Combining Trust Management, Jini, IPv6, and Wireless links: A Proposal for a Service Network Architecture for Ad Hoc Environments Extended Abstract

Pekka Nikander

Helsinki University of Technology
pekka.nikander@hut.fi

1 Introduction

In this paper, we present a novel communications architecture for future TCP/IP based networks. The architecture aims to provide unified services both at the network and service layers. The architecture is based on decentralized trust management, IPv6, and Jini. The aim of the architecture is to support infrastructureless operations. The desired features can be summarized into the next five requirements.

- Infrastructure free ad hoc networking
- Seamless connection to the fixed network
- Global roaming and mobility
- Anonymous Authorization and accounting
- Privacy

2 Architecture

The basic structure of our architecture is based on the standards. At the bottom of the protocol stack, we have standard link layers such as BlueTooth, IEEE 802.11 WLAN, etc. IPv6 is placed directly on the top of the link layers. TCP and UDP utilize the IPv6 layer, Jini is built on the top of them, and Trust Management lies at the top.

The important aspects lie in the details, presented in Figure 1. Starting from the top, the first difference is that Jini leases are represented at the operating system level as socket like objects instead of being pure Java objects. The reason for this is the way IP addresses are dynamically updated in our architecture, which requires that all objects that are logically bound to IP addresses need to have updating support from the operating system.

The next difference is that sockets (and other socket like objects such as leases) are at the logical level bound to hosts instead of interfaces. This allows a lower layer routing to dynamically decide which interface and which path to use for each outgoing packet. Thus, in practice, instead of using single IP addresses as the source and destination address in a socket, both the source and destination address are represented by a

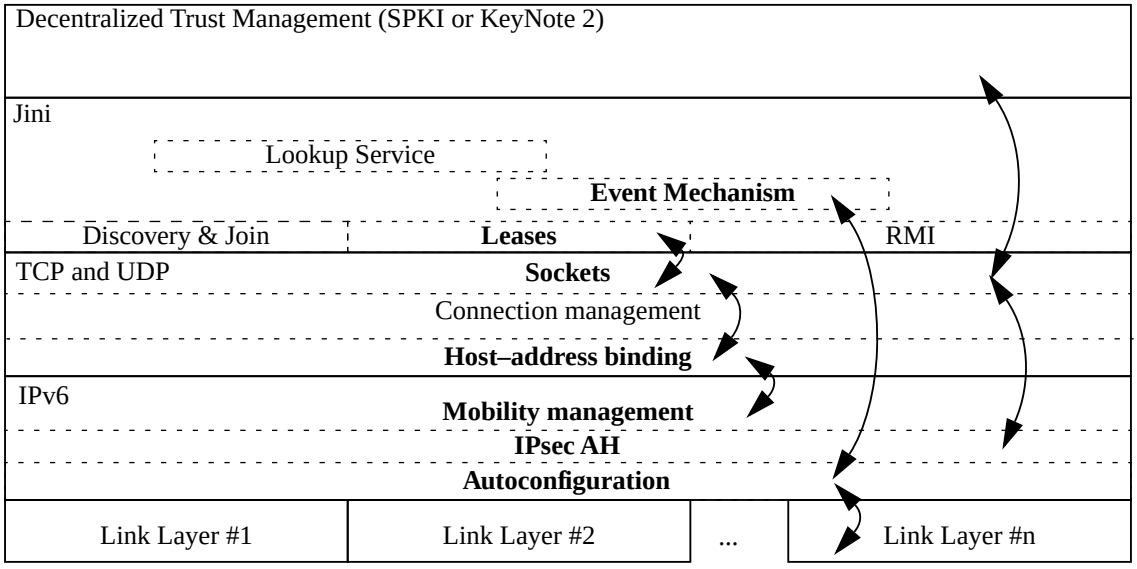

Figure 1: Details of the layered architecture

group of addresses. This change makes it also easier to support mobility through dynamically changing socket bindings.

A further change is a result of the tighter integration between the link layer node discovery, IPv6 layer autoconfiguration, and Jini layer discovery event mechanisms. Even though this difference is not so big from the architectural point of view, it considerably speeds up the time required by the application level to react to topological changes in the underlying network.

A number of architecturally minor modifications are needed inside the IPv6 layer. First, some sort of an ad hoc routing protocol is needed to discover and update the dynamic routing topology. Second, since we are using dynamically changing IP addresses instead of static ones, the way the mobility module handles binding update extension headers needs modifications. This change is also reflected at the socket level, as we already briefly mentioned.

The architecture creates a tighter binding between IPsec Security Associations (SA), sockets, and public key semantics. That is, while currently IPsec is typically used for VPN, i.e., there is only one SA between any two hosts, in our architecture there are typically several SAs. Each of these SAs would have different semantics, i.e., used for different purposes. Thus, there would be one SA for authenticating binding updates, another for securing Jini Lookup Service messages, etc. Occasionally, however, this means that a single IP packet may have more than one AH header.

Finally, the way that identifiers at various layers are allocated and bound together is changed. The functional purpose of the identifiers are different, and should be made distinct. That has not been the case with the current IPv4 addresses, which are today used at least for addressing, auditing, access control, and accounting. In the following, we summarize our findings considering indentifiers:

- Due to privacy reasons, Link Layer MAC addresses cannot be used in generating IPv6 addresses or Jini service identifiers.
- In our architecture IP addresses are fully dynamic in the sense that they are not only dynamically allocated, but both the host ID and routing prefix parts of any address may change and do occasionally change even when the address is in use.
- We recommend that each Jini service keeps a cache of a number of recently visited networks and corresponding ServiceIDs, and use the same ServiceIDs on re-entry to a recently visited network, and different ServiceID in different networks.
- The only type of globally available permanent identifiers are the cryptographic public keys. Each client and service application have their own key pair, or possibly several. When two parties meet for the first time, their exchange their public keys in the form of self signed SPKI or KeyNote 2 certificates. Additionally, the public keys are used for authorization.

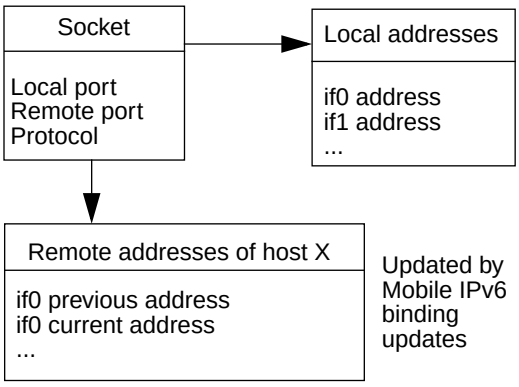

Figure 2: Socket bindings

2.1 Basic functionality

What you get is basically what you expect to get in any modern IP based network. This is intentional; the beauty lies in the way this is accomplished, i.e., in an ad hoc manner with no configuration.

Thus, basically, the architecture allows nodes to automatically create and join into ad hoc local subnets and to create networks of these subnets and to determine and use the network and higher level services available in the current ad hoc network.

3 Mobility support

We want to support almost unconstrained mobility with changing addresses. We do not aim to maximise reliability in our mobility solutions either, but aim for a balance between efficiency and reliability. An intrinsic part of our solution is the new way of binding sockets to packet streams.

In our architecture, a socket end is not bound to a single fixed IP address but to a dynamically changing set of IP addresses. Each set represents a single host or other addressable entity (such as a multicast or anycast channel). The sets are not represented at the sockets themselves, but each socket contains references to exactly two sets, one representing the remote host and the other one the local end-point. The sets may be, and usually are, shared by several sockets. In order to preserve socket uniqueness, we require that each IP address appears in at most address set. The data structures are illustrated in Figure 2. The address sets are updated through extended Mobile IPv6 Binding Updates.

4 Security

The main security goals of our architecture include privacy against tracking of user location and service usage, integrity of signalling and authorization information, and at least rudimentary resistance against denial-of-service (DoS) attacks. In a way, these are all signalling related functions.

In our architecture, each IPsec Security Association has semantics coupled with it. That is, instead of an SA being just an SA that can be used to protect any traffic between a given pair of hosts, each SA is used to protect certain specific kind of information.

4.1 Privacy

We have carefully designed our architecture to emphasize privacy. The architecture is designed in such a way that all globally scoped identifiers are either temporary or cryptographically protected. In fact, there are only two types of permanent identifiers. First, each device has its own IEEE 802 MAC address; this identifier need never be sent outside the local link. Second, each user and node may have one or more cryptographic key pairs. The public key (or a hash of it) may then be used as a permanent globally scoped identifier. However, the architecture allows a user to have several public keys, i.e., several pseudonyms, and it is up to the user's policy to decide which key to use with which service.

5 Conclusions

In this paper, we have described an architecture for the network and service layers for future networks. The architecture is based on existing technologies, including IP version 6, Jini, and SPKI or KeyNote 2. It supports completely infrastructureless operation, fully decentralized authorization, seamless connection to the fixed Internet with global roaming and mobility, and effective privacy protection.

Acknowledgements

We would like to thank Jari T. Malinen of Nokia Research Center for fruitful discussion in the early phases of this work.

References

[1] S. Deering and R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, Standards Track Request for Comments (RFC) 2460, IETF, December 1998.

[2] Ken Arnold, Bryan O'Sullivan, Robert W. Scheifler, Jim Waldo and Ann Wollrath, The Jini™ Specification, ISBN 0-201-61634-3, Addison Wesley, July 1999.

[3] Carl Ellison et. al., SPKI Requirements, Request for Comments (RFC) 2692, IETF, September 1999.

[4] Carl Ellison et. al., SPKI Certificate Theory, Request for Comments (RFC) 2693, IETF, September 1999.

[5] David B. Johnson and Charles Perkins, Mobility Support in IPv6, work in progress, IETF Internet draft, 27 April 2000.

[6] T. Narten, E. Nordmark, and W. Simpson, Neighbor Discovery for IP Version 6 (IPv6), Standards track Request for Comments (RFC) 2461, IETF, December 1998.

[7] S. Thomson, T. Narten, IPv6 Stateless Address Autoconfiguration, Standards Track Request for Comments (RFC) 2462, IETF, December 1998.

[8] Kent, S. and R. Atkinson, Security Architecture for the Internet Protocol, Standards Track Request for Comments (RFC) 2401, IETF, November 1998.

[9] Kent, S. and R. Atkinson, IP Authentication Header, Standards Track Request for Comments (RFC) 2402, IETF, November 1998.

[10] S. Kent and R. Atkinson, IP Encapsulating Security Protocol (ESP), Standards Track Request for Comments (RFC) 2406, IETF, November 1998.

[11] M. Blaze, J. Feigenbaum, and J. Lacy, Decentralized Trust Management, In Proceedings of the 1996 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA, May 1996.

[12] Blaze, M., Feigenbaum, J., Ioannidis, J., and Keromytis, A. *The KeyNote Trust Management System Version 2*, Request For Comments (RFC) 2704, IETF, September 1999.

[13] Ilari Lehti and Pekka Nikander, "Certifying trust," in Proceedings of the Practice and Theory in Public Key Cryptography (PKC) '98, Yokohama, Japan, Springer-Verlag, February 1998

[14] Thomas Narten and R. Draves., Privacy Extensions for Stateless Address Autoconfiguration in IPv6, work in progress, IETF Internet draft, October 1999.

[15] F. Stajano and R. Andersson, "The resurrecting duckling: Security issues in ad-hoc wireless networks," in the Proceedings of the 7th International Workshop on Security Protocols, LNCS, Springer-Verlag, 1999.