Inferno Security 6

Overview

Inferno provides several levels of security:

- Mutual authentication requires that two users or applications that want to communicate establish that they are who they say they are. This is the most basic level of security provided by Inferno. It allows you to establish that you are a legitimate user when you sign on to a service provider.
- Message digesting ensures that an interloper cannot modify messages sent between users.
- Encryption protects the confidentiality of messages so that only the party or parties for whom the messages are intended can decrypt and read them.

Authentication

Authentication requires a combination of elements: a third party that each user can trust, an algorithm or mathematical method to secure messages between users, and a protocol for exchanging messages that ensures that a

third party or intruder cannot pretend to be one of the users, or use some other method to undermine their communication.

One important method for authenticating users in Inferno is the use of digital signatures. Like signing a letter a digital signature testifies to the identity of the sender. However, it is possible for someone ‘listening’ to their communication to read and modify their messages without the users knowing the messages were changed. Therefore, authentication solves only some security needs.

Message Digesting

Message digesting is a technique used to assure that a message was not modified. It uses a mathematical hashing algorithm to change a message into an indecipherable string of fixed length. The hashed value is appended to the message. The recipient can verify the authenticity of the message by applying the same hashing algorithm used by the sender, and then compare the value to the message received. If the values are the same, the message received must be the same as the one that was sent.

Inferno includes a counter in the message digest to ensure that messages are received in the correct sequence and that no messages are inserted by a third party listening in on the line. A secret key is included in the digest to verify the identity of the sender, too.

A message digest ensures that no one has tampered with a message. However, it does not prevent someone from reading it.

Encryption

Encryption provides confidentiality; it involves translating a message that is readable into something unreadable. The readable message can be called plaintext or clear, and the unreadable message, ciphertext. Only

someone able to decrypt the message, or translate it back to its original form, can interpret it.

A mathematical algorithm is used to both encrypt and decrypt a message. Encryption algorithms depend on keys or bit strings of a specified length for encryption and decryption. The nature of an algorithm and the size of the key determine the degree of security.

Two basic types of algorithms are used in cryptography: public key and private or symmetric key. With symmetric algorithms, the same key is used to encrypt and decrypt a message. This key must be a secret, known only to the users who want to communicate. Therefore, it is known as a secret key, also.

A public key algorithm can use a private or secret key to encrypt a message and a public key to decrypt it, or vice versa. The private or secret key is known only to one user. The public key, however, does not have to be kept secret and may be distributed to anyone the user wishes to communicate with.

Inferno uses a public key algorithm for digital signatures and symmetric key algorithms for encryption.

A user can encrypt a message with or without appending a message digest.

Inferno Supplied Algorithms

The choice of algorithms involves speed, degree of security, and export restrictions. The government restricts the exportation of certain algorithms and the key size for certain algorithms. The United States Department of Commerce and the United States State Department regulations restrict the export of the DES algorithm and the RC4 algorithm when the key length is greater than 40 bits.

Note: Anyone who wants to purchase Inferno with DES (Date Encryption Standard) or a version of RC4 supporting a key size greater than 40 bits should contact Lucent Technologies at the telephone number listed in the *Readme* file delivered with the Inferno system.

Hashing Algorithms

MD5 (Message Digest Algorithm #5) is a one-way hashing algorithm that converts or digests a message of any length into a 128-bit number. It is known as a high-speed, 128-bit hash and is used to build a digital signature and prove the origin of messages.

SHA (Secure Hash Algorithm) is a one-way hashing algorithm that is somewhat slower than MD5, but is a more secure 160-bit hash. It was adopted by the National Institute of Standards and Technology as part of the Secure Hash Standard (SHS). Input of any length generates a 160-bit message digest.

Public Key Algorithm

The ElGamal algorithm is a public key system used for creating digital signatures. It uses a private key for signing a message and a public key for verifying it. Using this algorithm will make it easier to adopt a different signature algorithm in the Inferno system as requirements change. This will make it easy to use new algorithms as they are developed.

Encryption Algorithms

The Date Encryption Standard (DES) was adopted by the United States government in 1976 as a standard encryption and decryption system for unclassified data in the United States. There are two types of DES offered by the Inferno system: DES-ECB and DES-CBC. EBC or Electronic BC

or Chain Block Coding are part of the ANSI Banking Standard. CBC is more complex and less vulnerable than ECB. Both versions of DES provide 56-bit keys and, therefore, are not exportable.

RC4 is a symmetric or private key system that is about 10 times faster than DES. It has an unlimited key length, but with a key size of 40 bits, it is exportable. You must contact Lucent Technologies if you want to use RC4 with a key size greater than 40 bits.

Key Exchange Algorithm

The Diffie-Hellman algorithm is used to create a secret key to be shared by others for encrypting messages. Because of this feature, it is sometimes called a shared secret. It requires each user to exchange certain information with the other. This information can be exchanged without encryption. Each user creates the same, secret key from this information. However, no one else listening to their exchange can create or determine the secret key.

Security Protocols

The Inferno system uses the Encrypted-Key-Exchange (EKE) and Station-To-Station (STS) protocols to permit keys to be exchanged and the identities of communicating parties to be verified.

Inferno Authentication

The Inferno system uses the ElGamal algorithm to append digital signatures that are used to authenticate a communication. A third party or certifying authority (CA) is used. See Chapter 9, *Inferno Security Modules and Utilities*, in the *Inferno Reference Manual* for further information.