IoT Standard Diagnostic Logging

Version 1.0

04 January 2018

This is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Antitrust Notice

Table of Contents

1 Introduction

1.1 Overview

The purpose of this document is to provide a standardized method to log modem data and messaging on an Internet of Things (IoT) device (Long Term Evolution (4G), category M1 (LTE-M1) and Long Term Evolution (4G), category NB1 (LTE-NB1)), eliminating the need for tethered logging. The primary user of the logging tool is expected to be mobile network operators.

Possible use cases are listed below:

- Report Radio Frequency (RF) parameters (Reference Signal Received Power (RSRP), Received Signal Strength Indicator (RSSI), Signal to Noise Ratio (SINR) etc.) to determine network coverage
- Capture handover statistics to debug handover issues
- Report Voice over LTE (VoLTE) call statistics (e.g. Delay, Jitter and Packet Loss) to aid in VoLTE analysis
- Real time reporting on the device
    - Txpower
    - Cell selection
    - RF parameters

The operator will be able to log with any device/chipset compliant to the interface outlined in this document by downloading and integrating the Lightweight Machine to Machine (LwM2M) client application on the device. The LwM2M client should be downloaded and installed from a secure location on the operators' server and the authenticity of the client should be verified by mutual authentication before the client is installed on any device. The logs will be sent up from the device to the logging server using LwM2M Application Programming Interface (API) calls. This document provides the APIs and logging Key Performance Indicator (KPIs) to capture the modem and other components' log data; and the security protocol to authenticate the device before logging can be initiated. The API interface code that contains the function declarations is available on the GSMA GitHub site, the manufacturers and application vendors are encouraged to download this source code to implement the standard logging solution.

The APIs defined in this document to retrieve the modem and other components' log data are part of the LwM2M architecture adapted to collect IoT modem logging KPIs. Using LwM2M, the diagnostic application will be able to retrieve information from the components being logged, such as a KPI or a protocol message. As per LwM2M convention, all information retrieved from the components being logged, is passed as objects/resources and are defined in the interface header file. Some key advantages of using this framework is a simple, low-cost method that embraces modern architectural principles and works over wireless connections to define the interface to the wireless modem. It is an industry standard method that is meant for memory and processing power constraint devices. The standard transport mechanisms and security architecture defined in OMA TS LwM2M is used instead of proprietary methods. The transport and security mechanisms used in implementing the IoT Standard Diagnostic Logging (SDL) is defined in section 5 of this document.

This document is organized as follows:

Section 2 introduces LwM2M and presents an overview of how it is used to collect modem logging KPIs from IoT devices for LTE-M1/LTE-NB1. This overview includes illustrations of interfaces between the device and the LwM2M client framework.

Section 3 presents architectures on the device to implement the LwM2M interface.

Section 4 describes the adaptation of the LwM2M interface to collect LTE-M1/LTE-NB1 KPI logging. The messaging interface is defined in this section.

Section 5 outlines the security architecture for collecting KPI logging and transporting it between the client and server. The authentication mechanism and remote control session is also discussed in this section.

Section 6 describes the implementation details to enable KPI logging on a LTE-M1 or LTE-NB1 device. The logging objects and structures are defined in this section..

Annex A References the source code location.

Annex B Document History.

1.2 Scope

The initial scope of the GSMA Standard Diagnostic Logging interface (version 1.0) is restricted to LTE-M1 and LTE-NB1. Future expansions of the scope to other technologies requires further study.

1.3 Definitions

Term	Description
Callback Function	A callback function is a function that is passed to another function as a parameter, and the callback function is called (or executed) inside the other function
Client	LwM2M Client Application
Event	A Log Record pushed by LwM2M client to a LwM2M server
KPI	Key Performance Indicator - KPI Log Records report KPIs
Log Record	A single piece of diagnostic information from device,
Message Queue	Message queue is a queue of messages sent between applications. A message is data transported between the sender and the receiver application, it's essentially a byte array with some headers on top. This is different from Queue Model of LwM2M.
Protocol Message	A Log Record containing an OTA message of a particular protocol
Registration	Register the Client and its Objects
Server	LwM2M Server

1.4 Abbreviations

Term	Description
API	Application Programming Interface
CoAP	Constraint Application Protocol
DTLS	Datagram Transport Layer Security
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Task Force
IoT	Internet of Things
IPC	Inter-Process Communication
IMS	IP Multimedia Subsystem
IMEI	International Mobile Equipment Identity
KPI	Key Performance Indicator
LDI	Lightweight Diagnostic Interface
LTE-M1	LTE Cat-M1 or Long Term Evolution (4G), category M1
LTE-NB1	LTE Cat-NB1 or Long Term Evolution (4G), category NB1
LwM2M	Lightweight Machine to Machine
NAS	Non-Access Stratum
OEM	Original Equipment Manufacturer
OID	Object ID
PDU	Protocol Data Unit
PRD	Permanent Reference Document
PUSCH	Physical Uplink Shared Channel
REST	Representational State Transfer
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RSSI	Received Signal Strength Indicator
RTCP	RTP Control Protocol
RTP	Real Time Transfer Protocol
RRC	Radio Resource Control
SINR	Signal to Noise Ratio
SMS	Short Messaging Service
SW	Software
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TLV	Type Length Value
UDP	User Datagram Protocol
VoLTE	Voice over LTE

1.5 References

Several standards were used to develop IoT logging. They are listed here as a reference.

References	
LwM2M	
LwM2M standards were used to define the objects and messaging format.	
Lightweight Machine to Machine Architecture	OMA-AD-LightweightM2M-V1_0-20131210-C.pdf
Lightweight Machine to Machine Requirements	OMA-RD-LightweightM2M-V1_0-20131210-C.pdf
OMA Lightweight Machine to Machine Technical Specification	OMA-TS-LightweightM2M-V1_0-20170208-A.pdf
3GPP	
All logged messages should be reported in their original format without modification, as described by the 3GPP standard.	
RRC	http://www.3gpp.org/DynaReport/36331.htm
NAS	http://www.3gpp.org/DynaReport/24301.htm
MAC	http://www.3gpp.org/DynaReport/36321.htm
PDCP	http://www.3gpp.org/DynaReport/36323.htm
RLC	http://www.3gpp.org/DynaReport/36322.htm
IP	
IP Packets (and all protocol messages contained therein) logged should also be in their original format without modification, as described by IETF.	
IPv4	http://www.ietf.org/rfc/rfc791.txt
IPv6	http://www.ietf.org/rfc/rfc2460.txt

2 LwM2M Logging Interface Overview

The standard logging interface is referred to in this document as the Lightweight Diagnostic Interface (LDI), which is an API and a messaging interface between a LwM2M client application running on a mobile device and the device component being logged. The modem is the main component that is being logged. LDI enables applications to monitor and control the activities of the component being logged. The messaging interface is based on a modified version of the LwM2M client with interfaces suited to logging LTE-M1 and LTE-NB1 devices. This scope of this interface can be expanded in the future to log more protocols used in IoT devices. The LwM2M Client Application allows the use of the following LwM2M features that facilitates the collection and transmittal of logging KPIs from the device to the operator server:

- Register devices.

- Securely communicate with Server over the DTLS protocol.
- Fully control the logging application logic.

Table 1 outlines an example use case of primary functions of this interface, and how LwM2M is used to achieve this function.

Function	Example	Mechanism
Interrogate	What is the device's current RSRP?	LwM2M "Read" operation
KPI Subscribe	Subscribe to certain groups of KPIs	Subscribe using LwM2M "Observe" operation Receive Information using LwM2M "Notification" operation
KPI Unsubscribe	Unsubscribe to certain group of KPIs	Subscription can be cancelled using LwM2M "Cancel Observation" operation

Table 1 Primary Functionality

The logging information that can be retrieved from a device through LwM2M can be described as objects and resources in LwM2M terminology. Types of log records are listed in Table 2.

Type	Examples	Pushed/Pulled	Format Definition
KPI Log Record	• RSRP • PUSCH Tx Power • Path Loss	Push or Pull	Defined as LwM2M object/resource
Configuration Log Record	• Device Name • IMSI	Pull	Defined as LwM2M object/resource

Table 2 Log Record Types

As illustrated in Figure 1 below, the mobile IoT device Original Equipment Manufacturers (OEMs) are expected to implement an interface that provides diagnostic logging for both LTE-M1 and LTE-NB1 devices as specified in this document. Access to the interface may be restricted to one application which is approved by the mobile network operator. Since it is a security risk, and to therefore to block hackers, the operating system will not allow other applications to have access to the interface. There are some additional restrictions imposed by LwM2M described in section 5 of this PRD, which makes the logging solution more secure and robust. The application with access to the interface should be fully secure as determined by the mobile network operator and should have all authentication, data integrity and security features as described in Section 5 of this document.

LwM2M client applications utilize the programming interface APIs to monitor and control the device being logged. As an example, when invoked by an LwM2M client application using the messaging interface between the client and the device, the device will fetch the logging KPI and can communicate via Inter-Process Communication (IPC) to perform the monitoring and control functions requested by the LwM2M client application. LwM2M is defined in a generic way with the goal that the interface between the device and the client is implemented by an OEM, regardless of the device operating system or the device chipset.

2.1 LwM2M Architecture Supporting Device Logging

Figure 1 below illustrates a possible modem chipset architecture for IoT supporting either LTE-M1 or LTE-NB1. All logging KPIs supported by this interface are defined in a header file that need~~s~~ to be included by the device.

LwM2M client applications will develop the messaging Interface to link to the LwM2M client to the device. Standard messaging between client and device is established using either message queues or call back functions. The message queues referred to here are the operating system message queues, and is different from the queuing model of the LwM2M protocol. The device communicates the LTE-M1/LTE-NB1 modem logging KPIs to the LwM2M client over these standard messaging interfaces. The client is responsible for sending the logged KPIs to the operator server where it is written to filesystem maintained by the mobile network operator. The format and structure of the filesystem is defined by the network operator. This includes the filesystem location on server, security, and access privileges. The LwM2M client resides on either the Baseband Processor or the Application Processor on the device as defined in Section 3 of this Permanent Reference Document (PRD).

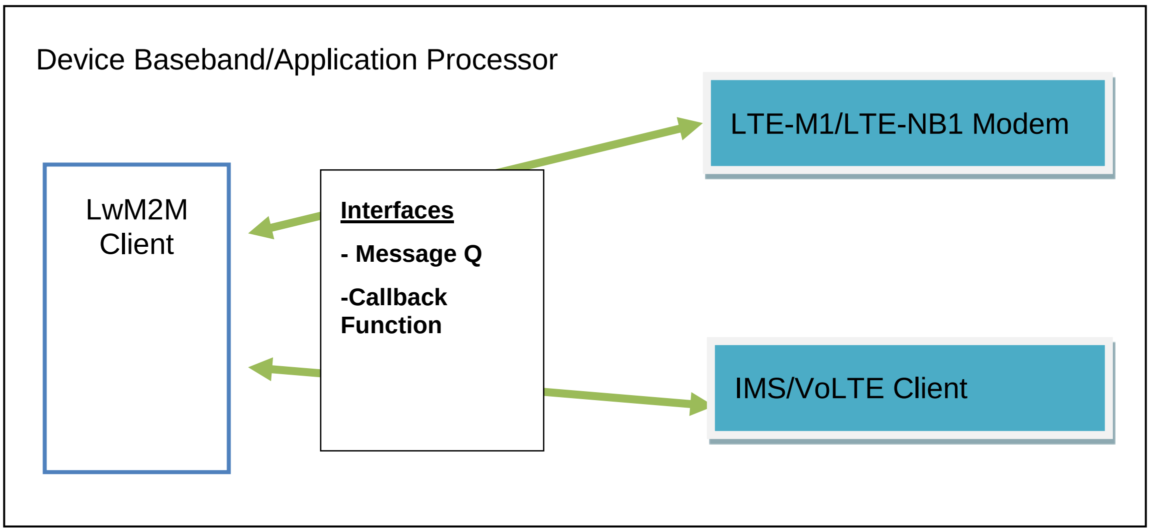

Figure 1 Example of IoT Chipset Architecture with Logging

Error! Reference source not found. illustrates the generic LwM2M client-server architecture that uses the Constraint Application Protocol (CoAP) with User Datagram Protocol (UDP) and Short Messaging Service (SMS) bindings. Datagram Transport Layer Security (DTLS) provides security for User Datagram Protocol (UDP) transport layer. Figure 2 in this PRD is a compilation of Figures 1 and 2 from OMA-TS-LightweightM2M technical specification.

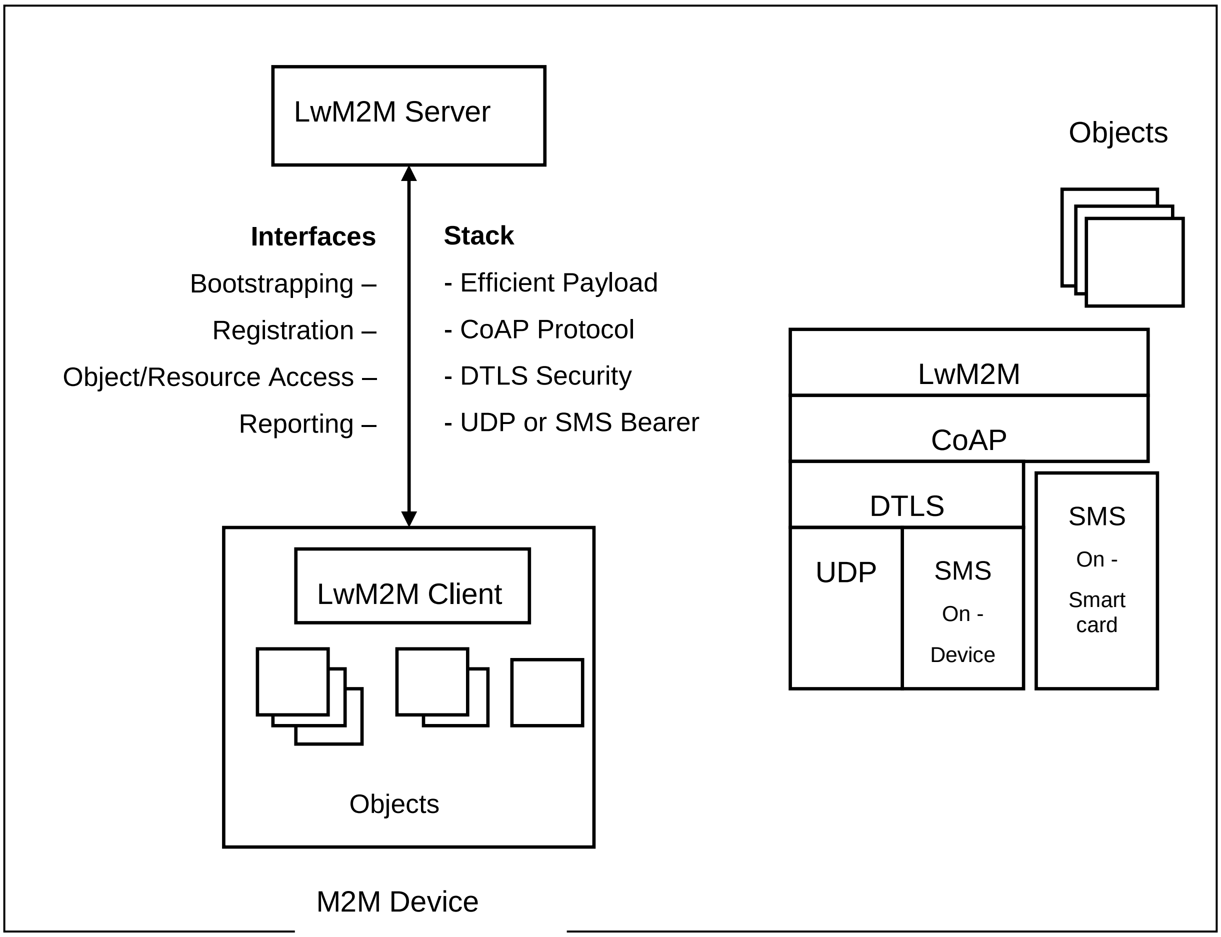

Figure 2 Generic LwM2M Server and Client Architecture

The device communicates the LTE-M1/LTE-NB1 logging KPI to the LwM2M client whenever it is generated. The client is responsible for communicating the KPIs back to the server based on the rules and masks defined. The server can subscribe to a subset of logging objects or can request a one-off KPI at any point of time.

3 Cross Platform Compatibility

LwM2M client for KPI logging is defined with the goal that it can be implemented by one or more chip vendors, regardless of the device operating system or the device chipset. Consequently, a LwM2M client application should run on any such device without modification. The LwM2M client can either run on the Application Processor (AP) or the Baseband Processor (BP) of the IoT device. Depending on the logging interface on a particular chipset, the LwM2M client may reside either on the AP or BP of the device. The architecture is completely flexible and allows this possibility with no caveats.

This library has been designed to make integrating an LwM2M client onto devices as seamless as possible. Interfaces between the device and the LwM2M client have been defined using callback function calls and message queues. The interface between the client and the server follows the standard OMA LwM2M. To the extent possible, the usage should be identical across devices, chipsets, OEMs and even operating systems. However, due to underlying differences between current IoT mobile operating systems, some differences will be inevitable.

Additional logging KPIs and new protocol messages as well as new protocols can be added to the logging set by defining new LwM2M objects. LwM2M is merely used to securely gather and transport the KPIs defined. The support for implementing the logging for a specific KPI needs to be supported on the device.

4 Use of LwM2M for Modem KPI Logging

The logging interface is based on LwM2M. The reason for basing the interface on LwM2M is to make use of a standardized approach. LwM2M is a communication standard defined by OMA to cater to the needs of constraint devices: with limited network bandwidth, computing power and memory, and depending on a limited battery lifetime and at very low production costs. It is a simple, low-cost remote management and service enablement mechanism, which embraces modern architectural principles (in line with Internet standards), also works over wireless connections and is fit for purpose due to being lightweight.

The LwM2M protocol, has at least four outstanding characteristics:

1. It features a modern architectural design based on REST appealing to software developers,
2. It defines a resource and data model that is extensible,
3. It has been designed with performance and the constraints of M2M devices in mind,
4. It reuses and builds on an efficient secure data transfer standard called the CoAP that has been standardized by the Internet Engineering Task Force (IETF) as a variation of the Internet’s Hypertext Transfer Protocol (HTTP) protocol (appropriate for data transfer to and from low-cost connected IoT devices).

All logging KPIs are defined in terms of LwM2M objects and resources. LwM2M defines a simple resource model where each piece of logging information made available by the LwM2M client is a resource. Resources are logically organized into Objects. An object must be instantiated first by either the LwM2M client or server, which is called the object instance, before using the functionality of the object. After an object instance is created, the LwM2M server can access that object instance and resources that belong to that object instance. Logging KPIs are organized in the form of LwM2M objects, and each object is identified by a unique object ID. Each resource within an object is identified by using a unique resource ID. The resources are defined to be either integer, Boolean, Opaque or String type. LwM2M also provides for Time and Float data types, but these are not used in the current version of the SDL logging implementation. Time and Float data types can be used in future SDL implementations. Most resources are available to be read, and some can be both read and written. For logging functionality, resources with read and write operations are primarily used, and some resources with execute operation are also used. The execute operation is used by the LwM2M server to instantiate some action, and can only be performed on individual resources. The execute operation is used on resources that are used to execute commands on the device similar to AT commands. A resource may consist of multiple instances called a resource instance as defined in the object specification. The scope of these objects covers a set of KPIs, as well as some basic configuration and commands. The design is extensible so that more KPIs and commands can be added in the future.

4.1 Method of Message Exchange

There are several data exchange mechanisms offered by LwM2M protocol to transport logging payload between client and server.

Client supports several data transfer formats:

- Plain text (for Single Resource)
- OMA TLV (for Multiple Resources, Object and Object Instances). (application/vnd.oma.lwm2M+tlv)
- OMA JSON (application/vnd.oma.lwm2M+json)
- and Opaque (sequence of bytes) data format

For the purpose of logging, the OMA TLV format is preferred over the plain text data format. Moreover the OMA TLV data format is the only data format which is mandatory to be supported by any LwM2M Client). The server can either pull logs from the client by using the LwM2M "Read" operation, or the server can request that logs to be pushed to it by specifying which events should be sent from the client to the server. When the read operation is used, the latest value of the logging KPI represented by the LwM2M resource or object is sent from the Client to the Server. When Server starts an observe request for a resource to Client, the client notifies the updated value of resource to server, each time it receives the same from the device. An observation ends when a “Cancel Observation” operation is performed. The LwM2M server can initiate an observation request for changes value updates to a specific resource, resources within an object instance or for all the object instances of an object within the LwM2M client. The “Notify” operation is sent from the LwM2M client to the LwM2M server during a valid observation on an object instance or resource. This operation includes the new value of the object instance or resource. Observation from the server side can also be initiated with various conditions (i.e., minimum period, maximum period, greater than, less than, step). The Notify operation is only sent from the client when all the conditions configured for the Observe operation are met. The cancel observation operation is sent from the LwM2M server to the client to end an observation relationship for object instance or resource.

Periodic and event triggered reporting about resource values from the LwM2M client to the LwM2M server is achieved through CoAP Observation. This simple mechanism allows the LwM2M server to send a “GET” request. The LwM2M server can cancel the “Observe” operation by sending Reset message as the response for Notify message in which the LwM2M server is not interested any more. When the LwM2M client receives a Reset in response of a “Notify” operation, the LwM2M client must cancel the Observation. Alternatively, an Observation can be cancelled at any moment in sending an Observation command with the Cancel option. Below messaging diagram shows an example of an information reporting exchange.

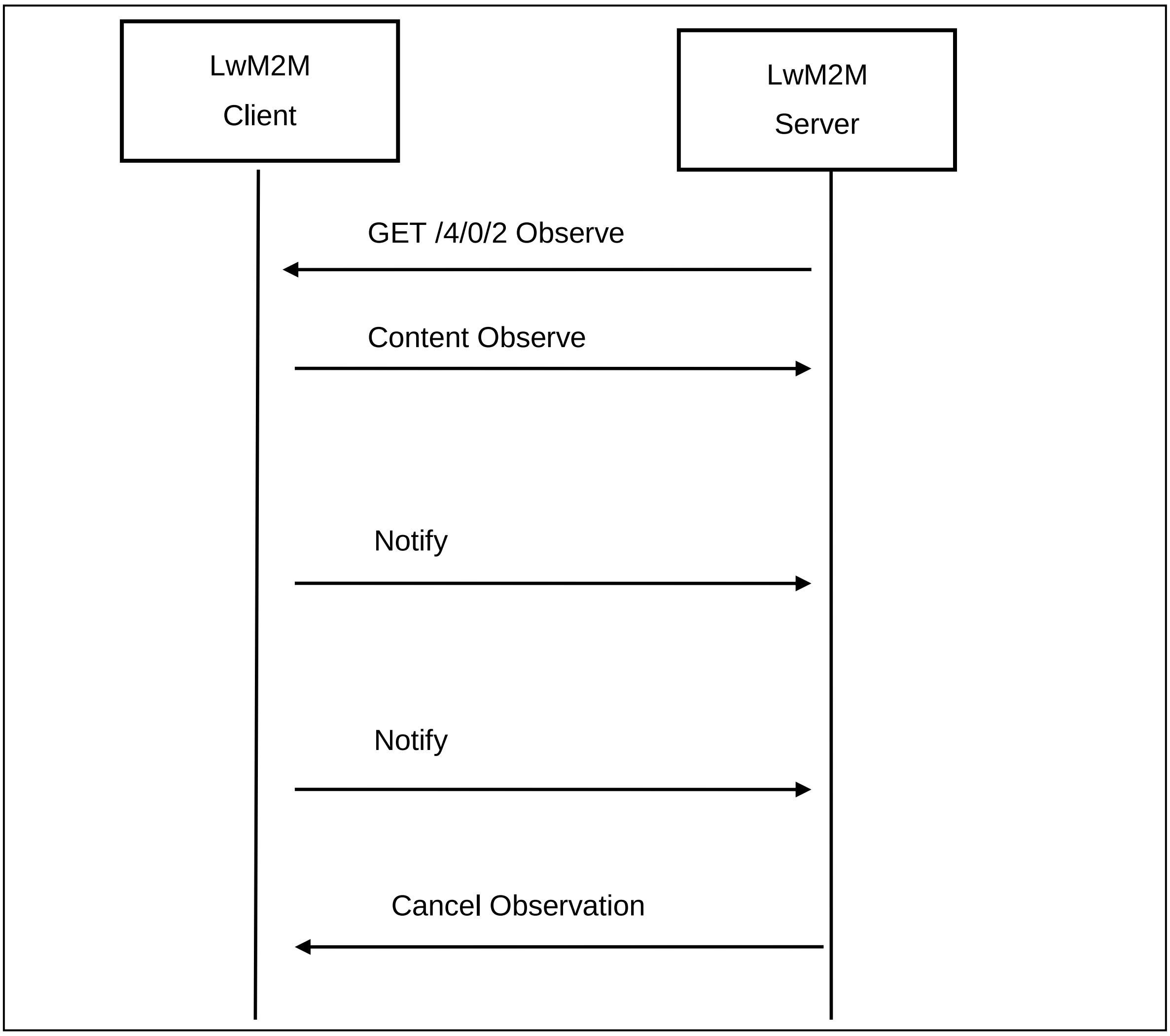

Figure 3 Information Reporting Exchange between LwM2M Client and Server

5 Security

This section defines the security architecture design for logging using LwM2M. The IoT SDL implements security for each endpoint as well as for the transportation of data over a secure transport channel. For authentication of communicating LwM2M entities, the LwM2M protocol requires that all communication between LwM2M client and server are authenticated using mutual authentication. Access to the device interface is restricted to only one LwM2M client which is approved by the mobile network operator. This prevents any malicious application from accessing the device interface. If any third party LwM2M client is developed, it should have the same end-to-end security features as an LwM2M client from the mobile network operator.

The end points of the IoT SDL implementation are the device and the operator server. The security of the data collected by the logging mechanisms is maintained by storing the payload of the logging data in a secure area of memory that is encrypted. Policies shall be set to clear this area of memory periodically. The LwM2M client data structures are defined to be stored in a secure section of memory and will be cleared as soon as the log data payload is communicated to the server over a secure channel. The log data is transmitted over the air on a best effort basis and there is no retransmission of data. This policy makes the storage of data for the least amount of time, to minimize the access of this data by a malicious application. On the server side, the mobile network operator is responsible for the security of the logging data that is collected from IoT devices. The data store on the server side needs to be encrypted and securely stored to prevent malicious applications from accessing this data stored on the server. The logging application is based on the native security mechanism offered by the LwM2M protocol. The LwM2M protocol is based on CoAP principles and utilities UDP and SMS transport channel bindings of the protocol. The LwM2M protocol utilizes the security mechanisms of these channel bindings to implement authentication, confidentiality, and data integrity features of the protocol between LwM2M entities. The next version of OMA TS LwM2M adds end to end integrity protection and security of the payload data and will be included in the future version of the IoT SDL logging application.

The UDP channel security for COAP is defined by the DTLS, which is equivalent of TLS v1.2 for HTTP and utilizes a subset of the cipher suites defined in TLS. All transmissions to / from the server shall use DTLS to ensure security of the user authentication information and log data while in transit.

The LwM2M server must support the Pre-Shared Key mode of DTLS with cipher suites. A LwM2M client must support the Pre-Shared key mode with at least one of the cipher suites specified for the LwM2M server. For all cipher suites using AES in an LwM2M implementation, the hashing functions should be SHA256.

LwM2M also supports certificate-based method of security. For the logging application, the LwM2M client must have a copy of the operator certificate stored and must check that the certificate presented by the LwM2M server matches this stored certificate.

The operator owned server manages both authentication and log sessions in devices. Selective logging can be initiated on the server side by sending a LwM2M Observe/Notify request on selected objects or resources. The client will only send notifications of the logging KPIs that are being observed. The LwM2M client runs as an application on the device. Metrics and KPIs are sent from the device to the LwM2M client using either message queues or callback functions. The LwM2M client in turn sends these metrics and KPIs over to the LwM2M server that is running on the operator’s server.

The log mask limits the scope of the logging information. The log mask can be set by the LwM2M server or privileges enforced by the Software load. The log mask setting can be managed by an application configuration/consent where a user is provided an option/interface to select what data can be logged. As a second option, the log mask can also be enforced with the software load on the device. For example, the operator field engineers may receive a software load where all log masks are available, but if 3rd party vendors are testing for the operator, they may get a software load where certain log masks cannot be enabled.

6 IoT Logging Implementation Using LwM2M

Logging IoT devices is modelled using LwM2M. The LwM2M client can either run on the Application Processor (AP) or the Baseband Processor (BP) of the IoT device. Depending on the logging interface on a particular chipset, the LwM2M client may reside either on the AP or BP of the device. The architecture is completely flexible and allows this possibility with no caveats. The LwM2M client needs to communicate with the modem that is responsible for sending logging KPIs whenever they are generated. The device OEM vendor and the baseband chipset vendor need to provide the interfaces to communicate between the logging interfaces on the device and the LwM2M client.

The OMA LwM2M has recently worked in the direction of the Log Event functionality to harmonize a capability usage across different implementations. OMA has defined a set of re-usable resources dedicated to logging which can be included in user-defined logging objects to contribute to the specific needs. Also, OMA is currently defining an object for the purpose of generic logging, using these reusable resources. This capability needs to be further explored and can be added as future enhancements to IoT SDL implementation.

6.1 LwM2M Logging Object

- The LwM2M logging objects follow the hierarchical structuring rules defined in LwM2M
- All LwM2M resources in an object correspond to data types allowed by LwM2M – namely Boolean, integer, opaque and string type. Time and Float data types are not used in the current version of the implementation.

Below is an example of the “plmnSearchEvent” logging KPI defined in terms of a LwM2M object. All the resources within that object are also depicted in the table below.

Object Definition:						
Name	Object ID	Instances	Mandatory	Object URN		
plmnSearchEvent	3352	Multiple	Mandatory	urn:oma:lwm2m:ext:3352		
Resource definitions:						

Object Instance ID	Name	Resource ID	Default value	Type	Operations	Description
0	timeScanStart	0		Integer	R	Start time
0	Plmn	1		Integer	R	PLMN - mcc/mnc
0	BandIndicator	2		Integer	R	Band Indicator
0	dlEARFCN	3		Integer	R	EARFCN - frequency

6.2 LwM2M Logging Record Structure

The interface between the device and the LwM2M client follows a simplified implementation where two methods are provided for the device to communicate with the LwM2M client – using message queues and callback functions. The client will update its in-memory data structures based on the log data received from the device. The interface between the device and the client defines each resource defined as a structure called Type Length Value (TLV) with two fields:

- Resource Number
- Value

The device is required to create a logging object for the KPI being logged, define each resource in the object and populate it with the actual logging data. This object is then sent to the client by calling the API “sendtoClientMessageQueue” or the callback “startDeviceLogging”. The log data is encoded by the client in standard LwM2M TLV format and sent to the operator server.

6.2.1 LwM2M Log Message Exchange

Logging information exchanges between the LwM2M client and server uses the following native methods defined by LwM2M protocol:

1. Observe ()
2. Notify ()
3. Cancel Observation ()
4. Read ()

Annex A Source Code

Open source code is publicly available for vendors who are complying with the GSMA TSG IoT standard diagnostic interface at GSMA site:

- https://github.com/GSMATerminals/TSG-IoT-devices-Standard-Diagnostic-Interface
- Summary of the files available are listed below.
- DiagnosticLog.txt - header file specifying the LwM2M logging interface.
- LwM2M_DiagnosticObjects.txt - file specifying the LwM2M log objects and resources
- LwM2M_sampleLogging.txt - example source code demonstrating LwM2M usage for logging.
- README.txt – LwM2M logging Implementation guideline and revision history.
- Note: A user account must be created at the following site before access to the GitHub site. https://github.com/
- All the objects have been officially registered with the Open Mobile Alliance (OMA) Naming Registry, and can be found here - http://openmobilealliance.org/wp/OMNA/LwM2M/LwM2MRegistry.html#extlabel

Annex B Document Management

B.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	January 2018	New PRD TS.41	TSG#30 TG	Dhruv Khettry / Verizon

Other Information

Type	Description
Document Owner	Terminal Steering Group
Editor / Company	Dhruv Khettry / Verizon

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.

questions are always welcome.