

Implementation of Service Location Protocol and Remote Device Control for IPv6 based Home Networking

Hyun-Wook Cha, Jung-Soo Park, Jae-Hoon Jeong, Hyung-Jun Kim
 Protocol Engineering Center, ETRI, 161 Gajong-Dong, Yusong-Gu, Daejeon, Korea
 E-mail : {jafy, pjs, paul, khj}@etri.re.kr

Abstract — We implemented IPv6 based service location protocol (SLPv6) and Remote Device Control (RDC) for IPv6-based home networking. Through SLP whis is an IETF standard (SLPv2 : rfc 2608), various users' devices can be attached to their home networks dynamically. And users can control remotely these home devices using RDC which is an ITU-T standard (H.282). Our implementaion consists of three parts : an residential gateway, SLP enabled service devices, and their clients. On the gateway, a SLP daemon as DA (Directory Agent) is runnging and web server, mysql daemon also running as the interface with users. SLP enabled devices are controllabile through RDC. Currently, we implemented a SLP enabled camera server as a service. In our testbed, we use NAT-PT called 6TALK made by ETRI to enable users in any distant IPv4 based network to control devices in the testbed. In this paper, we describe our implementation and demonstration results.

Keywords — Service location protocol, remote device control, home networking, IPv6.

1. Introduction

The need for home networks is growing at a rapid rate. As many service devices are attached to home networks, efficient automatic service discovery methods are needed. In recent years, competing industries and standards developers have proposed their solutions including Jini [6], Universal Plug and Play (UPnP) [4], Salutation [5], and Service Location Protocol (SLP) [1]. Among them, SLP is an IETF (Internet Engineering Task Force) standard and provides decentralized, lightweight, and extensible service discovery. We implemented IPv6 based SLP by modifying OpenSLP [9] codes to apply to IPv6 based home networks. And we also implemented Remote Device Control (RDC) protocol which is an ITU-T standard (H.282) [10] for a home application. In this paper, we describes implementation details and demonstration results. In section 2, overview of SLP including its modification for supporting IPv6 and RDC are are presented. In section 3, we describe our implementaion. In section 4, organization of our testbed and demonstration results are presented. Finally, we conclude in section 5.

2. Backgrounds

2.1 Service Location Protocol (SLPv2)

SLP provides a framework for resource discovery that includes three “agents” that operate on behalf of the network-based software:

- User Agents (UA) perform service discovery on behalf of client applications.
- Service Agents (SA) advertise the location and attributes on behalf of services.
- Directory Agents (DA) aggregate service information into what is initially a stateless repository.

SLP has two modes of operation.

- When a DA is present, it collects all service information advertised by SAs, and UAs unicast their requests to the DA. Figure 1 illustrates this case of operation.
- In the absense of a DA, UAs repeatedly multicast the same request they would have unicast to a DA. SAs listen for these multicast requests and unicast responses to the UA if it has advertised the requested service. Figure 2 shows this case of operation.

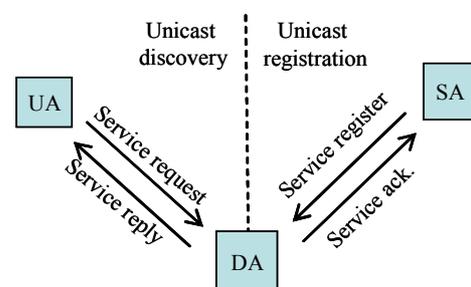


Figure 1. Normal operation of SLP. The SA registers a server's location with the DA, and the UA obtains the server's location from the DA via a Service Reply message in respense to the UA's Service Request message.

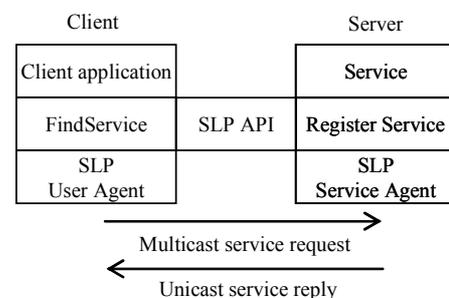


Figure 2. Service discovery using SLP without DAs. The client application uses a UA to multicast a Service Request that the SA

Modifications to support IPv6 is specified in rfc 3111.[2] Since IPv6 provides different scopes for interface address configuration and multicast addresses, SLPv2 scoping rules for IPv6 are defined.

- Joining SLPv2 Multicast Groups : a SLPv2 Agent MUST NOT join a multicast group which has greater scope for an interface than it is configured with for use with unicast.
- Sending SLPv2 Multicast Messages : An SLPv2 agent MUST issue this request using a source address with a scope no less than the scope of the multicast group.
- Rules for Message Processing : A SLPv2 SA or DA MUST NOT respond to a SrvRqst with a service: URL for a service with an address scope less than the request's source address scope.

SLPv2 for IPv6 uses the following multicast group-id assignments:

- FF0X:0:0:0:0:0:116 : SVRLOC
- FF0X:0:0:0:0:0:123 : SVRLOC-DA
- FF0X:0:0:0:0:1:1000 - FF0X:0:0:0:0:1:13FF : Service Location

2.2 Remote Device Control (RDC)

ITU-T H. 282 defines a set of services and a protocol which may be used to perform remote device control in a multimedia conference. Services which RDC should provide are categorized as follows:

- to obtain attributes of devices (DeviceAttribute)
- device control (DeviceControl)
- source selection service allows a node to request that a particular source be connected to a specific output stream (DeviceSource)
- to inquire of the status of a specific remote device (DeviceStatus)
- to be notified of event changes occurring on a remote device (DeviceEvent)

3. Implementation

Our implementation includes SLPv6, RDC client/server.

For SLPv6, OpenSLP which is currently based on IPv4, is modified to support IPv6. To do this, it is necessary to change IPv4 sockets in codes into IPv6 ones. And we also considered rfc 3111. Like OpenSLP, SLPv6 is also implemented to be compiled on both LINUX and windows platform. SLPv6 has same organization as OpenSLP: SLP library, SLP daemon which can operate as any SLP agent, and slptool which is a utility program using the SLP library. In addition to original specification, the SLP daemon of SLPv6 has mysql database update functionality for remote users to access and control some device in our home network testbed through web and cgi like PHP and Perl.

As for RDC, we implemented DeviceAttribute, DeviceStatus, DeviceControl among defined RDC services. When a RDC server receives a RDC message from a RDC client, the server decodes the message and sends

corresponding command to the camera device via a serial port. Captured pictures from the camera are compressed by H. 263 video codec and sent to rdc client. We used codec made by Telenor Research and Development, and university of british columbia. And RDC server registers its service to DA by calling SLP library function to make its service automatically discovered. When it registers its service, it uses its domain name as the service URL because any remote user on the IPv4 based network can access its service. For this purpose, our testbed leverages DNS-ALG of 6TALK.[7][8]

Currently RDC client only works on the windows since it was implemented using ActiveX technology to be downloaded and executed dynamically in a web page.

4. Demonstration

Figure 3 shows the organization of the 6SEED testbed.

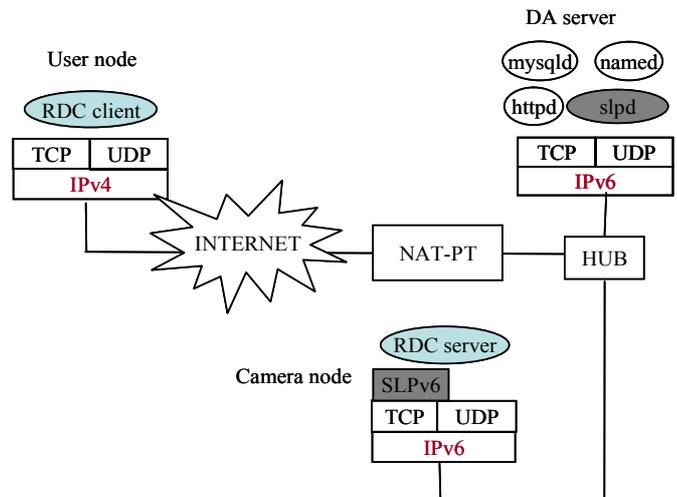


Figure 3. Organization of 6SEED Testbed. Circle means a processe.

The 6SEED testbed is organized three hosts and a 6TALK (NAT-PT) and a hub. As soon as RDC server on camera node begins to run, it registers its service URL "service:rdc.etri://cam.6talk.net" to DA. DA inserts this new service into the it own database as well as the database of mysql. After that, any remote users can discover and access this service by web browsing. When a user choose this camera device for his or her service, RDC client is downloaded into the user host and loaded in new web page ,and executed.

Figure 4 show web pages through which a user can communicate with DA server and find available RDC server which is pre-registered and control chosen remote device.

5. Conclusion

We implemented SLPv6 and RDC protocols to apply to IPv6 based home networking. As an application for our testbed 6SEED, we also implemented a part of RDC server/client to control remote camera device. In the 6SEED testbed, any SLP enabled services can be attached and discovered automatically by clients. Currently we only considers wired home networks. Issues accompanying extension of our implementation for

wireless networks including mobile ad hoc networks will be addressed in future works.

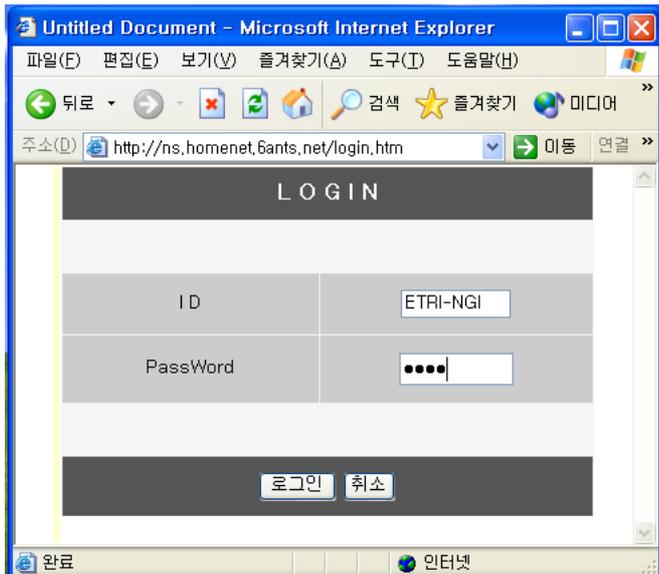


Figure 4.a First web site which a user visits to control SLP-enabled devices in the 6SEED testbed.

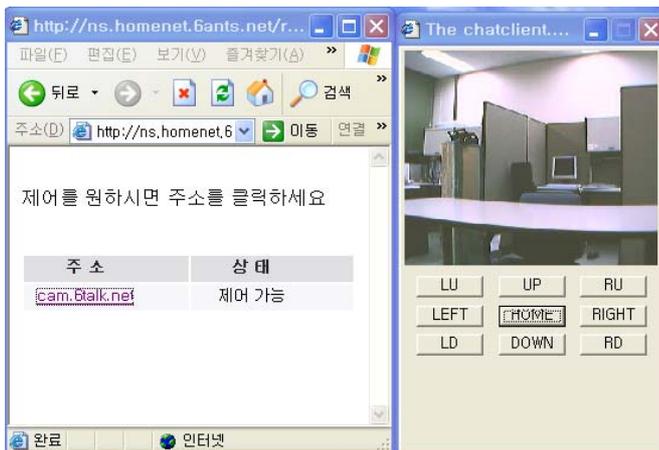


Figure 4.b Left web page shows an available device. When user choose the device, right web page with rdc client program embedded is pop-upped.

REFERENCES

- [1] E. Guttman, C. Perkins, J. Veizades, and M. Day, "Service Location Protocol, Version 2," IETF, RFC 2608.
- [2] E. Guttman, "Service Location Protocol Modifications for IPv6," IETF, RFC 3111.
- [3] HAVI. *Home Audio Video Interoperability network*. <http://www.HAVI.org>
- [4] UPnP. *Universal Plug and Play*. <http://www.UPnP.org>.
- [5] Salutation. *Salutation*. <http://www.salutation.org>
- [6] Jini. *the Jini Community*. <http://www.Jini.org>
- [7] Rfc 2694, DNS extensions to Network Address Translators (DNS_ALG)
- [8] 6TALK, ETRI, <http://www.6talk.net>
- [9] OpenSLP, <http://www.openslp.org>
- [10] "Remote Device Control Protocol for Multimedia Applications", ITU-T, H.283