ETSI TR 103 527 V1.1.1 (2018-07)

SmartM2M;
Virtualized IoT Architectures with Cloud Back-ends

Reference
DTR/SmartM2M-103527

Keywords
cloud, IoT, virtualisation

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

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (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
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Contents

List of figures

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

In addition to interoperability and security that are two recognized key enablers to the development of large IoT systems, a new one is emerging as another key condition of success: virtualization. The deployment of IoT systems will occur not just within closed and secure administrative domains but also over architectures that support the dynamic usage of resources that are provided by virtualization techniques over cloud back-ends.

This new challenge for IoT requires that the elements of an IoT system can work in a fully interoperable, secure and dynamically configurable manner with other elements (devices, gateways, storage, etc.) that are deployed in different operational and contractual conditions. To this extent, the current architectures of IoT will have to be aligned with those that support the deployment of cloud-based systems (private, public, etc.).

Moreover, these architectures will have to support very diverse and often stringent non-functional requirements such as scalability, reliability, fault tolerance, massive data, security. This will require very flexible architectures for the elements (e.g. the application servers) that will support the virtualized IoT services, as well as very efficient and highly modular implementations that will make a massive usage of Open Source components.

These architectures and these implementations form a new approach to IoT systems and the solutions that the present document investigates also should be validated: to this extent, a Proof-of-Concept implementation involving a massive number of virtualized elements has been made.

The present document is one of three Technical Reports addressing this issue:

- ETSI TR 103 527 (the present document): "Virtualized IoT Architectures with Cloud Back-ends" (the present document);
- ETSI TR 103 528 [i.1]: "Landscape for open source and standards for cloud native software for a Virtualized IoT service layer";
- ETSI TR 103 529 [i.2]: "Virtualized IoT over Cloud back-ends: A Proof of Concept".

1 Scope

The present document:

- makes a description of some use cases that benefit from virtualization and outlines which one will be used for the Proof-of-Concept that is described in depth in ETSI TR 103 529 [i.2];
- addresses the rationale and requirements for the use of virtualization - and of the cloud in general - in support of IoT systems. It also introduces some features that will be key for the definition and further implementation of virtualized IoT systems such as microservices;
- provides the identification of new architectural elements (components, mappings, Application Programming Interfaces (API), etc.) that are required to address IoT on a cloud back-end. In particular, one objective of the present document is to describe how current IoT nodes e.g. the oneM2M CSE, can be modified and improved by the introduction of micro-services.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI TR 103 528: "SmartM2M; Landscape for open source and standards for cloud native software applicable for a Virtualized IoT service layer", 2018.

[i.2] ETSI TR 103 529: "SmartM2M; IoT over Cloud back-ends: a Proof of Concept", 2018.

[i.3] ITU-T News: "What is 'cloud-native IoT' and why does it matter?", October 2017.

NOTE: Available at http://news.itu.int/what-is-cloud-native-iot-why-does-it-matter/.

[i.4] Amazon Web Services: "What is Auto-scaling".

NOTE: Available at http://docs.aws.amazon.com/autoscaling/latest/userguide/WhatIsAutoScaling.html.

[i.5] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation).

NOTE: Available at https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=celex%3A32016R0679.

[i.6] Deloitte: "Data Privacy in the cloud", 2016.

NOTE: Available at https://www2.deloitte.com/content/dam/Deloitte/ca/Documents/risk/ca-en-risk-privacy-in-the-cloud-pov.PDF.

[i.7] ETSI TS 118 101 (V2.10.0): "oneM2M; Functional Architecture (oneM2M TS-0001 version 2.10.0 Release 2)".

[i.8] Recommendation ITU-T Y.3600: "Big data - Cloud computing-based requirements and capabilities", 2015.

[i.9] ETSI GS NFV 002: "Network Functions Virtualisation (NFV); Architectural Framework".

[i.10] ETSI GS NFV-INF 001: "Network Functions Virtualisation (NFV); Infrastructure Overview".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

Open Source Software (OSS): computer software that is available in source code form

NOTE: The source code and certain other rights normally reserved for copyright holders are provided under an open-source license that permits users to study, change, improve and at times also to distribute the software.

source code: any collection of computer instructions written using some human-readable computer language, usually as text

standard: output from an SSO

Standards Setting Organization (SSO): any entity whose primary activities are developing, coordinating, promulgating, revising, amending, reissuing, interpreting or otherwise maintaining standards that address the interests of a wide base of users outside the standards development organization

NOTE: In the present document, SSO is used equally for both Standards Setting Organization or Standards Developing Organization (SDO).

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AE	Application Entity (in oneM2M)
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
ARM	Acorn RISC Machine architecture
BCP	Best Common Practices
CAPEX	Capital Expenditure
CEP	Complex Event Processing
CoAP	Constrained Application Protocol
CPU	Central Processing Unit
CSC	Cloud Service Customer
CSE	Common Services Entity (in oneM2M)
CSF	Common Service Function
CSP	Cloud Service Provider
DDoS	Distributed Denial of Service
EU	European Union
GDPR	Global Data Protection Regulation
HLA	High Level Architecture
HTTP	HyperText Transfer Protocol
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
ICT	Information and Communication Technology
IoT	Internet of Things
IP	Internet Protocol
IPC	Inter-Process Communication
IPE	Interworking Proxy Entity (in oneM2M)

ISG	Industry Specification Group
IT	Information Technology
MANO	MANagement and Organization (in NFV)
MQTT	Message Queuing Telemetry Transport
NFV	Network Function Virtualisation
NFVI	NFV Infrastructure
ONAP	Open Network Automation Platform
OSM	Open Source Mano (in ETSI)
OSS	Open Source Software
PaaS	Platform as a Service
PoC	Proof-of-Concept
PoP	Point of Presence
SaaS	Software as a Service
SDO	Standards Development Organization
SE	Service Entity (in oneM2M)
SPOF	Single Point Of Failure
SSO	Standards Setting Organization
UC	Use Case
URI	Uniform Resource Identifier
VM	Virtual Machine
VNF	Virtualized Network Function

4 Rationale for IoT Virtualization

4.1 IoT: towards massive deployments

The focus of IoT in the recent years has been on connecting devices and applications. To this extent, a number of standards, frameworks, solutions have been developed. Now that the maturation of the industry is progressing rapidly, IoT is facing to major challenges.

On the one hand, connected devices as well as applications have to be integrated with existing, evolving or entirely new business processes: this creates the need for very adaptive frameworks that offer the possibility to easily introduce new applications and to ensure that they are properly connected to the existing enterprise systems, and to process enormous quantity of data.

One the other hand, IoT systems are transitioning from proof-of-concept deployments or new projects with limited size and scope towards full-fledge systems. These new systems may require extremely high numbers of connected devices (thus generating needs for scalability or deployment automation) as well as stringent non-functional requirements (such as low latency).

In both cases, new IoT systems will require a high degree of availability, adaptability and flexibility. In particular, the resources used by those systems may have to be very dynamic, both in terms of configuration and run-time flexibility. The models provided by Cloud Computing, which have been designed upfront with these two requirements in mind, seem very attractive in this context.

4.2 Cloud Computing and Virtualization

Cloud computing is allowing the provision of very sophisticated capabilities; for computing, storage, analytics, etc.; to very dynamic and potentially massive number of users. Those capabilities are provided as services (Platform-as-a-Service, Infrastructure-as-a-Service; Software-as-a-Service; etc.) that provides functional and also non-functional support (e.g. low latency fault-tolerance, horizontal scalability, cost-optimization, or geo-optimization together with Service Level Agreements (SLAs), and security.

The technical capabilities of cloud computing technology made it possible to provide the most demanding information and communication technology (ICT) infrastructures, such as communication networks, from specialized hardware and software to new software paradigms, referred to as 'cloud-native'.

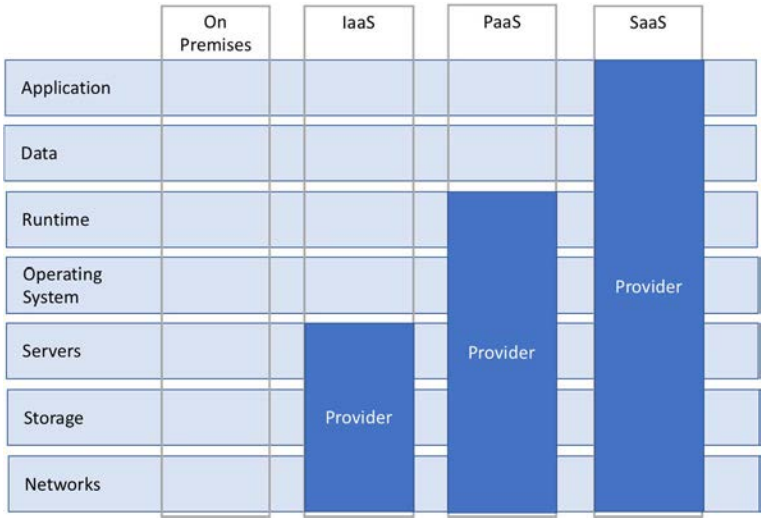

Figure 1: Options for adoption of Cloud Native solutions

The expectation of Cloud-Native applications is to benefit from offerings from Cloud Service Providers (CSP) that may cover parts or all of the layers of Virtualized application, via Infrastructure as a Service (IaaS), Platform as a Service (PaaS) or Software as a Service (SaaS). Figure 1 presents the possible usages of such offerings in delegating more and more important parts of the underlying layers to a third-party in charge of hiding complexity, resource usage, etc.

4.3 The new challenge: combining IoT and Cloud Computing

The IoT industry starts to understand the potential benefits of combining the strengths of both IoT and Cloud industries in a new value proposition (see [i.3] for example). IoT virtualization - i.e. IoT built on cloud-native principles - is to IoT platforms as what Network Function Virtualisation (NFV) is to communication networks.

When applied to IoT, virtualization is expected to provide technical benefits such as more flexibility on assigning IoT virtualized objects and functions to physical resources. Moreover, virtualization should bring as well financial benefits (e.g. greater CAPEX efficiency) or operational benefits (e.g. improvement of automation and operating procedures) altogether resulting in boosted service innovation.

The scope of IoT standards and protocols has so far focused on interface specifications and related data models. Developing IoT platforms that use cloud-native principles will benefit from guidelines and Best Common Practices (BCP) in building operational grade IT applications using cloud technologies.

The convergence of cloud and IoT is of major important to those (e.g. architects) aiming at building IoT solutions that can dynamically reach massive scale in support of large IoT deployments, e.g. in Smart Cities. It is of great importance for technical actors of IoT to benefit from guidelines for IoT virtualization, in particular regarding the 'containerisation' of IoT applications.

4.4 Content of the report

Clause 5 provides a number of Use Cases (UC) that could benefit from virtualization of IoT. Each UC is described from a functional standpoint, together with the expectations towards virtualization. Finally, one UC is highlighted since it is the one that will be selected for implementation (as it is described in ETSI TR 103 529 [i.2]).

Clause 6 of the present document presents the Cloud Computing features that are relevant in the context of IoT Virtualization. First, some functional requirements are introduced that correspond to specific functionalities that are (better) supported by Virtualization. Similarly, some non-functional requirements are presented that are expected to be specially supported through Virtualization. Finally, two key features that will play a key role in architectures and implementations: microservices and inter-process communications.

Clause 7 investigates the main dimensions that will be addressed in order to define layered architectures supported by microservices. A reference model for such an architecture is introduced that also serves as a basis for the description of the "Landscape of Open Source and Standards" that is developed in ETSI TR 103 528 [i.1].

Clause 8 summarizes the main finding of the present document and provides a set of recommendations for architects and developers in charge of potential IoT virtualization projects.

Annex A is addressing the relationship of IoT with Big Data and, in particular, regarding the question of Data Quality and some potential solutions.

5 Some use cases for IoT Virtualization

5.1 Introduction

This clause introduces a (limited) number of generic Use Cases (UCs) that are illustrative of the expected benefits and potential challenges of IoT virtualization. There is probably a large number of Use Cases for which a "traditional" (i.e. non-virtualized) approach can and will apply. However, the introduction of IoT Virtualization is expected to make some UCs more effective: it would generally improve the efficiency of their implementation or support interoperability at a more fine-grained level (or both).

For the presentation of UCs, rather than present them based on the needs of a given business domain (aka a "vertical"), the approach taken is to present the major features outlined by a class of applications in different "verticals". The name of the UC will refer to the major underlying feature involved (e.g. fault-tolerance, data privacy, etc.).

5.2 Horizontal up and down Auto-Scaling

The amount and type of data transmitted by IoT devices may vary drastically in time depending on some events that can be internal or external to the virtualized IoT system (e.g. road traffic increase during holiday departure).

A cloud-native IoT platform shall be able to continuously monitor its resources, scale-up its capabilities when needed, then scale-down to an optimized state to avoid wasting resources. This capability is referred to as "Auto-Scaling" (see [i.4] for example).

The main objective of Auto Scaling is to ensure that the number of Virtual Machine (VM) instances available for and used by the virtualized application are optimal at a given time. Practically, a minimum number of VM instances is defined (lower threshold for the auto-scaling down) as well as a maximum number (upper threshold for the auto-scaling up). When needed, additional VMs are added (with an increment that can be predefined), used as long as needed and released when the usage is no longer needed.

This UC can be illustrated by a number of examples taken from various verticals:

- Intelligent Transport Systems with a sudden increase in traffic (e.g. vacations).
- Electrical (Smart) Grids with burst reconnection of IoT devices after a power cut.
- Smart Metering with burst transmission of intelligent meters data at given time slots.
- And many more, etc.

The benefits of Auto Scaling are largely related to the non-functional support it provides to the virtualized applications:

- Improved availability: the virtualized application has, at any time, the best adjusted capacity to deal with the most complex and hard to predict traffic patterns.
- Improved fault tolerance: when an instance (or a group of) VM(s) does not function properly, Auto-Scaling allows to quickly terminate it and launch an adequate replacement.
- More effective cost management: thanks to the dynamic increase and decrease of the needed capacity, the usage is constantly adjusted to reduce the consumption, hence the cost, of the computing resources.