The Internet Architecture Board
Interconnecting Smart Object with the Internet Workshop
Prague, Czech Republic, Friday, 25th March 2011
Position Paper

Simplified SIP Approach for the Smart Object

Ryota Ishibashi, Takumi Ohba, Arata Koike, Akira Kurokawa
NTT Corporation, NTT Service Integration Laboratories

Abstract

To achieve the Smart Object world, we need to create appropriate application layer protocols. We consider high extensibility and applicability for human-to-machine type communications as well as machine-to-machine are especially needed for the protocol design of the application layer protocol. Those characteristics are important to support new devices, applications and type of communications which will form the future Smart Object world as well as currently envisioned ones. In this paper, we show that SIP (Session Initiation Protocol) fulfills these requirements of the protocol design and it is suitable for the Smart Object. On the other hand, it is shown SIP lacks lightweightness, which is also one of the most important requirements for the Smart Object. To bring significant sustainability into the Smart Object world by using SIP, we propose "simplified SIP approach" for improving SIP lightweightness and remark the future works in this approach.

1. Introduction

Although it is not easy to define the Smart Object clearly, we consider that in the world of the Smart Object, huge amount of and wide varieties of devices, including RFID tags, sensors, and actuators, which may have extremely strict constraints for physical implementations and/or computing resources are interconnected through various networks such as the Internet, telecom/mobile networks, and utility networks such as power, gas, and water. These devices communicate in varieties of ways for varieties of purposes, such as data sensing, processing, delivering and device controlling, including machine-to-machine (M2M) communication which does not necessarily require the human intervention. To achieve this new communication paradigm, we need to design appropriate protocols.

Especially, we see the great chance of SIP (Session Initiation Protocol) [1] as one of application layer protocols for the Smart Object. In this paper, firstly we will show special requirements of application protocol design for the Smart Object in section 2. Next, we will show that, while SIP fulfills our proposed requirements in section 3, there is a major disadvantage of SIP from the viewpoint of the protocol lightweightness in section 4. Last, we will argue the guideline to simplify SIP to tackle this disadvantage of SIP in section 5 and section 6 summarizes the discussion and shows future works to achieve lightweightness of SIP.

2. Protocol design requirements for the Smart Object

Devices in the Smart Object world usually have poor processing resources such as CPU power and/or installed memory, have no keyboard, display, nor other input/output equipment for human intervention, require extremely low power consumption, and are in poor communication environments where bandwidth is limited or significantly large delay exists. The protocols for the Smart Object tend to be optimized to these strict requirements.

However, we have to consider that the Smart Object will not stand still in the currently envisioned scope. In the future, many kinds of devices and applications will be newly applied, and also the type of communications will become more varied. This brings new functional requirements for the protocol, and therefore it is necessary for the protocol to be extensible to meet the future new fictional requirements.

Regarding the type of communications, M2M communication, where there is no human intervention, is the main area for the Smart Object.

However, human-to-machine (H2M) communications, such as remote surveillance camera control or remote diagnosis in e-health, will become important in the Smart Object world as well.

Therefore, we consider that the protocols used for the Smart Object should have the following requirements from the aspect of protocol design.

- Extensible: the protocol should be designed to be enough extendable in order to support new devices, applications which will appear in the future.
- Applicable for H2M: the protocol should be designed to be capable of supporting H2M communication, rather than being too much specialized for M2M.

These two requirements are not exhaustive, and there are many other protocol requirements for the Smart Object, but we consider these two requirements for the protocol design are the key factors for the Smart Object world being sustainable in the long term.

3. SIP suitability for the Smart Object

We consider SIP fulfills the requirements proposed in Chapter 2, extensibility and applicability for H2M, and it is suitable for the Smart Object in its protocol design.

- ✓ SIP is extensible: SIP is designed as a text-based protocol, while many of other Smart Object specific protocols, for example CoAP [2], are binary-based in fixed length headers principle. Although binary and fixed length design is useful for better processing performance, it may lead to limited extensibility. SIP is highly extensible to cover new functional requirements and applications for the Smart Object.
- ✓ SIP is applicable for H2M: SIP is originally designed

to enable human-to-human (H2H) communications, such as voice/video call, interactive chat, file transfer, etc, and SIP has sufficient functionalities for the human intervention, e.g. "Ringing" response, "CANCEL" request, multimedia session setup, and human readability comes from text-based design. Therefore, SIP is more easily applicable for the H2M rather than the ones completely M2M-specific that do not have these functionalities inherently. In fact, SIP is already being applied to H2M, e.g. remote camera monitoring as shown in the Appendix I.

Of course, M2M specific protocols may be possible to cover H2M communications by using HTTP-based web application interface on the human side and the M2M protocol on the machine side, via HTTP-M2M protocol interworking, for example. However, bi-directionality and mobility will be impaired on HTTP, and also complex protocol interworking may degrade user experiences.

Moreover, we can say SIP brings another advantage to the Smart Object world.

✓ SIP is stable: SIP has become mature with many debugging and practical feedbacks provided from the market for these 15 years[1]. It is not easy to build up newly designed protocol for the Smart Object in the same level of stability.

We can devote such our efforts to developing and verifying new devices or applications and to fertilize the world of the Smart Object instead of developing new protocols.

Here we need to say that we are NOT trying to deny any other protocols than SIP, nor to insist completely alternative choice. Rather, we regard SIP as appropriate protocol for the Smart Object as well as them, and consider we should discuss SIP in comparable level as them.

In addition to the SIP suitability as we mentioned above, SIP also has sufficient fundamental functionalities to support the Smart Object. For the brief description of them, please see Appendix II.

4. Major disadvantage of SIP –lightweightness–

SIP brings significant benefits, extensibility and H2M applicability, to the Smart Object world. However, we have to say that SIP, as it is, cannot be widely applied for the Smart Object, because SIP has a disadvantage about its lightweightness, which is exactly one of the most important requirements of the Smart Object.

× SIP is NOT lightweight: SIP is designed as a text-based protocol, so it requires heavy text-parse processing, which is mostly unacceptable for the constrained devices in the Smart Object. In terms of traffic, the number of SIP messages in a certain sequence and the size of SIP messages tend to increase[2]to support a lot of functions, as part of them are stated in Appendix II. From the viewpoint of SIP server performance, we should say that even high-performance carrier grade SIP servers are insufficient to accommodate billions of devices in the Smart Object[3].

Although there may be some devices or situations that can avoid these problems of SIP, for example, relatively high-performance devices such as smart phones, or the domains requires only limited communications, most of the Smart Object devices and environments will not. Therefore, we consider that SIP needs to be appropriately simplified in order to be applied to wider devices and environments.

5. Simplified SIP approach for the Smart Object

To make SIP lightweight, SIP has to be simplified by eliminating unnecessary parts of SIP functionalities depending on the devices, applications, and patterns of communications. The way of simplifying SIP needs to be determined carefully.

We consider there should be basic policies to be always taken in consideration as general principles when we try to determine what can be eliminated and what cannot from the original SIP. Here are our understandings as a starter of investigation about such basic policies.

What should not be eliminated in any cases are the following SIP core functionalities;

- Registration: SIP registration mechanism allows IP-address-independent communication in much scalable way. Although combination of host name and Dynamic DNS also provides similar capability, it takes a time before the reflection of updates.
- Authentication: It will not be easy to preclude the abuse attempts by malicious attackers, especially in the Smart Object world where uncountable devices exist. Therefore application layer client authentication will be mandatory.
- Discovery: Since a lot of automated devices communicate each other and they have wide variety of resources and/or capabilities, systematic discovery mechanism of the devices, resources and capabilities without manual operation is required.

On the other hand, what can be eliminated in some cases depending on the situations are the followings;

[1] SIP is used as one of the main signaling protocols for high-quality voice/video services of NTT in Japan, delivered to over 10 million users. This is an evidence of the SIP having sufficient stability for the practical usage.

[2] As an example, we have obtained the result that, the inter-domain message traffic of extremely tuned SIP and that of normally operated XMPP [3] are almost in same level, by comparing the results in two internet drafts which give simple estimation of presence traffic by SIP and XMPP respectively[4][5].

[3] Assuming the performance of a SIP server is 1 million BHCA (Busy Hour Call Attempts) and connection is requested once every 10 minutes in Poisson arrival, even when the SIP messages can be reduced to one-third in consideration of the limited M2M communication from the normal call setup, 100 SIP servers can accommodate at most 50 million devices.

- Flexible representations: Co-existence of both compact and longer forms and arbitrary ordering of SIP headers bring processing complexity and they are not mandatory functionalities.
- Complex routing: SIP provides highly complex routing mechanism, but it requires more processing resources. This is unnecessary if the devices connect specific server only.
- Security (excludes authentication): SIP layer security mechanisms for message protection such as S/MIME, AIB [6], or safe parsing may require too much processing power for the constrained devices. When it is possible to force all SIP devices in certain domain use a specific SIP proxy for all SIP communications, such security handlings can be relegated to the proxy from clients.
- Session type procedure: Although some SIP methods, such as INVITE, SUBSCRIBE and REFER create "dialog", which is lasting SIP session, and they require state machine management and additional session refreshment messages, only single transaction type methods like MESSAGE, INFO, and OPTIONS may be enough for some Smart Object communications.
- Out-of-band channel setup: SIP can establish out-of-band media channels, such as RTP, SRTP, and MSRP. This setup is enabled by SDP [7] negotiation in an INVITE transaction, but this is unnecessary for some M2M communications that convey data only in the SIP message payload.

Moreover, incorporating other technologies rather than simply eliminating functionalities from SIP is sometimes useful;

- SIP layer retransmission: Because SIP is able to work on UDP, it has SIP layer retransmission mechanism. If SIP can support UDP-based reliable transport protocol, such as DTLS [8], more than half of retransmission timers will be unnecessary for SIP.

6. Summary and future works

In this paper, we described the requirements of protocol design for the Smart Object, and then indicated how SIP is sufficient for them, and pointed out about lightweightness as the major disadvantage of SIP for them. Then we proposed the simplified SIP approach with some basic policies for determining the way of eliminating the original SIP functionalities to improve the SIP applicability for the Smart Object.

For the future, we should investigate the way of simplifying SIP for some concrete use cases in the Smart Object, with identifying what is and what is not to be eliminated through the process of SIP simplification. Also, we should identify how much improvement of lightweightness is achieved in certain simplification quantitatively, and what sort of simplification is needed to achieve required level of lightweightness.

Appendix I. An example of SIP-based H2M application (remote camera control)

NTT group is providing a remote camera control service, named "Kururimo" [8], to the users of NTT FTTH service, as SIP-based H2M application in Japan. The camera has a telephone number, and when the user originates a call to it from his/her mobile phone, a SIP session is established and the user can check real-time video captured by the camera. Also the user can control angle or zoom of the camera by hitting the keys of phone, and the user can receive an automatic call from the camera when it detects anomaly behavior in its sight.

Appendix II. Fundamental SIP functionalities

Major and basic SIP functionalities which are also valuable as the Smart Object protocol are the followings.

- ✓ Aggregatable: SIP B2BUA (Back To Back User Agent) can aggregate several SIP messages from devices into single SIP message. Also, a SIP-URI can group a number of SIP clients.
- ✓ Bidirectional: Without TCP connection to the server, SIP client can receive incoming messages, queries and push notifications.
- ✓ Discoverable: OPTIONS method and presence mechanism allow querying and discovering capabilities, services, and resources provided by clients and servers.
- ✓ Error tolerant: Loop detection and retry repression mechanisms protect clients and network from meltdown tends to occur easily in automated communications.
- ✓ Mobile: SIP registration with SIP-URI makes communications independent of client's IP address. IP address can be changed during the communication.
- ✓ Minimum configuration: SIP client can connect SIP server without any more information than set of credentials.
- ✓ Secure: Authentication can be done by HTTP digest, AKA, etc. S/MIME and AIB can protect the payload and SIP headers respectively.

References

[1] J. Rosenberg, et al., "SIP: Session Initiation Protocol", IETF RFC3261, June 2002.
[2] Z. Shelby, et al., "Constrained Application Protocol (CoAP)", IETF Internet-Draft draft-ietf-core-coap-04, January 2011. (work in progress)
[3] P. Saint-Andre, "Extensible Messaging and Presence Protocol (XMPP): Core", IETF RFC3920, October 2004.
[4] A. Houri, et al., "Presence Interdomain Scaling Analysis for SIP/SIMPLE", IETF Internet-Draft draft-ietf-simple-interdomain-scaling-analysis-08.txt, August 2009. (work in progress)
[5] P. Saint-Andre, "Interdomain Presence Scaling Analysis for the Extensible Messaging and Presence Protocol (XMPP)", IETF Internet-Draft draft-saintandre-xmpp-presence-analysis-03, January 2008. (work in progress)
[6] J. Peterson, "Session Initiation Protocol (SIP) Authenticated Identity Body (AIB) Format", IETF RFC3893, September 2004.
[7] M. Handley, et al., "SDP: Session Description Protocol", IETF RFC4566, July 2006.
[8] E. Rescorla, et al., "Datagram Transport Layer Security", IETF RFC4347, April 2006.
[9] http://www.ntt-east.co.jp/release_e/0805srqn/hplq080513a_16.html