could result in huge liability as attacks could result in unwanted power disruption or surges to strategic points within the grid as critical times, which can motivate parties with malicious economical or political motives. With the current M2M infrastructure, **liability may fall on the M2M Service Provider**, e.g. in case of insider’s involvement. Though the role of the M2M service Provider may typically be assumed by a telecommunication operator, as it mainly involves information routing capabilities, in general it is difficult to divide the financial liability when multiple parties are involved in a breach. This gives an additional reason for the specification to **enable the separation of the roles related to information handling (M2M SP) and trust Provisioning** (Credential Service Provider in the current WG4 terminology).

Madjid>>We need to note that a credential service provider alone cannot bear all the liability. When a trust ecosystem is established, both the trust provider and the application owners have a set of well-defined responsibilities. For instance, the trust provider is responsible to keep its facilities and servers secure, while the application owners are responsible to keep the received credentials safe, which in term means they will hold device vendors accountable through robustness requirements. I don’t think the reasoning above dictates a solution based on third party trust providers, unless multiple application owners have to use the same set of credentials. On the other hand, it does provides a reason for keeping the credential provisioning for applications independent from underlying transport and service layer.

**Use Case 2: Scenarios involving multiple Service Providers, e.g. Smart Cities emergency**

Another need acknowledged in TC M2M Requirements (TS 102 689) but not addressed by the ETSI M2M Release 1 security architecture is to **secure communications between M2M devices affiliated with different M2M Service providers**. Such scenario could become frequent in future Smart Cities environments. An example is the case of vehicular accidents where personal devices with M2M subscriptions contracted by individuals (e.g. patient healthcare monitoring devices) or by enterprises (e.g. fire alarms monitored by insurance companies) will interact both with institutions (e.g. emergency vehicle fleet services)or private organizations applications (e.g. Medical centres/doctors) and municipal services using public network infrastructures (e.g. traffic flow optimization) .

Here again, the technical requirements are placed on devices to bootstrap credentials enabling end-to-end application level security independent of their M2M service providers.

Madjid>>I think this case is a much better motivation for a third party trust provider (using PKI), because now you have different applications deployed by very different organizations and if each organization deployed its own proprietary system, it would be very difficult to get them working. Still, in practice, it will be very difficult to get all these parties to adopt a single third party trust provider anyway.

**Use Case 3: Consumer networking applications, e.g. Digital picture display frame**

Most M2M applications today assume a client-server relationship and therefore work well in the context of a single M2M SP. But by the time the standards are deployed, an evolution towards **“many-to-many”** relationships is likely to happen. For example in the industrial sector data from a sensor deployed by one entity may be used by several other entities for their own applications. Providing interoperable security mechanisms at application level will enable such inter-application use cases.

The same evolution can be expected in the consumer segment with the advent of M2M devices in social networking. For example a digital picture display frame in the grandmother’s living room would subscribe to retrieve pictures from her daughters cameras to show her grandchildren.

While **the subscribe-notify mechanism in the M2M architecture has been designed to handle such use cases**, the **security framework definitely needs to be extended** to address the security and privacy preservation challenges resulting from such evolution.

**Service level considerations:**

This paragraph aims at identifying the various technical pieces that will need to be specified to complete the end-to-end security part of the ETSI M2M system.

We start from the assumption that the M2M platform should offer a number of services to help setting up end-to-end data security, as follows:

**Policy enforcement services**: The M2M platform should handle requests for M2M data distribution (request for data, requests for data notification...). With the current architecture where data routing and trust function are offered by the same provider, the policy decision point and the policy enforcement point can both be collocated within the platform. In other words, the access rights are managed within the platform with the current architecture.

However, if we suppose that the trust functions are offered by an external provider, then the platform will need to enforce the data access and dissemination policies defined by an external entity. Services should be defined for this purpose.

Madjid>> the policy regarding data for each application still comes from the application owner+user. We cannot put third parties in charge of the policies..

**Discovery services**: In order to understand the data transmitted from an M2M devices, recipients will need to obtain the proper credentials. To do so they need to discover and authenticate with the proper credential distribution authority. When recipients are affiliated to an M2M service provider, the M2M platform could offer discovery services to help discovering and authenticating with the credential distribution authority.

To enable the above services, the following Stage 3 aspects will have to be addressed:

**Protocol definition:** Definition of the protocols to be used to obtain, renew and revoke credentials used for end-to-end data protection. A number of such protocols already exist. The idea is not to reinvent them, but rather to define which existing protocols could be used.

**Inter M2M service provider communication protocols:** End-to-end data security should be applicable between parties affiliated to different services providers. This is already the object of another work item for Release 2, which should define mechanisms that would allow M2M service providers to channel requests from affiliated devices to other M2M service providers.

**Further technical considerations:**

As the purpose of the present document is to reassess WG2 Release 2 priorities and refine their scope in relation with expected workload, the following technical aspects also deserve considerations:

1. **Multitude of bootstrapping options:**

As there are many bootstrapping options in M2M Release 1 specifications, the complexity involved in bringing a complete End-to-end security solution is multiplied. Feasibility would be eased if we can agree to extend only one or two options for Release 2, such as GBA for an access network based solution and EAP-TLS for an access network independent solution. Insight on which options are most likely to be implemented on the field would be helpful in that respect.

Madjid>>GBA credentials would not be independent of transport, aren’t application keys derived from IK, CK, which is from network operator? Same thing about EAP-TLS. If EAP-TLS is performed with a network operator AAA, then the operator has access to EMSK and any key generated from EMSK is in the hands of the operator.

1. **How to extend end-to-end security from devices to applications:**

The considerations above would enable to provide end-to-end security between communicating D/G M2M Nodes. But End-to-end security is needed at the application level, which further involves the dIa reference point.

* In the case of a **D device holding several independent M2M DAs**, we need to ensure cryptographic separation between them, but the security of dIa could be left to implementation as these are internal interfaces. However we still need to consider how to move the credential from the D M2M node to the DA, e.g. should the D M2M Node perform an independent bootstrap procedure for each application, or could we diversify application keys from a unique device credential? Madjid>> Ok, so we are saying (correctly) that each DA uses its own separate credentials, but are we saying that the credentials are generated in D M2M? Why do they move from D M2M node to DA? Does the DA run within a secure environment or does it pass application data for security functions into the D M2M Node Secure environment (through an API, the way most secure environments work)?
* In the case of a **Gateway bridging an M2M Area Network made of individual D’ devices**, the further difficulty is that the security over dIa cannot be ignored. While service layer security over dIa could in some cases rely on features of the M2M Area Network technology, document M2MWG4(12)000008 highlights the interest for an M2M service provider to extend its security framework through the Gateway to the locally connected D’ devices, especially in the case of meshed networks such as Zigbee.
* In such deployments, the Release 1 assumption that Gateways would be application-specific deserves to be revisited, because of a cost incentive to rely on **public or shared gateways** (not application-specific): This has further security implications which would need to be addressed.