Applying Kerberos to the Communication Environment for Information Appliances

Shoichi SAKANE* Nobuo OKABE† Ken-ichi KAMADA‡ Hiroshi Esaki§
Yokogawa Electric Corporation The University of Tokyo

Abstract

When IPv6 deploys, each information appliance shall have a global IP address and communicates directly with each other. Some devices may have much lower processing performance than PCs have due to various limitations (e.g. cost, physical size, power consumption). Such devices must have security function, that is confidentiality, integrity and access control, for provision of privacy even with a home networking environment. The information appliances shall move around the global network with the users. In this paper, we assume these devices are used in the home and we describe the methodologies to achieve access control using Kerberos and to deal with changes of IP addresses using modified Kerberos. IPv6 has a security mechanism called "IPsec" for secure communication. In order to use the IPsec, peering communicating devices have to share a symmetric key to maintain the confidentiality and/or the integrity. We also describe a method that these restricted devices can share a symmetric key securely.

1. Introduction

As the Internet technology is going to be widely deployed, networking capability will be implemented even on various non-PC appliances (called as "device(s)" in this paper) as well as conventional PCs. In this paper, we describe the issues which occur when such devices are used in the home network and describe the methodology to solve these issues.

*Shouichi.Sakane@jp.yokogawa.com, Network and Security Engineering Center, Yokogawa Electric Corporation. 2-9-32 Nakacho, Musashino-shi, Tokyo, 180-8750 Japan.

†nov@tahi.org, Network and Security Engineering Center, Yokogawa Electric Corporation. 2-9-32 Nakacho, Musashino-shi, Tokyo, 180-8750 Japan.

‡kamada@hongo.wide.ad.jp, Graduate School of Information Science and Technology, The University of Tokyo. 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8654 Japan.

§hiroshi@wide.ad.jp, Graduate School of Information Science and Technology, The University of Tokyo. 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8654 Japan.

It is apparently valuable that various devices are connected by network and cooperate with each other. Many research projects [13, 2] focus on it. Such applications will challenge a set of new requirements to us. Particularly, the accessibility control of each devices from within a home and from/to the global Internet space must not be restricted by network topology. In order to achieve this requirement, each device has to be equally able to initiate the communication to the other, and they must have their own global IP addresses. Consequently, a large address space is necessary to allocate them their own global IP addresses. This means that IPv6 should be important infrastructure for the purpose because of its large address space [3].

This paper proposes the system architecture using the Kerberos [9], in order to fulfill the requirements discussed above. Section 2 discusses the detailed requirements of the discussing network environment, section 3 describes the mechanism of access control using the Kerberos, section 4 describes the mechanism to resolve the peering IP address using the Kerberos, section 5 describes the mechanism to establish the secured communication channel among devices, section 6 discusses the further study items and section 7 summarizes this paper.

2. System requirements

When the devices directly and transparently communicate to each other, the system must fulfill, at least, the following security requirements.

(1) Access control

From the privacy and secured operation points of view, the system must achieve the access control between devices explicitly identifying each device. For example, in the case of VCR player and it's controller, only the VCR controller corresponding to VCR player can access the VCR player. In other words, the VCR controller owned by or operated by the non-family member must not access the VCR player owned by a particular family. Non-family member could be a guest visiting the home, could be a neighbor, or could be a user accessing through the Internet. For example, in the case of guest, the access rights for the guest usually should

be expired within a determined period of time, though that for the family member would not.

(2) (Secured) IP address resolution for Plug and Play devices

The Plug and Play feature [15] of IPv6 can automatically generate the IP addresses of devices, when they connects to the network.

Users feel comfortable with this function because easy setup for network access. However, it means that the device must resolve the autocofigured (not manually configured) IP addresses of peering device. The devices are not only fixed devices, but are the nomadic devices. We have to assume that the nomadic devices move around the global Internet space with owners. MIP6 (Mobile IPv6) [5] can be one of solutions to fulfill this requirement. However, we do not depend on MIP6 at this moment because its specification is not stable yet.

From the view point of security consideration, this address resolution procedure must be performed securely. To achieve these requirements regarding the access control in this environment, address-independent identifier is necessary. We propose to use a new identifier except an IP address, in order to identify the devices even if their IP addresses are assigned automatically.

(3) Secured communication

The secured communication is mandatory to preserve users' privacy. IPsec [7] can be used to fulfill this requirement and it provides confidentiality and integrity. In order to apply the IPsec, the peering communicating devices have to share a symmetric key. IETF IPSEC working group [4] has standardized IKE [1], a key exchange mechanism with mutual authentication based on cryptographic technology for that purpose. However, that is not suitable for the cost-sensitive or physically-restricted devices because they only have less computing power than the normal PCs. We have experienced a case that it took several tens of seconds for SSL handshake when SSL web server is operated upon 8-bit CPU in our in-house test environment. Because of this observation, we may have to assume that the devices could not run the public key cryptography in reasonable time. This means that we would need to define the other key exchange mechanism, which is suitable for these devices.

3. Access control using Kerberos

In this paper, as a preliminary consideration, we propose a mechanism of access control with device-by-device. Finer-grain access control, that controls "which user can access which data object via which application", should be treated in the application layer. This finer-grain access control is not the scope of this paper, but is a candidate for future study. With our proposed method, devices can obtain the access policy rule of a peering device. This mechanism can also applied to the fine-grain access control described above, as well as device-device access control.

Since Kerberos maintains the information of entities on a central server, it is often said that a scalability issue exists for handling a large number of entities, such as a large enterprise. However, the number of devices in the home network is small enough to be managed by a single server. Additionally, there would be a few people (e.g., parents) who is responsible to define the security policy of the home networking environment. As a result, in the home network discussed in this paper, the centralized management model based on Kerberos could be appropriate.

The access control proposed in this paper is achieved by the following manner. Devices can communicate with each other if and only if they are mutually authenticated using Kerberos. Each member of a family has his/her own realm. A device, to which some person want to access, must be registered in a corresponding realm, i.e. a unique principal and a secret key must be allocated and registered. By this approach, his/her controller can mutually authenticate with the device using a appropriate principal, so as to achieve the appropriate access control.

For example, supposing that a family consists of two persons; A and B. There are CON-a and CON-b (TV controllers), TV, and VCR. A and B is the owner of CON-a and CON-b respectively. TV can be accessed by both A and B. VCR can be accessed only by A. Here, REALM-a and REALM-b are defined as realms of A and B. CON-a, TV, and VCR belongs to REALM-a. CON-b and TV belongs to REALM-b. Figure 1 shows the example.

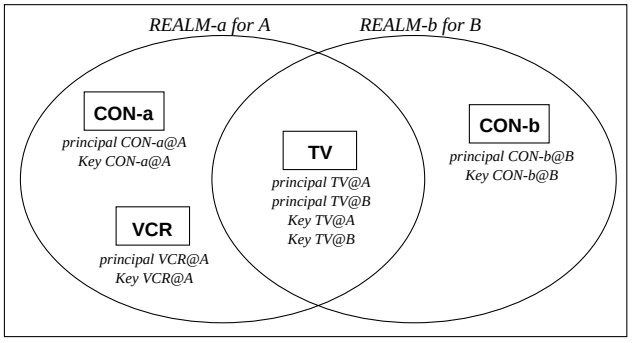

Figure 1. Mapping between family members/devices and realms/principals

A device, which belongs neither realm (i.e. owned by other person O), can not mutually authenticate with these devices, so that it can not establish any communication.

Next, let us consider a case where a guest is allowed accessing the TV with his/her controller CON-g within a limited time. In this case, REALM-g is defined as a realm for guests like figure 2 and a principal and a secret key are installed in TV in advance. When guest (G) visits a home

network, a new principal and a new secret key which belong to REALM-g is allocated to CON-g. Now, CON-g and TV both belong to REALM-g, so they can initiate a connection to each other. The period in which G can use TV is controlled by setting a expire time on Tickets in REALM-g.

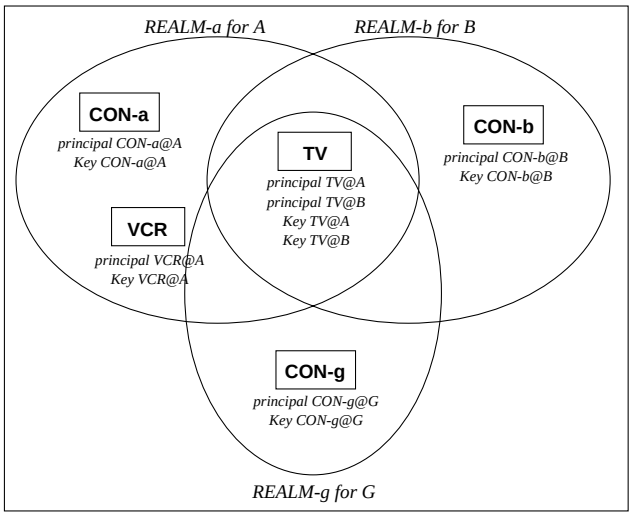

Figure 2. Realm for guests

4. IP address resolution using Kerberos

The Plug and Play feature of IPv6 can eliminates a bothering procedure of setting up IP addresses for the end users. However, with the IPv6 stateless autoconfiguration, devices can not know the IP addresses of corresponding peering devices, in advance. Also, the nomadic devices changes their IP addresses frequently. A solution to this issue using Kerberos is proposed below.

In Kerberos, there is a message called KRB_AS_REQ, which is used by principals to be authenticated by Authentication Service (AS) of Key Distribution Center (KDC). In this message, there is an optional field containing IP addresses of the principal.

Figure 3 shows the case when device B connects to the network. Device B is allocated an IP address, sends a KRB_AS_REQ message and registers its IP address to the KDC. The KDC authenticates B, and caches the IP address and sends back a KRB_AS_REP for indicating the success of registration.

Figure 4 shows the case when device A is initiating a communication to B. In this case, device A requests a Ticket for B to KDC with KRB_TGS_REQ message. KDC returns B's IP address, which has been cached in it, with KRB_TGS_REP. With this procedure, A can obtain currently registered IP address of B.

For implementing the above scheme, we need following three modifications to Kerberos.

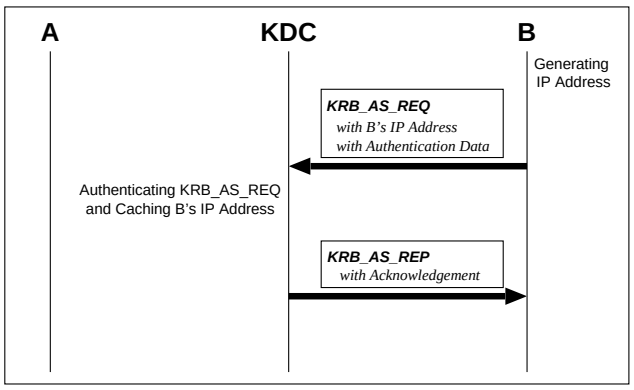

Figure 3. IP address registration with Authentication Service Exchange

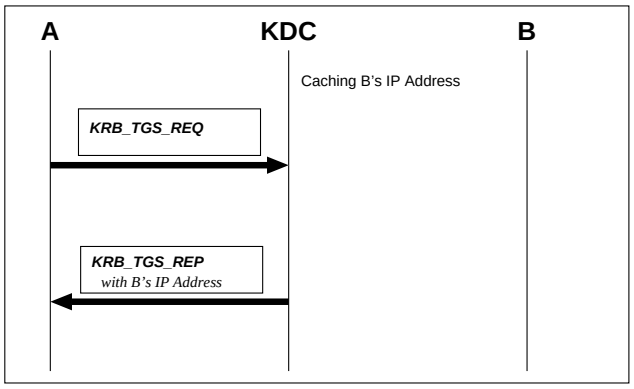

Figure 4. IP address retrieval with Ticket-Granting Service Exchange

(1) A field containing IP addresses in KRB_AS_REQ

This field is optional. In our proposing system, this option should be mandatory.

(2) Pre-authentication field

This field is optional. In our system, using this field, KDC authenticates a device sending the KRB_AS_REQ. KDC has to cache the IP addresses in KRB_AS_REQ and to associate them with the corresponding senders.

(3) A field containing IP addresses in KRB_TGS_REP

There is no address field defined in KRB_TGS_REP message. Therefore, new field has to be defined for carrying IP addresses of a peer device.

5. Setting up secure communication using Kerberos

In IPv6 system, the implementation of IPsec is mandatory. IPsec provides confidentiality and integrity to an IP packet with IP Encapsulating Security Payload [6] and IP

Authentication Header [6]. Confidentiality is achieved by encrypting the payload of a packet with symmetric cryptography. Integrity is achieved by signing the whole packet with HMAC [10]. Both are relatively easy to perform on devices which have low computational capability such as information appliances.

A mechanism to share a symmetric key of IPsec between two communicating devices is usually based on mutual authentication using asymmetric cryptography, like IKE. But, it would not be suitable for devices which have insufficient computational capability, which are the target of this paper. On the other hand, IETF KINK working group [8] is examining other lightweight key exchange mechanism called KINK (Kerberized Internet Negotiation of Keys) [14]. KINK defines the key exchange mechanism for IPsec using symmetric cryptography with Kerberos. Therefore, with KINK, the required computational cost should be significantly lower compared to the conventional key exchange protocol using asymmetric cryptography.

6. Future study items

The access control method described in this paper is based on the authentication between devices in the same realm. When considering IP phones, we must take care of the different communication model. Those devices belong to only each individual, but need to communicate with other IP phones. In our access control model, the devices that belong the different realms must communicate to with inter-realm authentication.

Devices must have a clock which is synchronized with other devices and KDC to use Kerberos authentication. But, devices cannot always have accurate clocks for the assumption of Kerberos by various reasons. The devices' clocks will drift freely if the devices do not have any adjusting mechanism. The clocks will have significant delay if the devices can only be given power when using and if their clocks are not backed up by batteries. The clocks will move forward/backward suddenly if the devices adjust their clocks periodically via NTP [12] or so. Consequently, an alternative authentication method, which does not depend on synchronization of clocks, should be considered.

In this paper, it is assumed that the initiator already knows the principal of the responder when it wants to establish a communication. But, it is inconvenient for users to handle the principal directly. A mechanism for mapping user-defined names to principals of Kerberos is necessary.

We described that the computational cost of asymmetric cryptography is not suitable for information appliances in this paper. Some quantitative evaluation is necessary to show and prove to what extent the symmetric cryptography is more suitable in these network environment.

KINK reuses the ISAKMP [11] format in a message to exchange several data including a symmetric key for IPsec. ISAKMP format is designed for general key exchange mechanism. There is some room to develop more simple format and improve efficiency for information appliances with limited functions or purposes.

7. Conclusions

When IPv6 is widely deployed, a new communication model is expected; various devices have their own global IP addresses and directly communicate with each other.

In this paper, we focused on information appliances and proposed the following three mechanisms; (1) an access control mechanism using Kerberos, (2) an address resolution using Kerberos with modification, and (3) a method to establish secure communication using Kerberos.

References

[1] D. Harkins and D. Carrel. The Internet Key Exchange (IKE). RFC 2409, 1998.

[2] HAVi organization. HAVi: Technical Information - White Paper. Hypertext document, http://www.havi.org/techinfo/whitepaper.html.

[3] R. Hinden and S. Deering. IP Version 6 Addressing Architecture. RFC 2373, 1998.

[4] IP Security Protocol (ipsec). Working group charter, http://www.ietf.org/html.charters/ipsec-charter.html.

[5] D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. draft-ietf-mobileip-ipv6-18.txt, 2002.

[6] S. Kent and R. Atkinson. IP Encapsulating Security Payload (ESP). RFC 2406, 1998.

[7] S. Kent and R. Atkinson. Security Architecture for the Internet Protocol. RFC 2401, 1998.

[8] Kerberized Internet Negotiation of Keys (kink). Working group charter, http://www.ietf.org/html.charters/kink-charter.html.

[9] J. Kohl and C. Neuman. The Kerberos Network Authentication Service (V5). RFC 1510, 1993.

[10] H. Krawczyk, M. Bellare, and R. Canetti. HMAC: Keyed-Hashing for Message Authentication. RFC 2104, 1997.

[11] D. Maughan, M. Schertler, M. Schneider, and J. Turner. Internet Security Association and Key Management Protocol (ISAKMP). RFC 2408, 1998.

[12] D. L. Mills. Network Time Protocol (Version 3) Specification, Implementation and Analysis. RFC 1305, 1992.

[13] Sun Microsystems, Inc. Jini Architecture Specification Version 1.2, 2001. Hypertext document, http://wwws.sun.com/software/jini/specs/jini1.2html/jini-title.html.

[14] M. Thomas and J. Vilhuber. Kerberized Internet Negotiation of Keys (KINK). draft-ietf-kink-kink-03.txt, 2002.

[15] S. Thomson and T. Narten. IPv6 Stateless Address Autoconfiguration. RFC 2462, 1998.