CoAP#3 and OMA LWM2M Plugtests;
Las Vegas, USA;
19 – 22 November 2013

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Contents

1 Executive Summary

The CoAP#3 and OMA LWM2M Plugtests event was held from 19 to 22 November 2013 in Las Vegas, USA, co-located with the OMA Technical Plenary and Working Group meetings.

This event was jointly organized by ETSI, the IPSO Alliance and the Open Mobile Alliance (OMA).

Following the 2nd CoAP Plugtests, held from 28 to 30 November 2012 in Sophia-Antipolis, (France), ETSI has been asked by several participants to hold another interoperability event on CoAP in 2013. The previous event used draft specifications which have significantly evolved over the past year. As CoRE CoAP is almost an RFC (Draft version 18 final), it was clear for many that it was a good time to have another event.

It was also a good opportunity to go beyond CoAP, by proposing that companies test the CoAP security using DTLS and the brand new protocol OMA Lightweight M2M, which is based on CoAP.

This is in line with the standardization work of the oneM2M Partnership Project, where CoAP, DTLS and OMA LWM2M are considered as a key component of the future global standardized M2M architecture.

This event had small but fruitful participation of 8 companies providing various CoAP clients and servers, DTLS and OMA LWM2M implementations.

The conclusions are that

- All implementations have been compatible on a basic level
- CoAP standards are mature (this applies to the parts of base standards that were covered in the Plugtests event)
- As could be expected, the recent changes in the Block and Observe specifications have caused some interoperability problems , in other parts than the ones that have been stable since CoAP #2. It shows the importance of keeping on the interoperability events on this technology.
- The level of Interoperability of OMA LWM2M is excellent, especially for a first event. As the scenarios were basic, it shows a good maturity on basic level. The testing needs now to be extended with more deep test scenarios

2 Introduction

This Plugtests event aimed to test the interoperability of CoAP client and server implementations, DTLS as well as OMA LWM2M client and server implementations

The implementations were connected via both IPv6 and IPV4 test networks.

2 Test documents have been used for the testing:

- A Plugtests guide was produced by ETSI containing 55 interoperability tests on CoAP (CoRE, Block, Obs and Link) and DTLS.
- An Enabler Test Specification of LWM2M was produced by OMA, containing 11 interoperability tests

ETSI provided the interoperability tool suite including the wiki, scheduling, test reporting tool and the network infrastructure.

Each day test sessions for IOP assessment were conducted. At the end of each day a wrap-up meeting was held to discuss the main interoperability points of the day.

3 Base Specifications

The following documents were used as basis for the tests:

[1]	Constrained Application Protocol (CoAP); draft-ietf-core-coap-18
[2]	Core Link Format; RFC 6690
[3]	Observing Resources in CoAP; draft-ietf-core-observe-11
[4]	Blockwise transfers in CoAP; draft-ietf-core-block-14
[5]	Lightweight Machine to Machine Technical Specification: Draft Version 1.0 – 05 Nov 2013

4 Abbreviations

CoAP	Constrained Application Protocol
NO	Test is recorded as NOT successfully passed.
NA	Test is not applicable.
OK	Test is recorded as successfully passed.
OT	Test is recorded as not being executed due to lack of time.
Test Session	A paring of vendors that test together during a given time slot.
TSR	Test Session Report. Report created during a test session.

5 Participants

The companies which attended the Plugtests event are listed in the table below.

Table 1: List of implementations that participated in the tests

#	Company
1	**ARM**
2	**ERICSSON**
3	**ETH Zurich**
4	**ETRI**
5	**HUAWEI**
6	**RIOT**
7	**TZI / Uni Bremen**
8	**University of Luebeck**

Table 2: List of Plugtests team

#	Company	Role
1	**ETSI**	Organization of Plugtest, Test Network, Test Descriptions, Lossy Gateway
2	**OMA**	Organization of Plugtest, Test Descriptions

6 Technical and Project Management

All the information presented in this chapter is an extract of the ETSI event wiki https://services.plugtests.net/wiki/CoAP3-OMA-LWM2M/index.php/Main_Page (access for registered people only).

6.1 Test Plan

The test plan containing 55 interoperability tests was developed by ETSI CTI. The coverage of the specifications has considerably been improved regarding the former CoAP Plugtests.

OMA has provided an Enabler Test Specification of LWM2M containing 11 interoperability tests.

During the event preparation, companies had the possibility to review the test plan and to propose additional tests.

The tests were grouped in 6 categories: CoRE, Block, Link, Observe, DTLS and OMA LWM2M tests.

The features covered by all tests are listed below:

- CoAP Testing based on updated base specifications (updated since the first CoAP Plugtests event)
  - Constrained Application Protocol (CoAP); draft-ietf-core-coap-18
  - Core Link Format; RFC 6690
  - Observing Resources in CoAP; draft-ietf-core-observe-11
  - Blockwise transfers in CoAP; draft-ietf-core-block-14
- DTLS as profiled in draft-ietf-core-coap-18:
  - Datagram Transport Layer Security Version 1.2, RFC 6347, January 2012
  - AES-CCM Cipher Suites for Transport Layer Security (TLS), RFC 6655
  - AES-CCM ECC Cipher Suites for TLS, draft-mcgrew-tls-aes-ccm-ecc-06
  - Out-of-Band Public Key Validation for Transport Layer Security (TLS), draft-ietf-tls-oob-pubkey-07
- OMA LWM2M
  - OMA-TS-LightweightM2M-V1_0-20131105-D: Lightweight Machine to Machine: Technical Specification
  - OMA-ETS-LightweightM2M-V1_0-20131017-D: Enabler Test Specification for Lightweight M2M

Table 3: CoAP Tests

TD_COAP_CORE_01	Perform GET transaction (CON mode)
TD_COAP_CORE_02	Perform DELETE transaction (CON mode)
TD_COAP_CORE_03	Perform PUT transaction (CON mode)
TD_COAP_CORE_04	Perform POST transaction (CON mode)
TD_COAP_CORE_05	Perform GET transaction (NON mode)
TD_COAP_CORE_06	Perform DELETE transaction (NON mode)
TD_COAP_CORE_07	Perform PUT transaction (NON mode)
TD_COAP_CORE_08	Perform POST transaction (NON mode)
TD_COAP_CORE_09	Perform GET transaction with separate response (CON mode, no piggyback)
TD_COAP_CORE_10	Perform GET transaction containing non-empty Token option (CON mode)
TD_COAP_CORE_11	Perform GET transaction containing non-empty Token with a separate response (CON mode)
TD_COAP_CORE_12	Perform GET transaction using empty Token (CON mode)
TD_COAP_CORE_13	Perform GET transaction containing several URI-Path options (CON mode)
TD_COAP_CORE_14	Perform GET transaction containing several URI-Query options (CON mode)
TD_COAP_CORE_15	Perform GET transaction (CON mode, piggybacked response) in a lossy context
TD_COAP_CORE_16	Perform GET transaction (CON mode, delayed response) in a lossy context
TD_COAP_CORE_17	Perform GET transaction with a separate response (NON mode)
TD_COAP_CORE_18	Perform POST transaction with responses containing several Location-Path options (CON mode)
TD_COAP_CORE_19	Perform POST transaction with responses containing several Location-Query options (CON mode)
TD_COAP_CORE_20	Perform GET transaction containing the Accept option (CON mode)
TD_COAP_CORE_21	Perform GET transaction containing the ETag option (CON mode)
TD_COAP_CORE_22	Perform GET transaction with responses containing the ETag option and requests containing the If-Match option (CON mode)
TD_COAP_CORE_23	Perform PUT transaction containing the If-None-Match option (CON mode)
TD_COAP_CORE_31	Perform CoAP Ping (CON mode)

Table 4: Link Tests

TD_COAP_LINK_01	Access to well-known interface for resource discovery
TD_COAP_LINK_02	Use filtered requests for limiting discovery results
TD_COAP_LINK_03	Handle empty prefix value strings
TD_COAP_LINK_04	Filter discovery results in presence of multiple rt attributes
TD_COAP_LINK_05	Filter discovery results using if attribute and prefix value strings
TD_COAP_LINK_06	Filter discovery results using sz attribute and prefix value strings
TD_COAP_LINK_07	Filter discovery results using href attribute and complete value strings
TD_COAP_LINK_08	Filter discovery results using href attribute and prefix value strings
TD_COAP_LINK_09	Arrange link descriptions hierarchically

Table 4: Block Tests

TD_COAP_BLOCK_01	Handle GET blockwise transfer for large resource (early negotiation)
TD_COAP_BLOCK_02	Handle GET blockwise transfer for large resource (late negotiation)
TD_COAP_BLOCK_03	Handle PUT blockwise transfer for large resource
TD_COAP_BLOCK_04	Handle POST blockwise transfer for creating large resource
TD_COAP_BLOCK_05	Handle POST with two-way blockwise transfer
TD_COAP_BLOCK_06	Handle GET blockwise transfer for large resource (early negotiation, 16 byte block size)

Table 5: OBS Tests

TD_COAP_OBS_01	Handle resource observation with CON messages
TD_COAP_OBS_02	Handle resource observation with NON messages
TD_COAP_OBS_04	Client detection of deregistration (Max-Age)
TD_COAP_OBS_05	Server detection of deregistration (client OFF)
TD_COAP_OBS_06	Server detection of deregistration (explicit RST)
TD_COAP_OBS_07	Server cleans the observers list on DELETE
TD_COAP_OBS_08	Server cleans the observers list when observed resource content-format changes
TD_COAP_OBS_09	Update of the observed resource
TD_COAP_OBS_10	GET does not cancel resource observation

Table 6: DTLS

TD_COAP_DTLS_01	Basic DTLS PSK (success case)
TD_COAP_DTLS_02	Basic DTLS PSK (failure case — wrong PSK)
TD_COAP_DTLS_03	Lossy DTLS PSK (success case)
TD_COAP_DTLS_04	Basic DTLS RPK (success case)
TD_COAP_DTLS_05	Basic DTLS RPK (client failure case)
TD_COAP_DTLS_06	Basic DTLS RPK (server failure case)
TD_COAP_DTLS_07	Lossy DTLS RPK (success case)

Table 7: LWM2M Tests

Registration	LightweightM2M-1.0-int-101 – Initial Registration
	LightweightM2M-1.0-int-102 – Registration Update
	LightweightM2M-1.0-int-103 – Deregistration
Device object-related use cases	Querying basic information from the client
	Querying the firmware version from the client
	Rebooting the device
	Querying power status of the terminal
Device firmware update	LightweightM2M-1.0-int-301 – Firmware update (via COAP)
	LightweightM2M-1.0-int-302 – Firmware update (via alternative mechanism)
Connectivity object monitoring	LightweightM2M-1.0-int-401 – Querying of connectivity parameters
Observe and Notify	LightweightM2M-1.0-int-501 – Observation and notification of parameter values inside MachineLink 3G

6.2 Test Scheduling

The preliminary test schedule was developed before the Plugtests and was circulated to all the participants in advance for comments. The initial test schedule allowed for each company to test against a fair number of other companies. Two companies were assigned one test slot which had duration of 3 hours. In this test slot the companies could run tests for the configurations: CompA-Client-CompB- Server and CompA-Server-CompB-Client for CoAP, DTLS and LWM2M. Up to 3parallel test sessions were planned.

During the test event the test schedule was updated according to the progress of the test sessions. This was done during the daily wrap-up meetings at the end of each day and during face-to-face meetings with the participants.

The figure below shows the final version of the test schedule.

Figure 1: Test Schedule

		Area 1	Area 2	Area 3	Area 4
Tue 19	9:00-12:00	Eth Zurich ETH Zurich device TZI TZI device	ETRI ETRI device University of Luebeck Uni Luebeck device		
	13:00-16:00	ARM ARM Device Ericsson Ericsson device	TZI TZI device University of Luebeck Uni Luebeck device	Eth Zurich ETH Zurich device ETRI ETRI device	
	16:00-19:00	ARM ARM Device Huawei Huawei device	Eth Zurich ETH Zurich device University of Luebeck Uni Luebeck device	TZI TZI device ETRI ETRI device	
Wed 20	9:00-12:00	ARM ARM Device Eth Zurich ETH Zurich device	TZI TZI device Ericsson Ericsson device	RIOT RIOT device University of Luebeck Uni Luebeck device	
	13:00-16:00	ARM ARM Device University of Luebeck Uni Luebeck device	ETRI ETRI device Ericsson Ericsson device	Eth Zurich ETH Zurich device Huawei Huawei device	
	16:00-19:00	Ericsson Ericsson device Huawei Huawei device	ARM ARM Device TZI TZI device		
Thu 21	9:00-12:00	Eth Zurich ETH Zurich device Ericsson Ericsson device	ARM ARM Device ETRI ETRI device	TZI TZI device RIOT RIOT device	
	13:00-16:00	University of Luebeck Uni Luebeck device Ericsson Ericsson device	TZI TZI device Huawei Huawei device		
	16:00-19:00	ETRI ETRI device Huawei Huawei device			
Fri 22	9:00-12:00	Eth Zurich ETH Zurich device RIOT RIOT device			
	13:00-16:00				

6.3 Interoperability Test Procedure

Each test was executed in the same manner as listed below:

1) Connect client and server over test network

2) Check connectivity between devices

3) Perform tests according to the Plugtests guide

   a. Check if test runs to completion

   b. Check results from an interoperability point of view:
   Is the intended result visible at the application layer?

4) Result determination and reporting

   a. Result OK: run next test

   b. Result not OK: check monitor tools to identify source of error

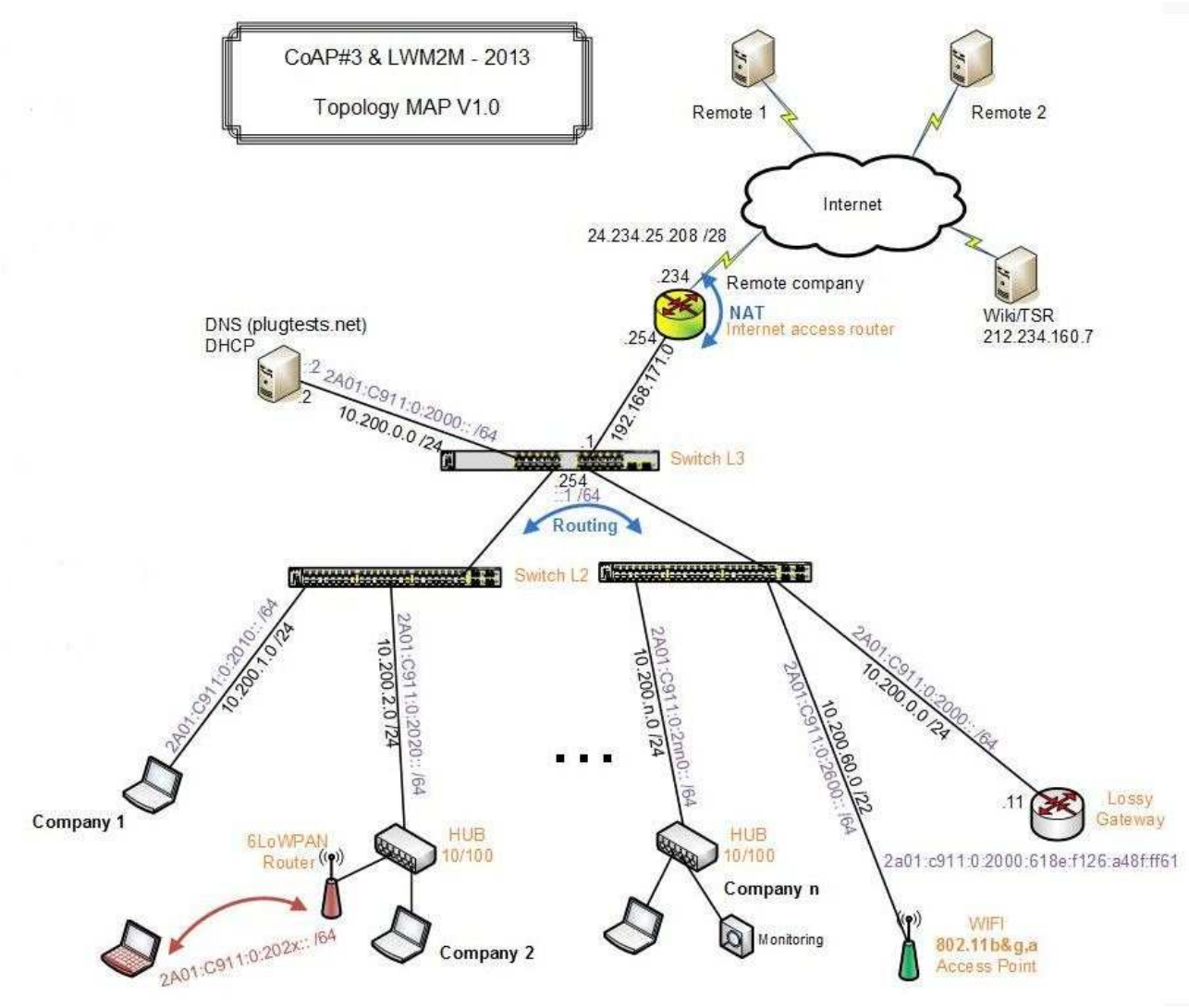
   c. Report results in ETSI Test Reporting Tool

5) Once all tests executed swap client / server roles and run all tests again

6.4 Test Infrastructure

The test infrastructure provided for the Plugtests event is shown below.

Figure 2: Test Network



6.5 Tooling

6.5.1 ETSI Test Reporting Tool

The purpose of the ETSI Test Reporting Tool (TRT) is to provide a means to report the test sessions. It provides statistical overviews of the test results. The graphical information in the latter section on results was created with the ETSI TRT. It also provides a means to create a test schedule (see section 6.2).

6.5.2 Lossy Gateway

The purpose of the UDP lossy gateway is to perform packet loss in CoAP conversations according to the lossy context test descriptions defined in the Plugtests guide.

The configuration of the setup is shown below:

CoAP Client ----- UDP Lossy Gateway ----- CoAP Server

Figure 3: UDP Lossy Gateway Configuration

The UDP lossy gateway assigns one listening port for each CoAP server. Thus the UDP lossy gateway provides for each CoAP server a unique lossy address.

A CoAP client that does lossy context test sends the CoAP message to the lossy address of the specified CoAP server. Then the UDP lossy gateway decides the right destination address according to the UDP socket on which the message was received.

Then the UDP lossy gateway starts a new UDP socket to communicate with the appropriate CoAP server. This UDP socket is also used for forwarding back the CoAP server's responses to the right CoAP client. The server-side communication expires after idling 5mn.

Packet loss is performed at 2 places:

- forwarding CoAP client's message to the CoAP server
- forwarding back CoAP server's message to the CoAP client

The program generates random numbers to decide whether to perform packet loss or not. A 50% packet loss rate was used for the Plugtests.

6.5.3 UDP V4-V6 Gateway

Some participants needed to perform testing between only-V4 devices and only-V6 devices. For enabling such pairings, the UDP lossy gateway has been used with a loss = 0 in the setting.

6.5.4 Pre-Testing

Prior to the event, 3 companies had posted on the wiki the addresses of CoAP servers, in order to enable the participants to run pre-testing. The feedback we received is that it has been appreciated and helpful for preparing the event.

7 Achieved Results

The achieved results show that all implementations have been compatible on a basic level, i.e. sent data could be decoded and interpreted properly by receivers and a vast majority of equipment performed well.

7.1 Overall CoAP Results

Due to NDA constraints, it is not possible to provide detailed results.

The figure below shows the overall result of mandatory and optional tests. In total more than 888 tests were executed.

The execution rate of 66.1% is a satisfying result, especially considering the high number of tests proposed in such short event. Each test session lasted 3hours which is very short as most of the companies had several devices (client and servers) which of course increased the number of possible pairing combinations. Globally the feedback that the participants gave is that the testing was very dense.

It was possible to add further test sessions in addition to the scheduled ones, to allow participants to re-run the tests or complete their testing.

94.1% of the test verdicts were OK which shows a very high level of maturity of the implementations.

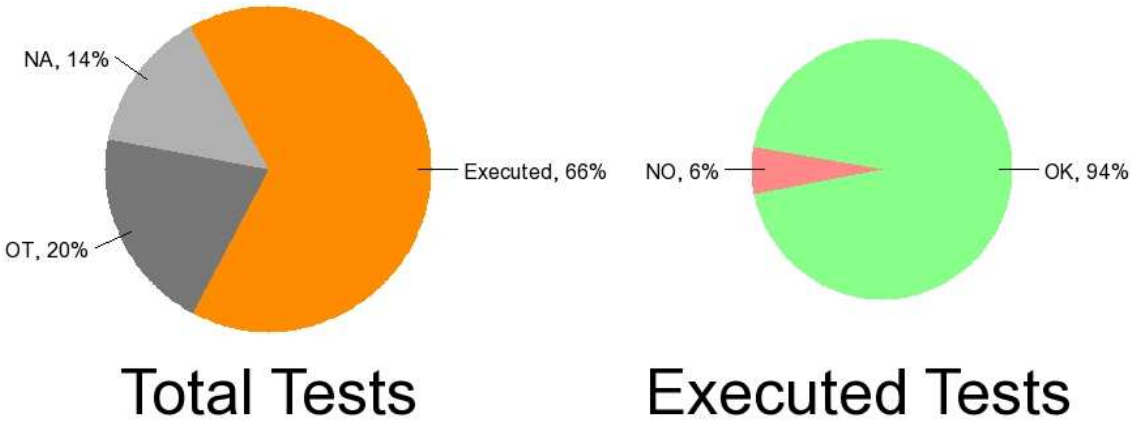


Figure 4: Overall CoAP Results

	Interop Test Executed		Not executed		Total	
Group	OK	NO	NA	OT	Run	Results
BLOCK	81 (86.2%)	13 (13.8%)	30 (17.9%)	44 (26.2%)	94 (56.0%)	168
CoRE	543 (95.9%)	23 (4.1%)	40 (6.0%)	66 (9.8%)	566 (84.2%)	672
LINK	128 (98.5%)	2 (1.5%)	53 (21.0%)	69 (27.4%)	130 (51.6%)	252
OBS	84 (85.7%)	14 (14.3%)	62 (24.6%)	92 (36.5%)	98 (38.9%)	252

7.2 Results of CoRE tests

There were 24 test scenarios defined in the test plan. In total 566 tests were executed with a success rate of 95.9%.

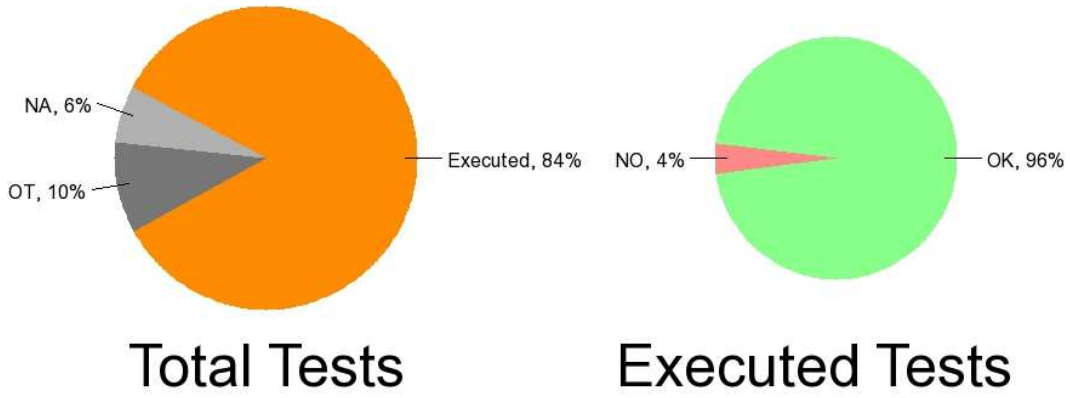


Figure 5: Results of CoRE tests

7.3 Results of Block tests

There were 6 Block test scenarios defined which were to be executed per session. In total 98 tests were executed with a success rate of 86.2%.

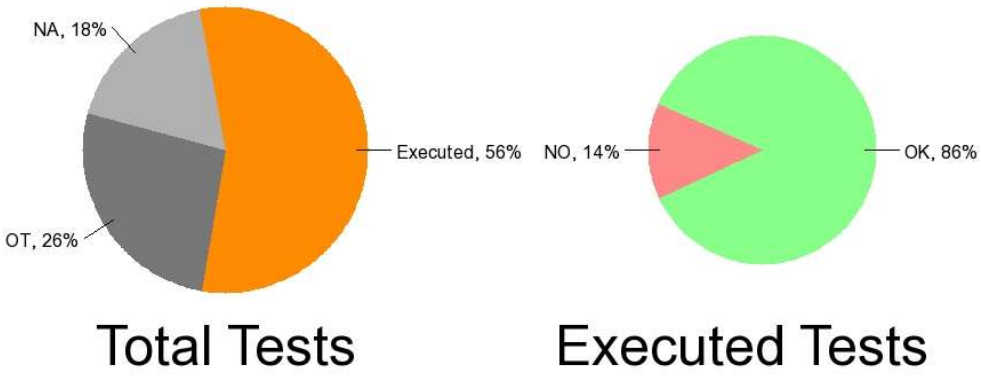


Figure 6: Results of Block tests

7.4 Results of Link tests

There were 9 Link test scenarios defined which were to be executed per session. In total 130 tests were executed with a success rate of 98.5% which show a high maturity of the RFC.

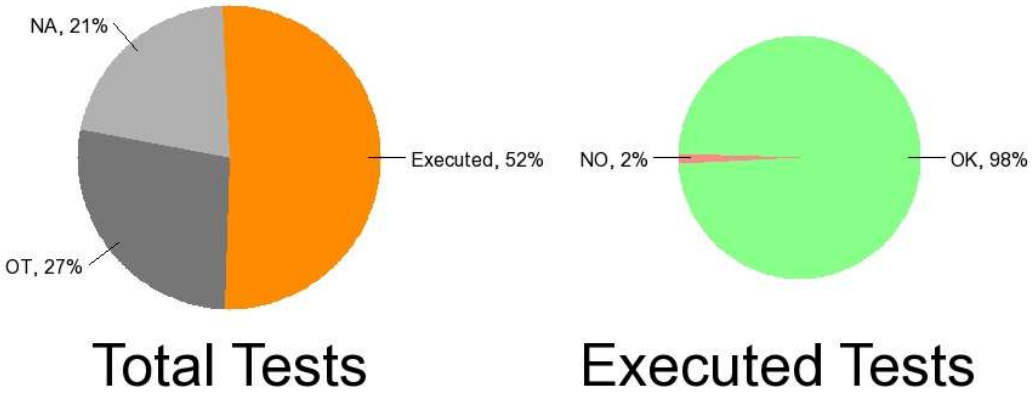


Figure 6: Results of Link tests

7.5 Results of Observe tests

There were 9 Observe test scenarios defined which were to be executed per session. In total 98 tests were executed with a success rate of 85.7%.

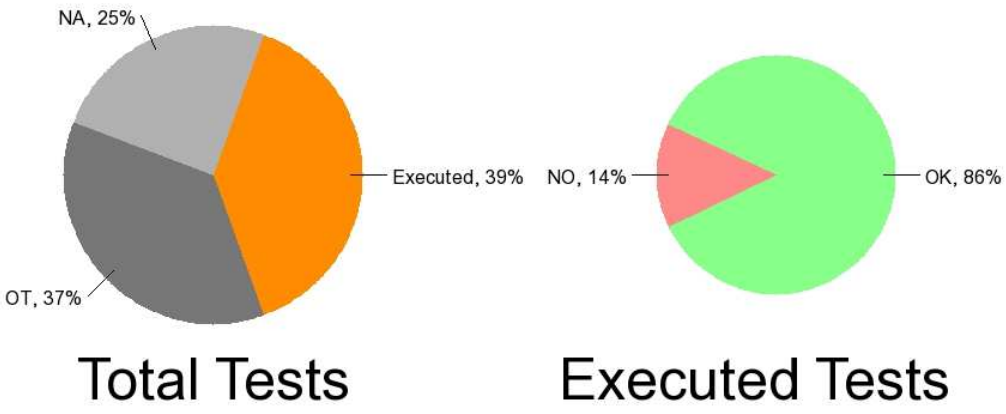


Figure 6: Results of Link tests

7.6 Results of DTLS tests

There were 7 DTLS test scenarios defined which were to be executed per session. In total only 3 tests were executed with a success rate of 66.7%. This result is not significant due to the low number of tests which were run. Only one session was performed due to a lack of implementations supporting DTLS.

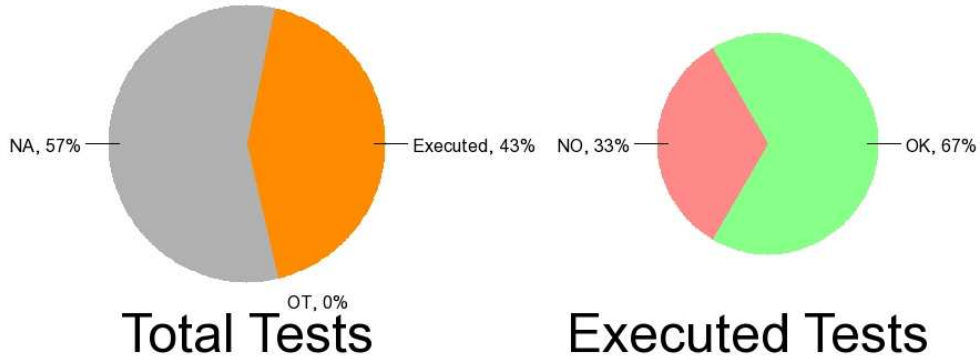


Figure 7: Results of DTLS tests

7.7 Results of OMA LWM2M tests

There were 11 LWM2M test scenarios defined which were to be executed per session. In total 39 were executed with a success rate of 97.4% which is an excellent rate showing the maturity of the new enabler.

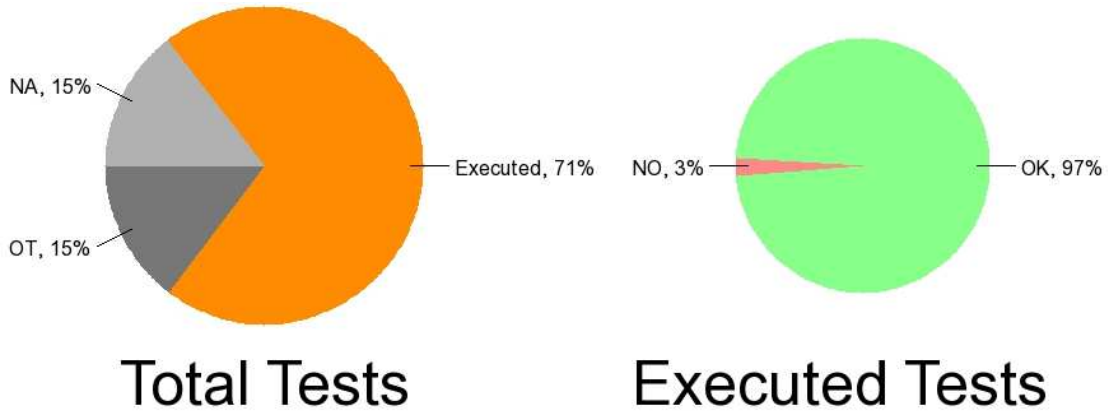


Figure 8: Results of OMA LWM2M tests

8 Summary of Wrap Up Sessions

8.1 IOP Issues

As could be expected, the recent changes to certain less often exercised cases in the Block and Observe specifications still exhibit a larger number of interoperability problems than the parts that have been stable since CoAP #2.

Only a small number of implementations were available with DTLS support. With these, only the PSK set of tests could be completed. While 2 out of 3 tests succeeded with their main objective, all test runs exhibited problems with correctly handling retransmissions after packet losses.

8.2 Test Spec Issues

Nine test descriptions were not explicitly requiring a Content-Format option for a success response with non-empty payload; this has been amended during the event. Similarly, TD_COAP_OBS_07 and TD_COAP_OBS_08 were amended to no longer show an Observe option in a response when the observation relationship has ended. New test descriptions were developed based on problematic behaviour that was not covered by existing test descriptions: TD_COAP_BLOCK_06 covers the special case of a zero-length Block option; TD_COAP_OBS_10 tests that previously required behaviour with respect to GET without an Observe option is not accidentally still implemented; an additional check was added to TD_COAP_CORE_16 about stopping retransmissions after a loss; a lossy case was added for Observe (TD_COAP_OBS_11); TD_COAP_CORE_22 was completed to test that another update will still succeed. Finally, a new test description TD_COAP_CORE_31 was added to test interoperability of the CoAP "ping" mechanism that was not previously addressed by the test specification.

There were different interpretations about the meaning of a payload with LightweigthM2M-1.0-int-102 (registration information update). Since a payload is not strictly required to complete this test, this didn't cause test failures, but it is probably worthwhile adding test descriptions to explore the various interpretations further.

8.3 Base Specification Issues

As of November 2013, only the PSK tests for DTLS have protocol numbers defined by IANA. For the RPK tests, the test specifications had to invent some temporary protocol numbers to enable interoperability. (Unfortunately, in the end time did not suffice to complete the RPK tests enabled by these temporary numbers.) This is expected to be remedied once the remaining draft security specifications have been processed by IANA.

The initial timer values for the retransmission timers defined by RFC 6347 may be too short for the slow asymmetric cryptography operations exhibited by very constrained nodes. (This compounds the implementation problems with DTLS retransmission, but is an independent problem of its own.) The IETF DICE WG provides a venue that could

examine the retransmission approach of DTLS and propose a retransmission mechanism more adapted to constrained devices.

The OMA LWM2M tests uncovered that the specification can be interpreted in different ways with respect to forming URIs for Objects that do not support multiple Object Instances. After detecting this problem, all participants agreed to use the interpretation suggested by section 8.2.4 (for instance, GET /3/0/1, not GET /3//1) and achieved interoperability. One participant took on to submit a change request to OMA.

Annex A CoAP Interoperability Test Specification

The CoAP Interoperability Test Specification, which forms parts of the present technical report, is contained in the file IoT_CoAP3_TestSpecification_005.pdf.

History

Document history		
V0.0.1	November 2013	Initial version
V0.0.2	December 2013	Editorial Changes
V0.0.3	December 2013	Input of technical conclusion
V1.0.0	December 2013	Final version