3GPP TSG SA WG3 Security - S3#16   S3-000684
28-30 November, 2000
Sophia Antipolis, France

IPSP Working Group   M. Blaze, AT&T Labs
Internet Draft   A. Keromytis, U. of Pennsylvania
draft-ietf-ipsp-requirements-00.txt M. Richardson, Sandelman Software Works
Expires January, 2001   L. Sanchez, BBN/GTEI
July, 2000

IPSP Requirements

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet- Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

To view the entire list of current Internet-Drafts, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Northern Europe), ftp.nis.garr.it (Southern Europe), munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This document describes the problem and solution requirements for the IPsec Policy Protocol.

1.0 Introduction

1.1 Security Policy and IPSEC

Network-layer security now enjoys broad popularity as a tool for protecting Internet traffic and resources. Security at the network layer can be used as a tool for at least two kinds of security architecture:

a) Security gateways. Security gateways (including "firewalls") at the edges of networks use IPSEC to enforce access control, protect the confidentiality and authenticity of network traffic entering and leaving a network, and to provide gateway services for virtual private networks (VPNs).

b) Secure end-to-end communication. Hosts use IPSEC to implement host-level access control, to protect the confidentiality and authenticity of network traffic exchanged with the peer hosts with which they communicate, and to join virtual private networks.

On one hand, IPSEC provides an excellent basis for a very wide rage of protection schemes; on the other hand, this wide range of applications for IPSEC creates complex management tasks that become especially difficult as networks scale up and require different security policies, controlled by different entities, for different kinds of traffic in different parts of the network.

As organizations deploy security gateways, the Internet divides into heterogeneous regions that enforce different access and security policies. Yet it is often still necessary for hosts to communicate across the network boundaries controlled by several different policies. The wide range of choices of cryptographic parameters (at multiple protocol layers) complicates matters and introduces the need to for hosts and security gateways to identify and negotiate a set of security parameters that meets each party's requirements. Even more complexity arises as IPSEC becomes the means through which firewalls enforce access control and VPN membership; two IPSEC endpoints that want to establish a security association must identify not only the mutually acceptable cryptographic parameters, but also exactly what kind of access the combined security policy provides.

While the negotiation of cryptographic and other security parameters for IPSEC security associations (SAs) is supported by key management protocols (e.g., ISAKMP [RFC-2408]), the IPSEC key management layer does not provide a scheme for managing, negotiating and enforcing the security policies under which SAs operate.

IPSP provides the framework for managing IPSEC security policy, negotiating security association (SA) parameters between IPSEC endpoints, and distributing authorization and policy information among hosts that require the ability to communicate via IPSEC.

1.2 The IPSP Problem Space

IPSP aims to provide a scalable, decentralized framework for managing, discovering and negotiating the host and network IPSEC policies that govern access, authorization, cryptographic mechanisms, confidentiality, data integrity, and other IPSEC properties.

The central problem to solved by IPSP is that of controlling security policy in a manner that is useful for the wide range of IPSEC applications and modes of operation. In particular:

- IPSP hosts may be serve as IPSEC endpoints, security gateways, network management hubs, or a combination of these functions. IPSP will manage end-users computers (which may be fixed workstations controlled by a single organization or mobile laptops that require remote access to a corporate VPN), firewalls (which provide different services and allow different levels of access to different classes of traffic and users), VPN routers (which support links to other VPNs that might be controlled by a different organization's network policy), web and other servers (which might provide different services depending on where a client request came from), and so on.

- IPSP administration will be inherently heterogeneous and decentralized. A basic feature of IPSEC is that two hosts can establish a Security Association even though they might not share a common security policy, or, indeed, trust one another at all. This property of IPSEC becomes even more pronounced at the higher level abstraction managed by IPSP.

- The SA parameters acceptable to any pair of hosts (operating under different policies) will often not be specified in advance. IPSP will often have to negotiate and discover the mutually-acceptable SA parameters on-the-fly when two hosts attempt to create a new SA.

- Some hosts will be governed by policies that are not directly specified in the IPSP language. For example, a host’s IPSEC policy might be derived from a more comprehensive higher-layer security policy managed by some other system. Similarly, some vendors might develop specialized (and proprietary) tools for managing policy in their products. In such cases, it is necessary to to derive an IPSP policy specification only for those aspects of a host’s policy that involve interoperability with other hosts running IPSP.

- IPSP must scale to support complex policy administration schemes. In even modest-size networks, one administrator must often control policy remotely, and must have the ability to change the policy on many different hosts at the same time. In larger networks (or those belonging to large organizations), a host’s policy might be governed by several different authorities (e.g. several different departments might have the authority to add users to a firewall or open access to new services). Different parts of a policy might be "owned" by different entities in a complex hierarchy. IPSP must provide a mechanism for delegating specific kinds of authority to specific entities.

- The semantics of IPSP must be well defined, particularly with respect to any security-critical aspects of the system

- IPSP must be secure, sound, and comprehensible. It should be possible to understand what an IPSP policy does; the difficulty of understanding an IPSP policy should be somewhat proportional to the complexity of the problem it solves. It should also be possible to have confidence that an IPSP policy does what it claims to and that and IPSP implementation is correct; architecturally, the security-critical parts of IPSP should be small and well-specified enough to allow verification of their correct operation. Ideally, IPSP should be compatible with formal methods such as implementing security policies with provable properties.

2 Requirements for IPSP

2.1 General Requirements

An IPSP solution must include

- A policy model with well-defined semantics that captures the relationship between IPSEC SAs and higher-level security policies

- A gateway discovery mechanism that allows hosts to discover where to direct IPSEC traffic intended for a specific endpoint.

- A well-specified language for describing host policies

- A means for distributing responsibility for different aspects of policy to different entities

- A mechanism for discovering the policy of a host

- A mechanism for resolving the specific IPSEC parameters to be used between two hosts governed by different policies (and for determining whether any such parameters exist)

and

- A well-specified mechanism for checking for compliance with a host’s policy when SAs are created

The mechanisms used in IPSP must not require any protocol modifications in any of the IPsec standards (ESP, AH, IKE). The mechanisms must be independent of the SA-negotiation protocol, but may assume certain functionality from such a protocol (this is to ensure that future SA-negotiation protocols are not incompatible with IPSP).

2.2 Description and Justification

2.2.1 Policy Model

A Policy Model defines the semantics of IPsec policy. Policy specification, checking, and resolution should implement the semantics defined in the model. The model should, however, be independent of the specific policy distribution mechanism and policy discovery scheme, to the extent possible.

2.2.2 Gateway Discovery

The gateway discovery mechanism may be invoked by any host or gateway. Its goal is to determine what IPSEC gateways exist between the initiator and the intended communication peer. The actual mechanism employed may be used to piggyback information necessary by other components of the IPSP architecture (e.g., policy discovery, as is done in [SPP]). The discovery mechanism may have to be invoked at any time, independently of existing security associations or other communication, to detect topology changes.

2.2.3 IPSP Language

In order to allow for policy discovery, compliance checking, and resolution across a range of hosts, a common language is necessary in which to express the policies of hosts that need to communicate with one another. Statements in this language are the output of policy discovery, and provide the input to the policy resolution and compliance checking systems. Note that a host’s or network’s security policy may be expressed in a vendor-specific way, but would be translated to the common language when it is to be managed by the IPSP services.

2.2.4 Distributed policy

As discussed above, it must be possible for all or part of a host’s policy to be managed remotely, possible by more than one entity. This is a basic requirement for large-scale networks and systems.

2.2.5 Policy Discovery

A policy discovery mechanism must provide the essential information that two IPSEC endpoints can use to determine what kinds of SAs are possible between one another. This is especially important for hosts that are not controlled by the same entity, and that might not initially share any common information about each other. Note that a host need not reveal its entire security policy, only enough information to support the SA resolution system for hosts that might want to communicate with it.

2.2.6 SA Resolution

Once two hosts have learned enough about each other’s policies, it must be possible (and computationally feasible) to find an acceptable set of SA parameters that meets both host’s requirements and will lead to the successful creation of a new SA.

2.2.7 Compliance Checking

When a host proposes the output of the SA resolution scheme, it must be checked for compliance with the local security policy of each host. The security and soundness of the SAs created by IPSP-managed communication should depend only on the correctness of the compliance checking stage. In particular, the even if the SA resolution scheme (which is likely to be computationally and conceptually complex) produces an incorrect result, it should still not be possible to violate the specified policy of either host.

3. References

[RFC-2401] S. Kent, R. Atkinson, RFC2401: "Security Architecture for the Internet Protocol", November 1998.

[RFC-2408] D. Maughan, M. Shertler, M. Schneider, J. Turner, RFC2408: "Internet Security Association and Key Management Protocol (ISAKMP)", November 1998.

Author’s Address

Matt Blaze
AT&T Labs - Research
180 Park Avenue
Florham Park, NJ 07932 USA
Email: mab@research.att.com

Angelos D. Keromytis
Distributed Systems Lab
CIS Department, University of Pennsylvania
200 S. 33rd Street
Philadelphia, Pennsylvania 19104-6389   USA
EMail: angelos@dsl.cis.upenn.edu

Michael C. Richardson
Sandelman Software Works Corp.
152 Rochester Street
Ottawa, ON K1R 7M4   Canada
Telephone: +1 613 276-6809
EMail: mcr@sandelman.ottawa.on.ca

Luis A. Sanchez
BBN Technologies
GTE Internetworking
10 Moulton Street
Cambridge, MA 02140 USA
Telephone: +1 (617) 873-3351
EMail: lsanchez@bbn.com

Expiration and File Name

This draft expires January 1, 2001

Its file name is draft-ietf-ipsp-requirements-00.txt