

Web security

Basic Authentication



A screenshot of a web browser's basic authentication dialog box. The dialog has a title bar that reads "Username and Password Required" with a close button (X) on the right. The main text inside the dialog says "Enter username for 'Program Committee authentication at akpublic.research.att.com:". Below this text are two input fields: "User Name:" followed by a text box, and "Password:" followed by a text box. At the bottom of the dialog are two buttons: "OK" and "Cancel".

How Basic Authentication Works



Problems with Basic Authentication

- Passwords easy to intercept
- Passwords easy to guess
- Passwords easy to share
- No server authentication
 - Easy to fool client into sending password to malicious server
- One intercepted password gives eavesdropper access to many documents

Digest (Challenge/Response) Authentication



Challenge and Response

- **Challenge (“nonce”):** *any changing string*
 - e.g. `MD5(IP address:timestamp:server secret)`
- **Response:** *challenge hashed with user’s name & password*
 - `MD5(MD5(name:realm:password):nonce:MD5(request))`
- **Server-specific implementation options**
 - One-time nonces
 - Time-stamped nonces
 - Method authentication digests

Advantages of Digest over Basic Authentication

- Cleartext password never transmitted across network
- Cleartext password never stored on server
- Replay attacks difficult
- Intercepted response only valid for a single URL
- Shared disadvantages
 - Vulnerable to man-in-the-middle attacks
 - Document itself can be sniffed

Setting up Basic auth in Apache

- in directory to protect, file called `.htaccess`
- File contents, example:

```
AuthType Basic
AuthName "Rubin's directions (User ID=rubin)"
AuthUserFile /usr/rubin/www-etc/.htpw1
AuthGroupFile /dev/null
require valid-user
```

- In `/usr/rubin/www-etc/.htpw1`

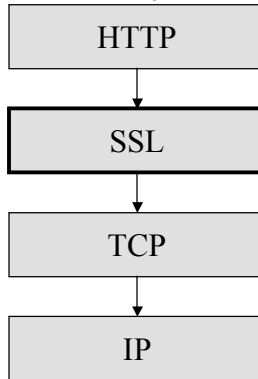
```
rubin:l7FwWEqjyzmNo
```

generated using `htpasswd` program

- Can use different `.htaccess` files for different directories

Secure socket LAYER

- Security at the socket layer



Advantages of SSL

- Confidential session
- Server authentication*
- GUI clues for users
- Built into every browser
- Easy to configure on the server
- Protocol has been analyzed like crazy
- Seems like you are getting security “for free”

Disadvantages of SSL

- Users don't check certificates
 - most don't know what they mean
- Too easy to obtain certificates
- Too many roots in the browsers
- Default settings are terrible
 - ssl v2 is on
 - totally insecure cipher suites are included
- very little use of client-side certificates
- performance! (sites are turning it off)

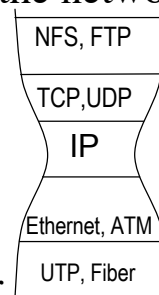
Reality of SSL

- SSL is here to stay no matter what
- credit card over SSL connection is probably safer than credit card to waiter
- biggest hurdles:
 - performance
 - user education (check those certificates)
 - too many trusted sites (edit your browser prefs)
 - enabled version 2 (disable it on the server too)
 - misconfiguration (turn off bad ciphersuites)

IPsec

Network-layer Security

- Network layer is choke-point in the network stack.
- “Hourglass” figure.
- Putting security in network layer allows both higher and lower-layer protocols to use it.



IPsec

- Security at the Network Layer.
- Transparent to applications.
- Transparent to underlying link layers.

- In development since 1993.
- In Draft Standard status.
- Available free for {Free,Net,Open}BSD and Linux.
- Available free for Windows 2000 (from MSR).
- Several commercial products support it.

IPsec Design Principles

- Security
 - Interoperable.
 - High-quality.
 - Cryptography-based.
- Services
 - Encryption.
 - Message Authentication.
 - Replay protection.
 - Access control.
 - Partial protection from traffic analysis.

Design Principles, cont'ed

- Network-layer
 - Transparent to applications (if desired).
 - Independent of link technology.
- No interference with non-users.
- Algorithm-independent.
 - Standard algorithms defined.
 - Closed communities may define their own.

IPsec Components.

- Security Services for Network Layer Packets (IPsec proper).
 - Packet filtering/processing: dropping, passing, or
 - Encrypting and/or Authenticating
 - Both available using Encapsulating Security Payload (ESP).
 - Authentication (only) of IP payload and parts of the IP header when using Authentication Header (AH).
- Key Management (IKE).
 - Security Association establishment:
Keys and Parameters
- Policy Management.
 - With whom to establish SAs and with what parameters.
 - Not part of the IPsec standards.
 - Most modern firewalls support some notion of policy management.

Main IPsec Tools

All you really need to know to understand IPsec:

- Layering
 - ESP and AH are protocol layers.
 - E.g., ESP in Transport mode:
 - the ESP layer between transport layer and IP layer.
 - Possible to layer ESP inside of AH.
- Tunneling
 - Packet is not passed up/down to the next protocol layer in the IP stack.
 - E.g., inbound, packet is passed to protocol layer at the same “height” in the protocol stack, e.g., IP-IP tunneling.
 - E.g., ESP in Tunnel mode:
 - the ESP layer between two IP layers.
- Filtering/Transforming
 - Firewall+: dropping, passing, plus some active processing.

Layers and Modes

Two Security Layers:

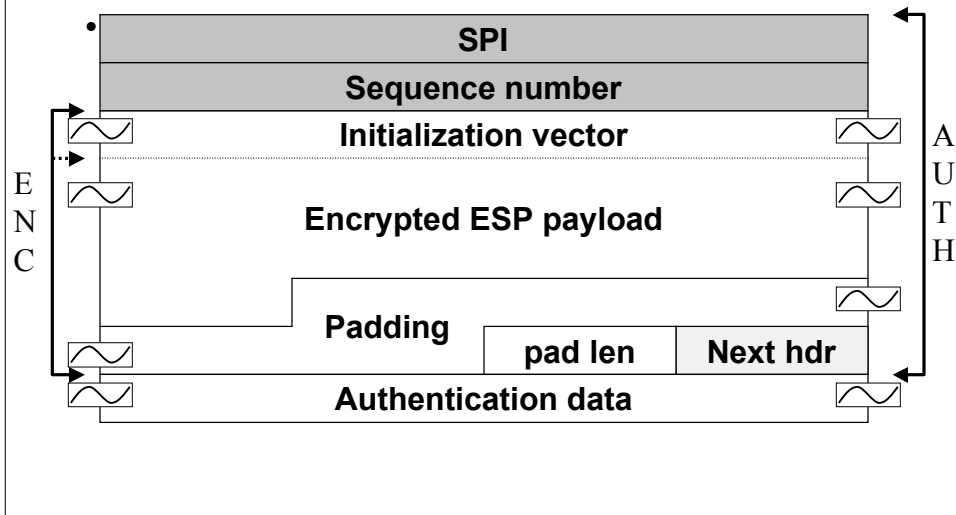
1. Authentication Header (AH).
2. Encapsulating Security Payload (ESP).
 1. Encrypts and/or authenticates IP payload

Two Modes

1. Transport Mode.
 - Between end hosts.
 2. Tunnel Mode.
 - Between end hosts.
 - Between security gateways.
 - Between end host and security gateway.
- All combinations are possible

	ESP	AH
Transport	X	X
Tunnel	X	X

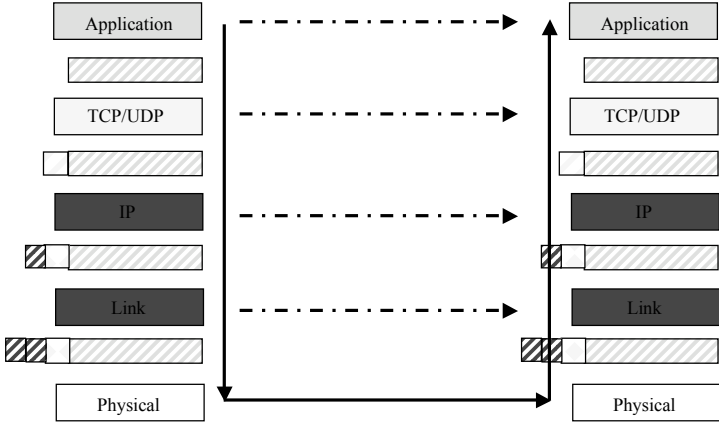
Encapsulating Security Payload



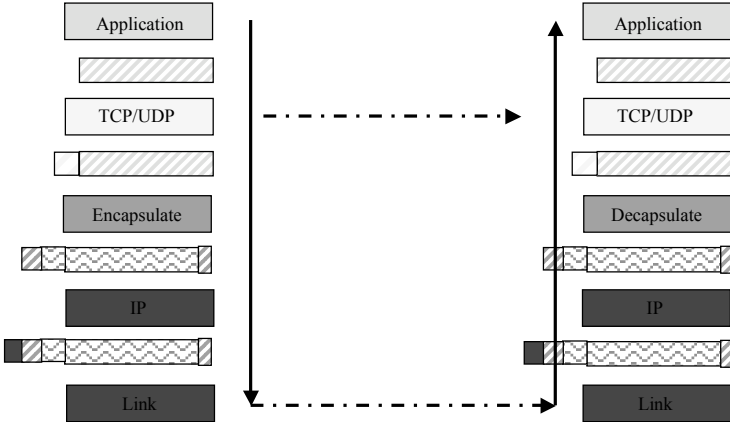
ESP, continued.

- Padding brings payload to multiple of 64bits.
 - Or to hide true length of payload.
 - Note strange location of Next Header.
- First encrypt, then authenticate.
 - Protects against splicing attacks.
- SPI and Sequence number from SAD—we'll come back to this.

IP Protocol Stack

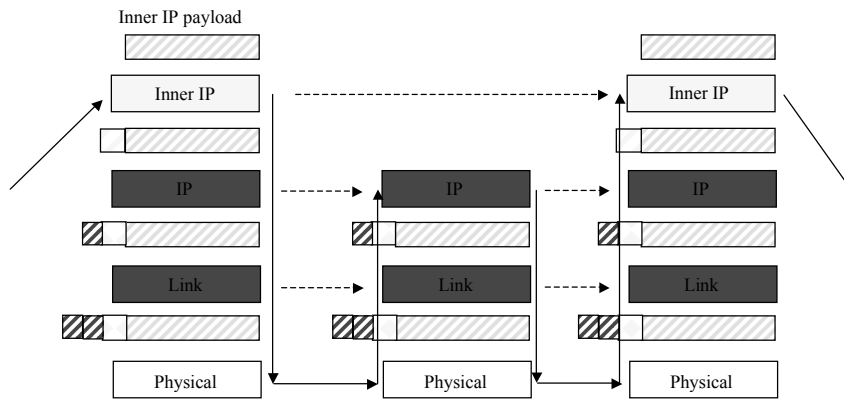


ESP in Transport mode

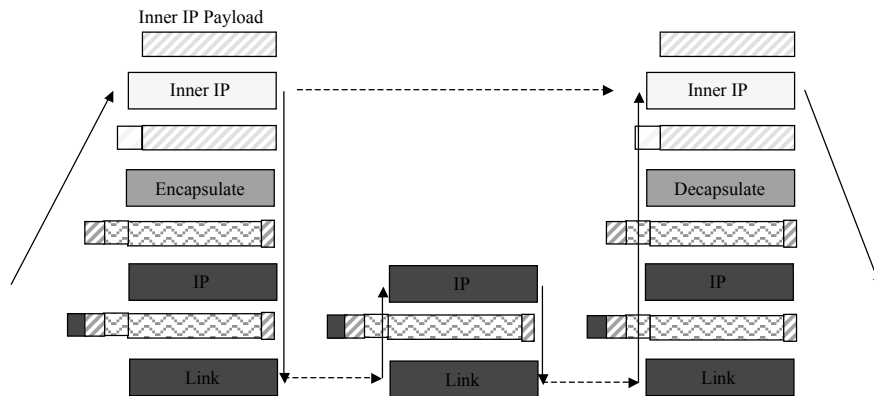


ESP layer is between Transport and IP layers.
 Payload of ESP is entire TCP/UDP packet.

IP in IP Tunneling



ESP in Tunnel mode



ESP layer is between two IP layers.
Payload of ESP is entire inner IP packet.

Filtering/Transforming

- Firewall+: dropping, passing, plus some active processing.
- Since packets processed at IP layer, F/T rules can be based on info in IP header, TCP/UDP header and possibly higher layer headers.
 - E.g., IP source & dest addresses, transport protocol, source & dest ports, header flags.
- F/T Database: Mapping from header info to actions: pass, drop, or apply ESP/AH x transport/tunnel mode with these parameters.
- Outbound processing: Examine packet header info and use it as index into F/T database; perform prescribed actions.

Security Associations

- Inbound processing: The F/T database doesn't work!
 - Transport headers might be encrypted so might not have enough information to look up the correct action.
- New idea: three components
 1. Compact, unique, Security Association (SA) name. SA name is included in ESP/AH header.
 2. Security Policy Database (SPD): Mapping between header info and {drop, pass, process} plus SA names for "process" rows.
 3. SA Database (SAD): Mapping between SA names and filtering/transformation actions and parameters.
- Instead of one lookup in the F/T db, use a two step process. E.g., on outbound:
 - Use header info to look in SPD to find SA name
 - Then use SA name to look in SAD to find actions, keys, etc

Security Associations

- *One-way* association.
- SA name consists of:
 - Security Parameters Index (SPI) of 32 bits.
 - Destination IP address.
 - Protocol (AH or ESP).
 - During IKE, destination host responsible for ensuring that the names of all SAs for packets to the host are unique. For static keying, network management system is responsible.
- Bi-directional stream requires two SAs.
 - One for each direction!

Outbound Processing

- Use header info as an index into SPD.
- Pass, drop, process according to SPD.
- If choice is "process," check for SA name
 - If found, use as an index into SAD and
 1. Process according to algorithms, keys,
 2. Write SPI into ESP/AH header, and
 3. Pass packet down the protocol stack.
 - If not found:
 - Trigger key management to create new SA,
 - Put SA name into SPD, and
 - Put all SA info into SAD.

Inbound Processing

- Use Dest address + SPI + protocol as an index into SAD.
- Process according to SAD. Drop packet if MAC does not verify.
- After header info revealed, use as index into SPD.
 - If SPD says drop or pass then drop.
 - If SPD says process then check that SA name in SPD is the same as SA name of packet.
- Consult “next protocol” field of ESP/AH header, strip the ESP/AH header and pass the packet to the next protocol layer.

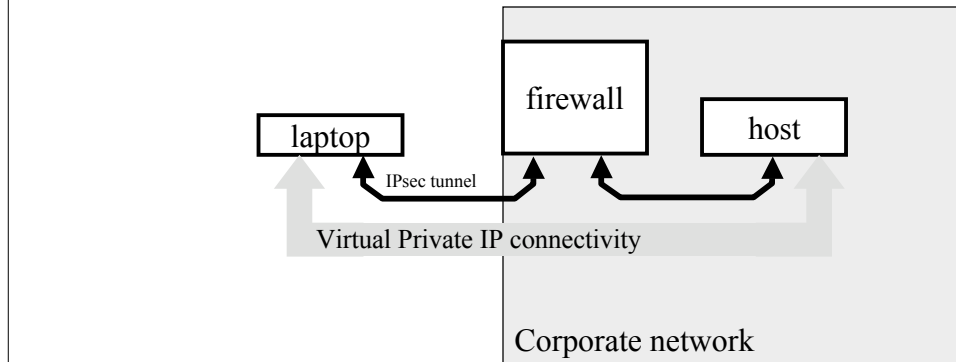
Secure End-to-End communication

- Hosts connecting to each other.
- End-to-end authentication/encryption.
- Tunnel mode is usually redundant, but may be used.



"Road Warrior" Example

- Laptop connects to firewall.
- `Outside` link is tunnel mode IPsec.
- `Inside` link may or may not be over IPSEC.

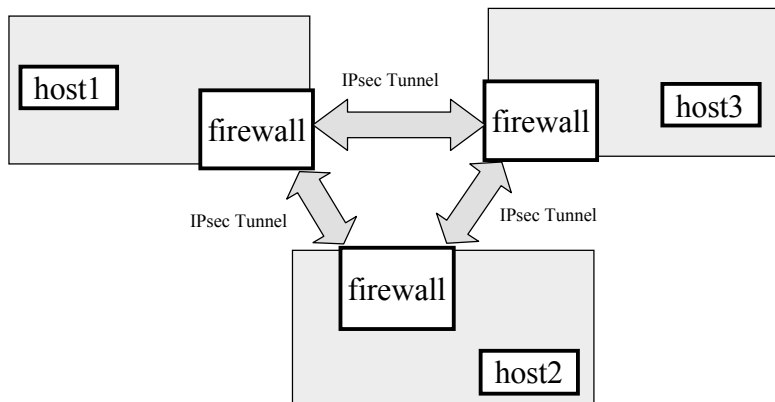


Road Warrior, continued

- Laptop must "get" a corporate network IP address otherwise most packet types will be dropped by firewall policies.
- Using IPsec tunnel:
 - Inner IP address of IPsec tunnel can be statically assigned to be a corporate address.
 - Outer IP address will be assigned by (PPP or DHCP) by local ISP.
- Using Layer 2 Tunneling Protocol (L2TP):
 - Layer 2 PPP between laptop and corporate gateway is carried over IP.
 - Inner IP address (above PPP) is dynamically assigned a corporate address by the PPP protocol.
 - Outer (lower) IP is IPsec between laptop and gateway.

VPN Example

- Tunnels connect firewalled networks.
- Security is invisible to hosts within networks.



VPNs, continued.

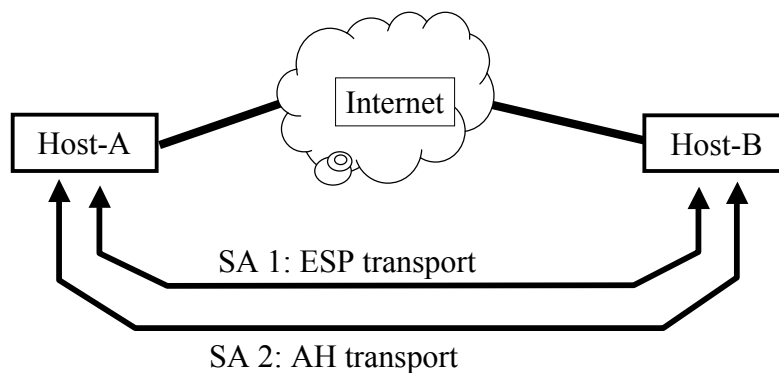
- Replace dedicated data lines.
- Cryptography used to provide privacy.
- Individual nets still have “regular” routing to Internet.
- Firewalls are natural choice for tunnel endpoints.
- Different SAs can be used to segregate travel according to:
 - Protocol type, e.g., QoS traffic over a different SA than best effort traffic
 - Subnetworks, e.g., Officer subnet traffic vs network management traffic vs....
- Additional benefits (other than security):
 - Internal non-routable addresses.
 - Network Address Translation.

Combining Security Associations

- Not all combinations make sense.
- Order of application (during output).
 - Encryption (ESP) first.
 - Authentication (AH) next.
- During input, stripping of headers in reverse order.

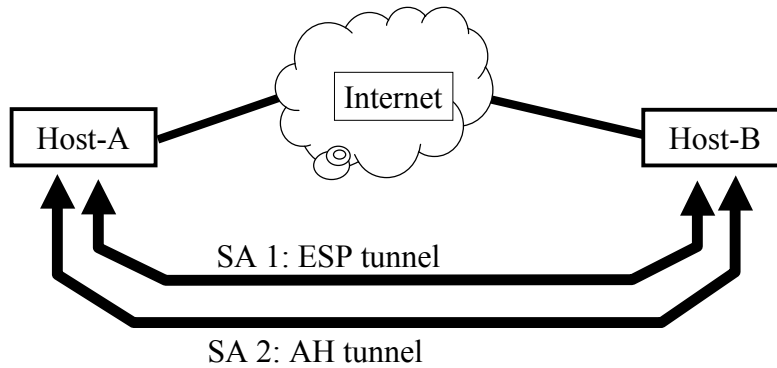
Combining SAs: Transport Adjacency

Transport Adjacency: applying AH and ESP between the transport and IP layers requires two SAs.



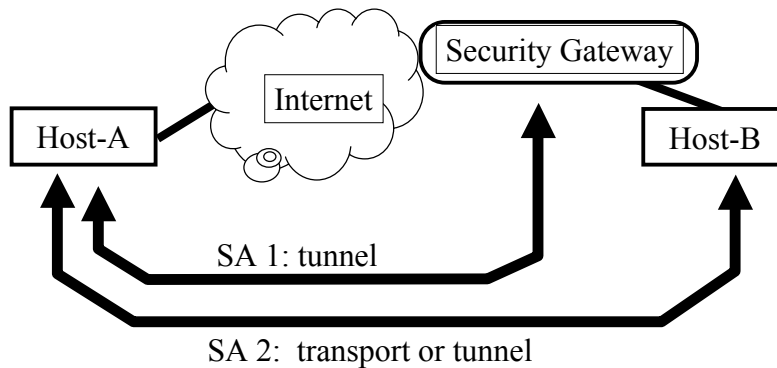
Combo SAs: Iterative Tunneling 1

Iterated tunneling: applying both AH and ESP between two IP layers for tunneling requires two SAs.



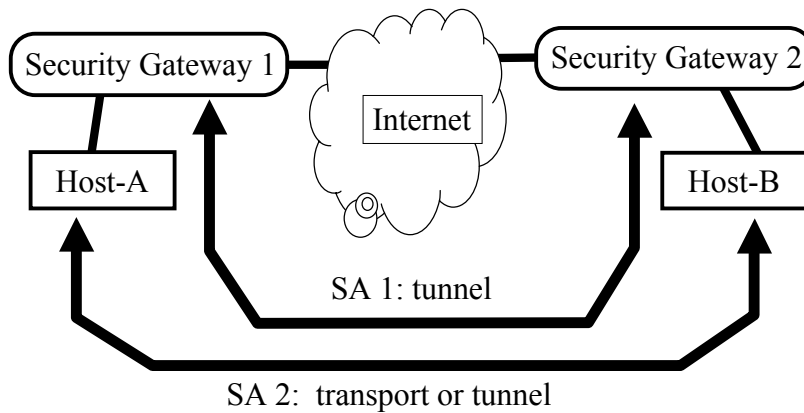
Combo SAs: Iterative Tunneling 2

One endpoint is the same for both SAs ("road warrior").



Combo SAs: Iterative Tunneling 3

Different SA for each endpoint (VPN case).



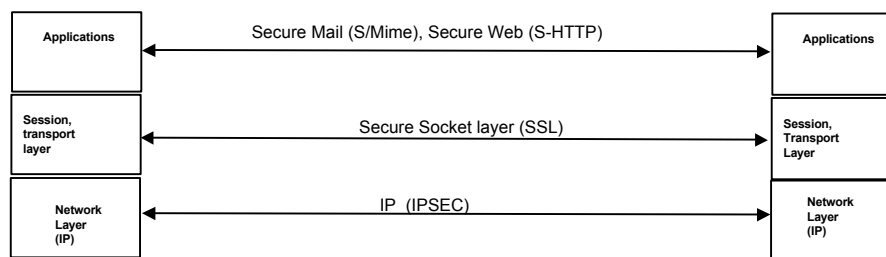
Interaction with Firewalls

- Firewalls already at borders of networks.
- Separate “inside” from “outside.”
- Access control lists.
- Authentication of packets implies access authorization.
- Policy enforcers.
- Proxies.
- Virtual private networks.
- IPsec.
- Management of security policies.

Implementation Options

- Native IP stack.
 - IPsec code is an integral part of the network stack.
 - Best approach.
- Bump-in-the-stack.
 - Shim between network and device drivers.
 - Useful for systems without a native IP stack.
- Bump-in-the-wire.
 - External box.
 - Useful when software on legacy systems cannot be touched.

Secure Internet Protocols: Various Layers



- Is IPsec everywhere sufficient?
- It doesn't work for applications which use storage by a third party, e.g., email.
- The interface between application data (e.g. authorization data) and SAs is not standardized.
- IKE is very inefficient for shortlived security associations
- Transport layer security can be better optimized to the characteristics of the transport protocol, i.e., TCP or UDP.

Internet Key Exchange

- IKE establishes new SAs
 - Limited management of type of SA or SA bundle.
 - Selection of crypto algorithms
 - Secret, shared session keys derived for crypto algorithms
- IKE is very complex. Still no fully interoperable implementations.
- Currently Security Policy Management not standardized—current effort of the IETF
 - SPD and SAD are low level configuration files.
 - Need high level, standardized policy description language and methods for negotiating policy.

Summary

- IPsec provides network-layer security.
- Works with IPv4 and IPv6 (mandatory).
- Most common use currently is VPNs.
- Deployment of DNSSEC may change that.
- Both open source and proprietary commercial products exist.
- Open issues:
 - IKE interoperability
 - Policy Mangament
 - Interface between IPsec and applications
 - More support for public keys either at the network layer or application layer