1 Cryptography

1.1 Symmetrical

Encryption and decryption is done using the same key.

1.2 Assymetrical

Encryption and decryption are done using different keyes. Also known as public key cryptography. RSA uses this.

There are many algorithms for public key cryptography. However there is a downside to public key cryptography in that it is fairly slow.

There is DES which uses 56 bit key. It was developed back in the 80's. Only known way to break it is brute force, which is just trying every possible key. There are no known shortcuts and there are 2**56 keys.

1.2.1 Triple DES

You take a message and run it through three keys before getting the output. Now there are 2**168 keys and it is agreed that it is not possible to brute force this.

Blowfish can go up to 448 bits if you are extremely paranoid.

AES - Advanced Encryption System.

These are all known as block ciphers, that is you break the message into blocks.

These methods of encryption does not prevent human error or prying eyes. When a password fails it is not because of technical means but because of circumstances mentioned above.

1.2.2 How Public key Works

Public key cryptography works as follows. An individual has a public and private key. When a person wants to send an encrypted message to someone, they encrypt it with that person's public key. Only that person's private key can decrypt that message and ONLY that person should have that privat key. This, of course, does not prevent against theft of the private key.

Likewise, a person can autenticate the origin of a message by decrypting a message that was encrypted with an individuals private key. That person's private key can decrypted only by that person's public key so therefore it can be used to autenicate the origin.

1.3 Where in the protocol stack?

Where in the protocol stack should we put the cryptography information.

By convention the eavesdropper is known as Steve, and the two people wanting to communicate are Alice and Bob. Steve is presumed to be able to see everything on the internet. There are many methods by which Steve can obtain

information that would otherwise be unavailable to them. One such example is ip spoofing.

You could simply encrypt it at the application level.

In the TCP layer you see SSL, or secure socket layer. It fits between the Application layer and the TCP layer and does some authentication and encryption and is transparent to the average user. However, one has to worry about traffic analysis.

For example, suppose you a terrorist cell and wish to communicate between various cells. How do you prevent an organization like the NSA from determing who you are sending a message to. There are various services out there at provide a means to which you can anonymously send data.

You can't directly encrypt IP addresses, so how do you prevent unauthorized or unwarranted viewing. You can use VPN, or Virtual Private Network. The way a VPN works is through tunneling.

Tunneling is as follows. Suppose you have the IP Header, TCP header, and the message. Encrypt it and prepend some encryption information to it and in front of that slap a new IP address. This is transmitted through the VPN to the destination. So an intruder may notice a lot of traffic between source A and source B, but that is as far as they can see.

Link to link encryption is a way for even more security but not possible on the internet.

Disclosure

Traffic Analysis

Masquerade

Message Modification

Sequence Timing Modification

Source

Destination

Denial of Service

Items 4 and 5 on the list belong to a category of attacks known as man-in-the-middle attacks.

1.4 Authentication

There are several methods for authentication.

To be even more sure you can append Message Authentication Code, which is like a hash with a key. It works as follow, you have a msg-goes into a function-fixed length hashcode. MD5 is an example of this.

1.4.1 Example

Alice Gets Msg

Append Time and Date stamp

Calculate hash function

Encrypt hash function with Alice's private key.

Encrypt msg with receivers public key

Send msg

Bob receives message from Alice

Bob decrypts msg with his private key.

Calcluates hash function

Decrypts the sent hash with Alice's public key

Compares the two hashes

This procedure can be use to verify that the message was in fact sent. This solves source reputability problem briefly mentioned above.

Problem arises in determining whether a public key is actually the individuals public key. This is where the area of digital certificates arises. As a standard it is the X509. The idea is you can have a site which lists these digital certificates. This site is a trusted source for these digital certificate. Verisign is an example of this, and essentially they assure that you are who you say you are.

1.5 SSL

Here is a site with the latest draft for SSL: http://wp.netscape.com/eng/ssl3/draft302.txt