IPsec support for SCTP in IPv4 and/or IPv6

R. Vamshi Krishna  Kuldeep Gharat  Gautham Anil

November 27, 2004
Introduction to IPsec

Motivation for using IPsec
IPsec Architecture
Security Association and Security Policy

SCTP and IPsec

Introduction to SCTP
Issues in IPsec support for SCTP

Kernel Implementation

Proposal

References
**IPsec : a framework for security at network layer**

There exist many application-specific security mechanism in many application areas.

- PGP,S/MIME for e-mail security
- Kerberos for Client/Server etc.
- Secure Sockets Layer (SSL) for web access

But by using IPsec we implement security at the IP level.

Thus IPsec provides security for not only applications that have security mechanisms but also to the many security ignorant applications also.
IPsec Architecture

IPsec encompasses these functional areas.

- Authentication Header (AH)
  - AH protocol is used to authenticate packets.
- Encapsulating Security Payload (ESP)
  - ESP protocol is used for packet encryption and optionally also authenticate packets.
- Encryption Algorithms
- Authentication Algorithms
- Key Management
  - Internet Key Exchange (IKE)
- Domain Of Interpretation (DOI)
  - DOI Contains various values like identifiers, selectors for SA etc..
IPsec services

IPsec provides these services.

- Access Control
- Connectionless Integrity
- Data Origination Authentication
- Rejection of Replayed Packets
- Confidentiality
- Limited Traffic flow Confidentiality
**Transport Mode vs Tunnel Mode**

**ipv4**
- IPH
- TCP
- Data

**ipv6**
- IPH
- Ext. HDR
- TCP
- Data

(a) Original Packets

**ipv4**
- IPH
- AH
- TCP
- Data

**ipv6**
- IPH
- dest, routing
- AH
- dest
- TCP
- Data

(b) Transport Mode

**ipv4**
- NIH
- AH
- IPH
- TCP
- Data

**ipv6**
- NIH
- xHDR
- AH
- IPH
- xHDR
- TCP
- Data

(c) Tunnel mode
**Security Association (SA)**

A Security Association (SA) is a collection of various parameters that two peers negotiate during connection establishment.

- Source and Destination addresses of resulting IPsec header.
- IPsec protocol identifier: AH or ESP or IPCOMP
- Algorithm and secret key used by IPsec protocol
- Security Parameter Index (SPI) - a 32 bit value to identify an SA

Some implementations might include these additional parameters

- IPsec mode (Transport or Tunnel)
- Size of sliding window to protect against replay attacks.
- Lifetime of Security Association
As SA’s include both source and destination addresses, it can protect only one direction of the traffic in a full duplex IPsec communication.
Hence we need two unidirectional SA’s, one for inbound packets and the other for outbound packets.

Security Associations only tell us how IPsec is supposed to protect the traffic. We need additional information to define which traffic to protect.

This info is stored in Security Policy(SP) which in turn is stored in Security Policy Database(SPD).
Security Policy (SP)

Security Policy specifies

- Source and Destination address of packets to be protected
  - In transport mode they are same as SA src. and dst. addresses
  - In Tunnel mode they may differ
- Protocol (and port) to be protected
  - Some implementations do not allow defining specific ports
  - In that case all ports are protected
- The SA to use to protect the packet
Stream Control Transport Protocol (SCTP) is a new transport layer protocol approved by IETF.

- Uses IP as the network layer protocol.
- Is similar to TCP, UDP protocols.
- Like TCP, SCTP
  - Provides reliable transport service
  - Is connection oriented
**SCTP : An Improvement over TCP**

Unlike TCP, SCTP

- Supports Multi-Streaming
  - This feature allows data to be partitioned into multiple streams
  - They have the property that they can be delivered independently
  - Message loss in any of the streams affects the delivery within that stream

- Supports Multi-Homing
  - Multi-Homing is the ability of a single end point to support multiple IP addresses
  - Benefit is greater survivability of the session in the presence of network failures

- Is message oriented against the stream oriented nature of TCP
Issues in IPsec support for SCTP

For any IPsec implementation to claim support for SCTP the following issues must be considered:

- Proper changes to code to incorporate the new protocol
  - SCTP is very similar to other protocols like TCP
  - Thus will not require much changes

- Supporting Multiple address lists
  - Only one SA must be created even if end-points are multi-homed
  - Every query on an address from the list must return the same SA
  - Hence need for 'list' type for addresses.
Issues in IPsec support for SCTP

- Supporting Multiple SA’s in case of destination does not fully implement the address list feature.
  - Not all implementations at present support the multiple address lists.
  - So until the implementations support multiple address lists, they must be able to setup multiple SA’s.
A new framework has been introduced since Linux Kernel 2.5.x series.

It is called XFRM and Stackable Destination.

XFRM stands for transformer.

struct xfrm_state stands for SA.

struct xfrm_policy stands for SP.

struct xfrm_tmpl is an intermediate structure between xfrm_state and xfrm_policy
Introduction to IPsec

SCTP and IPsec

Kernel Implementation

Proposal

References

**IPsec Output packet Processing**

Output Packet Processing

- ip6_output
  - xfrm_lookup
    - xfrm_temp_resolve
      - xfrm_bundle_create

Lookup Routing Table

- sk_buff
  - original_dst

Binding xfrm_policy as IPSEC policy

- xfrm_policy
  - xfrm_tmp
  - xfrm_state

Look up xfrm_state by comparing with xfrm_tmp in policy

Connect xfrm_state with dst to create stackable destination

- sk_buff
  - dst
  - XXX_output
  - dst
    - dst
      - XXX_output
  - original_dst
    - XXX_output
**IPsec Input Packet Processing**

1. **xfrm6_policy_check**
   - Comparing xfrm_policy and xfrm_state stored in sec_path[]

2. **xfrm_policy**

3. **xfrm_temp**

4. **sk_buff**

5. **sec_path**

6. **xfrm_state**

7. **IP | AH | ESP**

8. **xfrm6_rev**

9. **ip6_input_finish**

10. **IP layer Process**

---

- Look up xfrm_state[] and process packet. If succeeded, link it to sk->sec_path[] list.
- Call protocol processing functions by protocol number.
Proposed modifications and enhancements

After studying the underlying kernel implementation we propose the following

- Create a new data structure to implement address lists
- Implementation can create multiple SPD entries or a single entry.
- Include the ‘address list’ as selector in SA
- Incorporate changes to IKE to be able to send address lists in digital certificates.
- Write a new function or modify an existing function to create multiple SA’s if a destination does not support address lists.
Bibliography

Steve Bellovin.


S. Kent.


S. Kent.


Q. Xie R. Stewart.

# Dummy MultiColoured Table

<table>
<thead>
<tr>
<th>Course code</th>
<th>Course Name</th>
<th>Instructor</th>
<th>Credit/Audit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 621</td>
<td>Artificial Intelligence</td>
<td>Prof. Pushpak Bhattacharyya</td>
<td>Credit</td>
</tr>
<tr>
<td>CS 601</td>
<td>Algorithms and Complexity</td>
<td>Prof. Sundar Vishwanathan</td>
<td>Credit</td>
</tr>
<tr>
<td>CS 631</td>
<td>Implementation techniques in DBMS</td>
<td>Prof. Krithi Ramanritham</td>
<td>Credit</td>
</tr>
<tr>
<td>CS 701</td>
<td>Software Laboratory</td>
<td>Prof. G. Sivakumar</td>
<td>Credit</td>
</tr>
<tr>
<td>CS 694</td>
<td>Seminar</td>
<td></td>
<td>Credit</td>
</tr>
<tr>
<td>HS 699</td>
<td>Communication and Presentation skills</td>
<td></td>
<td>Credit</td>
</tr>
</tbody>
</table>
Dummy Graph using GnuPlot

10 exponential request generators with mean 5.5, uniform update coherency requirement between 1 and 20 and Zipf request for 5 pages (alpha = 0.9). 5 pages updated exponentially with mean 2.5.

Push cost: 1, pull cost: 1.2