Webology (ISSN: 1735-188X)
Volume 18, Number 2, 2021

Enhance The Security Of Upnp Group Services Using Group Key Distribution

Rakesh Kumar Khare[1] , Somesh Kumar Dewangan[2] , Rajesh Tiwari*[3]

[1]Associate professor Department of Computer Science and Engineering
SSTC Bhilai, India.
[2]Assistant Professor Department of Computer Science and Engineering
CSIT Durg, India.
[3]Professor Department of Computer Science and Engineering CMR Engineering College
Hyderabad, India.

Abstract :

UPnP architecture is particularly designed for the ease of device discovery and management in automatic home networks, but it lacks a useful feature for supporting dynamic group service utilization over unreliable channels. In response to this limitation, we suggest enhancing UPnP procedures by means of key pre-distribution (KPD) scheme, which has a good compromise between complexity and security. We show how the KPD mechanism can be integrated into UPnP service discovery and utilization processes. This would be contributed to the extensive use of UPnP technology in some applications, where group security is needed.

Key Words: Group security, UPnP networks, Key pre- distribution scheme.

1. INTRODUCTION

Universal Plug and Play (UPnP) architecture [1] works on existing Internet standards and web technologies for enabling seamless proximity networking among networked devices and intelligent home appliances in autonomous fashion. Despite of its dominant features, UPnP specifications have no support of secure group communication among networked UPnP devices, hence trailing significant acceptance in some commercial, medical or military applications, where group security is a prime concern.

In response to the aforementioned inadequacy, variants of public-private key mechanisms have been actively studied for accommodating group security in the UPnP network. However, we are particularly interested in the key pre-distribution (KPD) scheme. Due to its salient feature [2] that trade-offs effectively between security and resource utilization, the KPD scheme looks promising for securing UPnP group services, more than the Public-key Infrastructure (PKI) counterpart. The main contribution of this paper is therefore an efficient group key communication via the KPD scheme that supports well on both secured and non-secured UPnP devices, and also takes the guideline of UPnP Security [3] into accounts. As shown in Figure 1.

The rest of paper is organized as follows. In Section 2, we give a brief overview of KPD group security scheme and automatic processes of UPnP networking. In Section 3, we present our proposed extension

to UPnP processes so that it would support for group security services in the efficient manner, through the notable strength of KPD mechanism. We provide the experiment results in Section 4 and conclude the paper in Section 5.

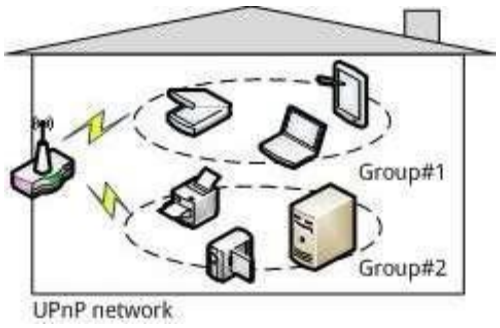

Figure 1. Secure group services in UPnP network.

2. GROUP KEY DISTRIBUTION SCHEMES AND UPNP ARCHITECTURE

• Group Key Distribution Schemes

Secure group communication is about securing group multicast message using the same group key. Distributing the same group key to group member distribution scheme (KDS). Typical KDS is fix-centralized group controller, which identity of fixed group controller is preconfigured to group member. In addition, it is not suitable for UPnP because network environment of group of device is dynamic such as ad-hoc or public location network, which cannot deploy group controller in advance.Other KDS is non-fix group controller, which chooses the group controller from member in the same group. Also this scheme has advantage over fix centralized Units in dynamic environment but requires addition mechanism to verify group membership among any of member node in the group. A number of cryptography schemes seem tobe applicable for group membership verification process, which are

• Public-key Infrastructure (PKI) scheme: Member node stores signed public key called certificate" which can be verified by other node using public key of certificate authority (CA). This technique is strength in term of security but not friendly for resource constraint UPnP device [4].

• Key Pre-distribution (KPD) scheme:

As same as PKI, member node need to be pre-loaded group "key information" generated by trusted authority (TA), which composed of "public identification (PID)" and "secret information (SI)". KPD provides lightweight algorithm that trade-offs effectively between security and resource utilization [2]. In order to generate a pair-wise secret key between any pair of member in the same group by only exchange their PID, which can be applied to verify secure group membership between chosen group controller member and other group member.

UPnP Architecture

Universal Plug and Play (UPnP) architecture enable device to connect without user intervention, it

works on broadcast model for achieving simplicity due to small network size. During UPnP networking processes devices is divided into service provider so called “Device” and client or service invoker so called “Control Point” which one device can be either or both “Device” and “Control Point”.There are 5 main steps for UPnP networking which are discovery, description, control, event notification and presentation.

The foundation of group security processes are group controller discovery, membership verification, rekeying, and ciphering. In order to minimize communication overhead, UPnP networking process should be extended with group key distribution processes as shown in mapping Table 1.

Table 1. Group security and UPnP cooperation

Order	**Group Security Process**	**UPnP Networking**
1	Discover a group controller	Discovery
2	Verify group membership	Control
3	Renking	Eventing and control
4.	Secure group communication	Control

The main reason extending UPnP protocol to support group security is to avoid using proprietary protocol because using standard protocol can achieve ease of development where a number of device manufacturers are different.

3. PROPOSED UPNP EXTENSION FOR GROUP SECURITY

Our proposed framework called SG-UPnP (Secure Group UPnP Security Framework). This section, we will explain system components and sequences in each state start from discovery a group controller, verify group membership and group key distribution (Rekeying). Finally, we will give details of service-level security.

• Architecture Overview

The SG-UPnP framework is designed for small network due to UPnP standard design goal which is for a small network such as Small Office network not enterprise network and devices connection are LAN (Local Area Network) or WLAN (Wireless Local Area Network). System composed of two basic components (depicted in
Figure 2.) as follow:

1) Security Aware Device (SAD): Constructed by extending UPnP Device standard. It is composed of twobasic stacks. First is “Group Controller”, which act as GC in secure group network this module will be enabled only when SAD is selected as GC of its “group id” in GC selection process. Second, a stack layer is “UPnP Device”, which connects between GC and Group Client through UPnP Control Protocol (SOAP). SAD uses modified SSDP protocol as mentioned in previous section inGC selection process.

2) Security Aware Control Point (SAC): Design approach is same as SAD which compose of two stack “SecureClient” and“UPnPControl Point”,first stack is responsible for a “Group member” functional, receive a Group Key then provide to user defined application upon UPnP stack. Second stack layer is defined service based UPnP standard Control Point to support SG-UPnP framework.

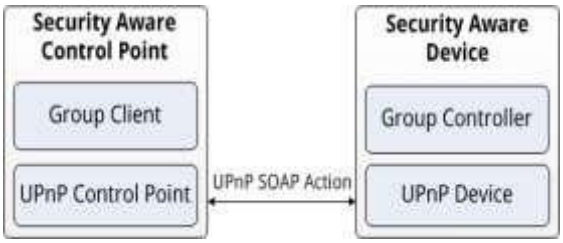

Figure 2. System components.

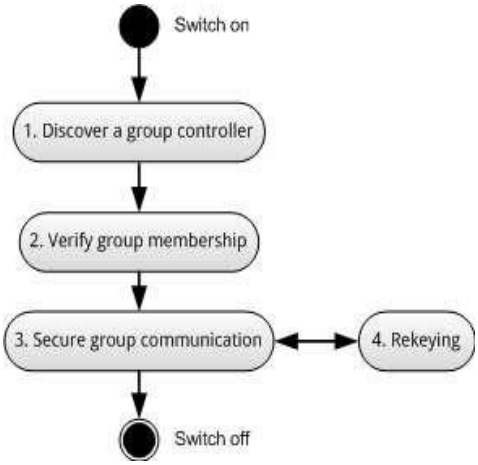

Figure 3. State of Security Aware Control Point.

State of SAC in SG-UPnP framework is as shown in Figure 3, start when it joins group the first state GC need to be discovered. Then SAC needs to verify group membership at GC. After that, in rekeying process new random selected group key will be distribute to all existing member and new comer. Next state is secure group communication if new member join group all member need to be rekey again. Finally, SAC leave group by sending notify message to GC this also need to rekey other existing member.

• Service-level security

In our proposed service-level security architecture as shown in Figure 4, there is a security portal which will design type of service (pre-defined by developer); leads to refinement in the way that services are accessed by Control Point. Security manager deals with secured UPnP service by wrapping and unwrapping SOAP action using corresponding group key at both SAC and SAD. Proposed framework avoids using proprietary protocol because it is designed for UPnP device which will be developed by many different manufacturers using existing standard protocol is easier for developer to develop application than has to learn all new protocol. Thus, we implemented service called "Device Security" based on UPnP Standard as a communication protocol between SAC and SAD. The different between our approach and public key based security service from UPnP security ceremonies [3] are following:

1) Supported secure group communication mechanism including membership verify and join/leave rekeying.

2) Key pre-distribution based, which has to design all new action prototype and sequence of communication. Typical UPnP service can be access by all Control Point in network. In our proposed service-level security architecture as shown in Figure 4, there is a security portal which will design type of service (predefined by developer), leads to refinement in the way these services can be accessed by Control point. Security manager deals with secured UPnP service by wrapping and unwrapping SOAP action using corresponding group key at both SAC and SAD.

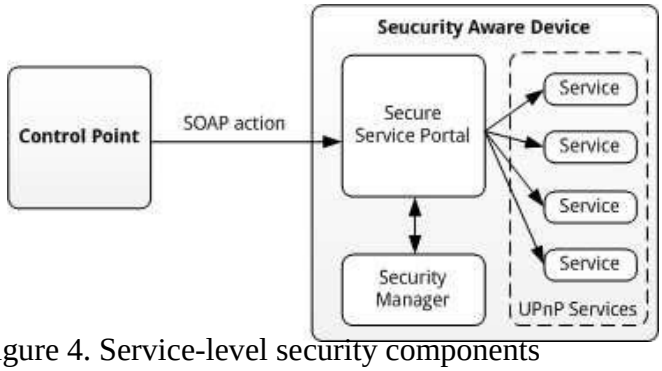

Figure 4. Service-level security components

• Offiline key pre-installation

The main reason for replacing PKI technology with KPD technology in member verification process is constraint of UPnP device resources KPD is mainly based on Blom's symmetric key generation algorithm[5]. Briefly explain, there are two steps install key information file on each of corporate devices after that any pair of prepared devices will able be used to verify member later are as follow:

Step 1: Installing individual <key information iles>, composed of "public identification (PID)" and "secret information (SI)" on all of corporate devices. This process can be done by Trusted Authority (TA) and performed by system administrator in practical. All key information files are computed by TA at ones called key information pool this pool is strict to large enough for amount of devices that will used in corporate. Larger amount of device than key information pool size required to regenerate and install <key information file> to all existing devices.

Step 2: Those prepared devices can compute pair wise secret key between any pair of them by only exchange their PID. After that each device can compute pair- wise share secret key by operate other PID with its own SI.

• Group controller selection process

Important reason considered to choose GC from existing group member instead of use fixed centralized GC is to resolve dynamic secure group deployment problem, which is corresponded to nature of UPnP network that group of users can move to conference on any public network location. In this case, it is complicated to install fixed centralized GC.

Proposed algorithm consists of two phases: group initialized and group maintenance. We will explain each phase from view of device and control point respectively.

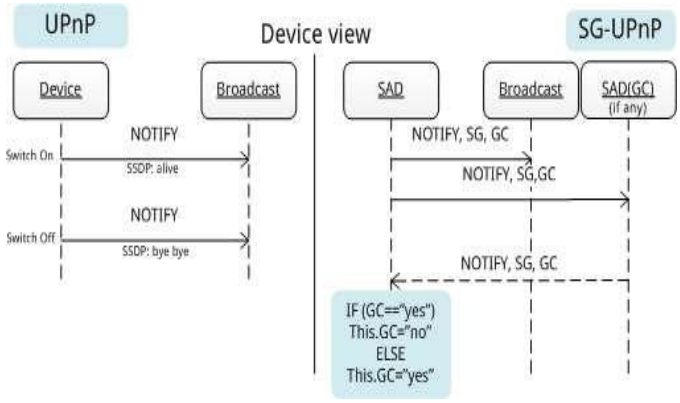

Figure 5. UPnP discovery process of SG-UPnP

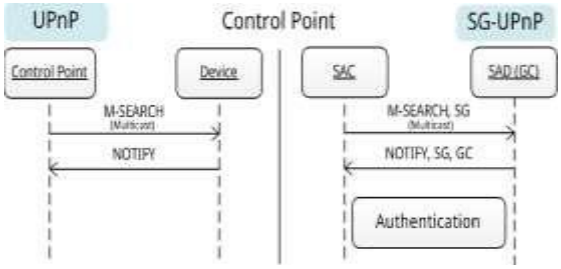

Figure. 6 (a) GC selection process (b) GC discovery process

First, group initialized process is depicted in Figure. 6 (a), UPnP discovery standard is on the left. After device is switched on it sends multicast message NOTIFY with “ssdp: alive” or “ssdp: bye bye” (see Listing 1.) when it logs off. Thus, other device can know whether device online or not. On the right side is a modified version for SG-UPnP framework, its processes are as follow:

Step 1: SAD sends multicast modified NOTIFY message (see Listing 2.) with extra headers SG (secure group id) and GC (is this a group controller), which are “SG: <secure group id>” and “GC: yes”.

Step 2: If GC corresponds to <secure group id> exist it will responses NOTIFY message with “GC: yes” to newcomer (new SAD). Otherwise, new comer will be a Group Controller.

In Figure 6 (b) when Control Point just connected to the network it sends multicast M-SEARCH messages to discover the device in the network which is responded by a NOTIFY message from the device. Process of SG-UPnP is as follows:

Step 1: SAC sends modified M-SEARCH message with extra header SG: <secure group id>. ere, SAC can discover both the original UPnP device and SAD at the same time.
Step 2: Thus, if there is a GC in the network SG can determine by SG and GC headers, which should be "SG: <same group id>" and "GC: yes". Later, SAC can verify group membership to GC.

```
NOTIFY * HTTP/1.1
HOST:239.255.255.250:1900
CACHE-CONTROL:<max-age>
LOCATION:<url of device
description> NT:<search target>
NTS:<ssdp:alive/ssdp:bye bye>
```

Listing 1: The UPnP NOTIFY message

```
NOTIFY * HTTP/1.1
HOST :239.255.255.250:1900
CACHE-CONTROL:<max-age>
LOCATION :<url of device description>
NT<search target>
NTS:<ssdp:alive/ssdp:bye bye>
SG:<device secure group id>
GC:<yes/no>
```

Listing 2: The SG-UPnP NOTIFY message

Second, group maintenance process occurred when GC will logout from network or it failed, first case it can notify to other member node in the network to select the new GC but second case member node need to check if GC is not work anymore. Our framework deal withbothcases as follow: Case 1: SAD notifies to other using NOTIFY message with header "ssdp: bye bye". After that GC selection process will start over again, existing SAD need to wait for random time to send multicast NOTIFY request to be a GC.
Case 2: Lease time for NOTIFY message fromSAD was defined, when it expired GC need to sends NOTIFY message again, so that other SAD can check if GC failed and determine to start GC selection process over again.

• Membership verify (Authentication)

There are three main processes, SAC verify its membership to GC as follows:
Step1: Exchange "public identification (PID)": SACrequest "GetPublicIdentification" action with its

own PID as input parameter, then GC sends its PID back. Both sides can compute pair-wise secret keys by using the other's PID and its own "secret information (SI)", this should get the same result to successfully verify membership. Otherwise, <key information file> of which was from a different key information pool and resulted in membership verification failure.

Step 2: SAC requests LifeTimeSequenceBase (LTSB) state variable which is state variable of SAD. It will change every time it is read. Freshness of authentication message is checked by LTSB to protect from replay attack. SAC can get LTSB by requests "Get Lift Time Sequece Base" action from GC.

Step 3: Member verification: SAC requests authenticate" action from GC. There are three input parameters: SID, LTSB and Auth Secret, which is the hash result of "LTSB" and "pair-wise secret key". GC will check

LTSB first if it's up-to-date and then compare the hash result with its own to authorize group membership of SAC.

• Join rekey process

Rekey process is important to protect forward/back secrecy. Basically, join rekey can multicast new Group Key to existing member by encrypted using old Group Key but leave rekey will need to encrypted new Group Key by pair-wise secret key and send to each member, this is not scalable for large group member size, so numbers of key management scheme had been proposed [6]. But our UPnP environment is a small network, according to small size of network, current work we picked simple rekey scheme to simply demonstrate secure group communication for UPnP environment. GC random selects new Group Key and distributes to existing member using multicast message. Since efficient UPnP event notification exist, we modified it to support multicast notification to distribute new Group Key encrypted with old Group

Key. Processes are simply as follow:

Step 1: SAC sends a modified SUBSCRIBE multicast request to GC after verified membership to subscribe "GroupKeyEncrypted" state variable.

Step 2: When new member joined GC random select new Group Key and encrypts it with old Group Key. Then save as "GroupKeyEncrypted" state variable. Since this is set to event notify state variable, the new value of GroupKeyEncrypted" will be multicast to the subscriber at once. However, the rekey process requires reliable multicast communication, in case a member does not receive just only one rekey message in sequence it needs to rejoin the secure group again because it will not be able to decrypt any further rekey messages.

• Leave rekey process

Multicasting new Group Key encrypted with old Group Key does not work when member leave because leaving member has old Group Key. To protect forward secrecy GC must encrypt new Group Key with pair-wise secret key between each existing member. By applying UPnP original event notification architecture each member has to subscribe "GroupKeyID" state variable which

represent version of current group key, when rekey process occurred each of member will be notified. Then, they will retrieve new group key via “GetGroupKey” action.

4. EXPERIMENTAL RESULT

We implemented a prototype of SG-UPnP framework on the base of UPnP device development library called CyberLink for Java [7] and common Java Cryptography Architecture (JCA). Our experimental environment uses two conventional personal computers connected through local area network where virtual UPnP device and control point run on Eclipse platform. We implemented a simple “DeviceSecurity” service follow our proposed framework and performed comparison of processing time used in membership verification process between two cryptography mechanisms: PKI and KPD. The results are shown in the table. KPD scheme gives significant faster response time than PKI.

Process	SAD (msec)		SAC (msec)	
	PKI	KPD	PKI	KPD
SAC group membership verification	1.2830	0.1539	0.5631	0.0624
Freshness verification	1.2762	0.3022	9.5317	0.1670
Secure channel used to distribute a group key	3.9565	0.5400	16.3263	0.2580

There are very few published works attempting to enable the support of group communication in UPnP networks. Lee et al. [8] proposed a secure-UPnP framework (SUPnP). The objective is to provide exclusive channels for secure group communication, besides standard UPnP channels for normal communication. Compared to our work, this work aresignificantly different as described in the following.

• It demands a new protocol to be devised for several purposes, such as during the establishment of secured communication channels, or in cooperative works with a special group controller. Unlike in our work, there is no extra protocol to be involved, since minor modification of some standard UPnP protocols are considered.

• It works on the trusted-server scheme, so facing inherent problems, such as the single point of attack problem at centralized group controller. However, due to application of Blom’s scheme used in member verification process, such those problems can be avoided in our work since a group controller can be chosen from group member in KPD scheme[9-11].
• It can provide only the link-level security, where the same security policy can be only applied for all services carrying over the secured link. In contrast, our work can offer the service-level security for group services over unsecured link, therefore supplementing the traditional services found in UPnP networks [12-13].

Regarding works applying the key p redistribution (KPD) technique for constraint resource

environments, Ramkumar et al. [2] showed how the KPD algorithm should be modified to cope with the usage from alarge number of devices, such as in the wireless sensor networks. However, it is not the case for UPnP networks, where available nodes are quite limited due to the addressing capability within a locally single network, according to the standard UPnP architecture.

5. CONCLUSION

We have presented the SG-UPnP framework that enables UPnP protocol to support secure group communication. Its distinctive points are:

(1) It makes interoperability between the KPD scheme and UPnP protocols (discovery, control, and eventing) possible,

(2) It supports dynamic deployment without user intervention required.

(3) It has a backward compatibility with existing UPnP standard due to the design which based on service- level. However, group security pre-configured and the member is fixed. So, it is unable to communicate between different groups. Future work will consider this issue.

REFERENCES

[1] Kim, D., Song, H., Lee, K., & Sung, J. (2005). UPnP-based sensor network management architecture. In Second International Conference on Mobile Computing and Ubiquitous Networking (pp. 0-0). ICMU.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Ramkumar, M., & Memon, N. (2004, June). On the security of random key pre-distribution schemes. In Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004. (pp. 153-160). IEEE..

[3] Ellison, C. (2006). UPnP security ceremonies design document: For UPnP device architecture 1.0, 3 October 2003

[4] Xu, S. (2007). On the security of group communication schemes. Journal of Computer Security, 15(1), 129-169.

[5] Blom, R. (1985). An optimal class of symmetric key generation systems. In Advances in Cryptology: Proceedings of EUROCRYPT 84 A Workshop on the Theory and Application of Cryptographic Techniques Paris, France, April 9–11, 1984 3 (pp. 335-338). Springer Berlin Heidelberg.

[6] Rafaeli, S., & Hutchison, D. (2003). A survey of key management for secure group communication. ACM Computing Surveys (CSUR), 35(3), 309-329.

[7] Konno, S. (2005). CyberLink for Java Development Package for UPnP Devices.

[8] Lee, J. J., Huang, C. Y., Lee, L. Y., & Lei, C. L. (2007, April). Design and implementation of secure communication channels over UPnP networks. In 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07) (pp. 307-312). IEEE.

[9] Guo, X., & Li, J. (2013, October). Secure upnp services based on group signature algorithm. In Proceedings of 2013 3rd International Conference on Computer Science and Network Technology (pp. 951-955). IEEE.

[10] Kayas, G., Hossain, M., Payton, J., & Islam, S. R. (2021). SUPnP: Secure Access and Service Registration for UPnP-Enabled Internet of Things. IEEE Internet of Things Journal, 8(14), 11561-11580.

[11] Zhu, H., & Zhu, Y. (2012). A kerberos-based upnp exten-tion for secure home networks. In Proceedings of 4th Inter-national Conference on Computer Engineering and Technology (pp. 104-108).

[12] Khan, R., & Khan, S. U. (2017). Design and implementation of UPnP-based energy gateway for demand side management in smart grid. Journal of Industrial Information Integration, 8, 8-21.

[13] Williams, P., Dutta, I. K., Daoud, H., & Bayoumi, M. (2022). A survey on security in internet of things with a focus on the impact of emerging technologies. Internet of Things, 19, 100564.

.