IKE (Internet Key Exchange)

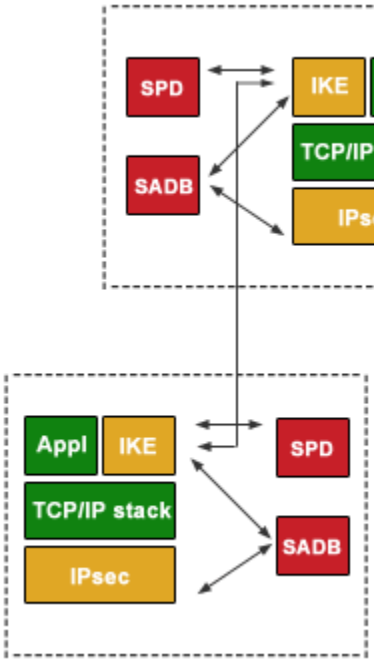
The term Internet Key Exchange refers to the networking protocol that designed to configure a SA (security association) within the IPsec protocol suite of applications. Internet Key Exchange (or IKE) is constructed on top of ISAKMP and the Oakley protocol and is often used in the VPN tunneling process.
X.509 certificates are used for authentication tasks within the architecture of the protocol and can be distributed with DNSSEC using DNS or pre-shared between users in addition to a Diffie-Hellman key exchange. The key exchange is used to aid in setting up shared session secrets that cryptographic keys are then derived from. IKE does require manual security policies for each peer that will connect to the session to be maintained manually.

Why was the Internet Key Exchange Created?

Prior to exchanging secure information between computers, a SA (security association) contract between the machines must be exchanged. During the exchange, each computer will agree on how they will both protect and exchange information during the transaction. IKE provides an IETF standardized way to conduct this exchange. The primary purposes of IKE are to: centralize SA management to reduce network connection time, and to generate and manage the authenticated IKE keys that are used to secure the information to be exchanged. The IKE process is designed to protect both computer to computer exchanges of information as well as any remote hosts that request secure access to an IKE-enabled network.

What is IPsec?

The IPsec protocol is designed to provide "end-to-end" security that operates at the Internet Layer of the OSI Internet Protocol model. The protocol is able to be used to protect information between a security gateway and host (network to host), between two security gateways (network to network), or to safeguard information between a set or pair of computer hosts (host to host).



Some of the other popular Internet security models or systems that have significant market penetration operated in the higher or upper layers of the TCP/IP model such as Transport LayerSecurity

(TLS), Secure Socket Layer (SSL), and Secure Shell (SSH). In order to use many of these protocols, their use had to be architected into the relevant application in order to providesecurity protection. IPsec improves on these models by being able to protect any application sent over a network without previous knowledge of the protocol.

What Is an SA?

A SA (security association) is defined as being a combination of previously agreed-upon security protocols, SPI, and a key. Together, these elements create the definition of security that will be used to safeguard the information exchange between a sender and receiver of data. SPI (security parameters index) is a uniquely identifying value located in the SA to help figure out which security association is being used on the receiving computer.

How Does the Security Parameters Index Work?

A security parameters index, or SPI, is a uniquely identifying entry or value in the security association. Its primary purpose is to help computers tell the difference between multiple SA agreements or associations located on the receiving computer. For example, a client or host computer may have multiple, secure interactions ongoing with other computers at the same time. These transactions are not necessarily intended to be shared amongst the other computers. This is most common on the server-side, when a computer is being used as a remote access or file server and needs to service multiple client computers at the same time. SPI is used to help the server determine which of the open SA's should use when processing incoming or outgoing data packets.

How Does Phase I of the SA Work?

In order to deliver a secure communication session, IKE will conduct a two-part or phase transaction. In order to ensure authentication and confidentiality during each phase, IKE makes use of both authentication and encryption algorithms that the two computers agree upon during the security negotiation. Since the operation is split between two phases, IKE keying is able to be accomplished much faster than when using alternative schemes.
In the first phase of the negotiation, an authenticated and secure channel will be established called the Phase 1 SA. This naming convention is used to help differentiate between the different SAs established during the two phases of the negotiation. IKE will protect the identities of the computers during the exchange to make sure there is no identity information transmitted without being encrypted during the transaction.

IKE Phase I Negotiation

During IKE Phase 1 negotiation, the following steps are taken:

Step 1 – First, the following parameters are negotiated as part of Phase 1 SA:
The encryption algorithm (DES, 3DES).
The hash algorithm (MD5 or SHA).
The authentication method (Certificate, pre-shared key, Kerberos v5 authentication).
The Diffie-Hellman (DH) group to be used for the base keying material.

If the computers choose to use Kerberos v5 authentication, then the computer identity will not be protected until the entire communications payload is encrypted during the authentication step. If pre-shared keys or certificates are used during the transaction, the computer identity will remain protected.

Step 2 – DH exchange of public values. During this step, there is no physical exchange of keys between the computers. They only exchange the baseline information that is required by DH to create the shared, secret key which is then exchanged. After the exchange is completed, IKE will create the master key on each computer that will be used for authentication.
Step 3 – Authentication. During this step, the computers will now try to authenticate the DH exchange. If this step fails, communication using IKE will not be permitted to proceed to the next step. The master key that was created in step two is used along with the agreed upon negotiation algorithms and methods to authenticate the identity of each computer. During this step, the full identity payload that contains the port, protocol, and identity type is first hashed and then encrypted using the keys created from the DH exchange. During this step, the identity payload is protected from being modified or interpreted.
Step 4 – The sending computer sends an offer of a security association to the receiving computer. The responding computer will not be permitted to modify the SA offer. If the offer is modified, the sending computer (or initiator) will reject the message. The responding computer then sends a message that either accepts the offer or an alternative to the proposed SA agreement.
During IKE Phase 1 negotiations, each message has an automatic retry cycle limit of five transmissions. If theIPsec policy allows, the computer will fall back to the clear mode after five failures. If the sending computer receives a reply prior to the cycle timing out, the SA negotiation session will being. The IKE protocol is designed to not provide a finite limit on

the total number of exchanges that can take place. Instead, the primary limitation is based on overall system resources.

How Does Phase 2 SA Negotiation Work?

During Phase 2 of SA negotiation, the SAs are negotiated on behalf of the IPsec service.

Step 1 – Policy Negotiation. During this step, the two IPSec computers will exchange their requirements for conducting a secure data transfer. These include agreeing on the following:
The IPSec protocol (AH or ESP)

The hash algorithm for integrity and authentication (MD5 or SHA)
The algorithm for encryption, if requested: 3DES, DES
Once the computers reach an agreement, two SAs are established. One SA is used for outbound and one for inbound communication.

Step 2 – Refresh or exchange of session key material. In this step, IKE will refresh the key material and generate new, shared, or secret keys for authentication and if negotiated, encryption of the data exchange packets. If there is a need to re-key, a second DH exchange will have to take place prior to this step or a refresh of the DH will be used for the re-key step.
Step 3 – The keys and the SAs are passed to the IPSec driver along with the corresponding SPI.

Phase 2 SA Negotiation

During the negotiation of keying material and shared policy information, the Phase I SA is used to protect the exchange of data. During the second phase, identity protection is provided by IKE through the refreshing of the keying material to help prevent a false SA. The IKE protocol is able to accommodate key exchange payloads for an additional DH exchange if there is a rekey required (PFS is enabled in this case). In other cases; however, the IKE will simply refresh the keying material from the DH exchange in the first place.

The second phase of SA negotiation will result in one SA generated for inbound and one for outboundcommunications. Similar to the retry algorithm used in Phase I, IKE will retry up to five times. If it times out however, a re-negotiation of the Phase 1 SA is attempted. If there is a message for phase II received without an established Phase I, then it will be rejected by IKE.

By allowing the use of a single Phase I SA for multiple Phase II SA negotiations, IKE phase II operations are extremely fast. During the life of the Phase I SA, re-authentication and re-negotiation are not required. IPSec policy attributes are used to determine the maximum number of Phase II SA negotiations that are allowed to be performed. If there is excessive rekeying of the same Phase I SA, the shared, secret key could become compromised.

IKE SA Lifetimes

During an IKE transaction, the Phase I SA is cached. This allows numerous Phase II SA negotiations to be entered (unless PFS is enabled for the master key or the session key policy lifetimes have been reached or exceeded). When the lifetime limit for a key is reached for the session or master key, the SA is forced to be renegotiated in addition to regenerating or refreshing the key.

Once the time-out period is reached for the Phase I SA, or when the key lifetime for the session or master key is reached, IKE will transmit a delete message to the responding computer. The delete message will direct the responding computer to expire the Phase I SA. This message helps false Phase II SAs from being created since Phase II SAs are only valid until the lifetime is expired by the IPSec driver independent of when the Phase I SA lifetime expires. IKE does not force the Phase II SA to expire since only the IPSec driver will know the number of bytes or seconds that have passed in order to reach the key's lifetime. When setting the key lifetimes of the different phases, one has to be careful on making the default timeframes differ significantly. The key lifetime will also determine the lifetime of an SA. If a master key lifetime of 10 hours is set, but a session key lifetime of two hours is used, then a Phase II SA can be in place for a number of hours after the Phase I SA is no longer valid or expired. This condition can result if the Phase II SA is created right before the Phase I SA expires.

Source:

http://www.tech-faq.com/ike.html