Encryption

IT managers must contact the IT Security Office for assessment and approval of any encryption tool algorithm and key management before implemented the tool to secure protected information.

Encryption/Hash

Encryption is the process of protecting information by obscuring it in such a way that it cannot be read, accessed or modified without special knowledge and/or a special token.

In many cases it is desirable that nobody can see the information as it travels the network or is stored on a computer locally. This may apply to the entire message being processed, or only to certain parts of it; in either case, some type of encryption is required to conceal the content.

When a computer receives information, it might be necessary to confirm if the sender created the information, or if the information was modified. Such confirmation may be achieved using encryption with a shared secret key.

File and Disk Encryption

File system encryption refers to encrypting selected files and directories on a hard drive. File system encryption does not usually involve encrypting the system files. As such, file system encryption does not protect the host operating system itself, which may then be open to attempted compromises such as brute force password guessing. File system encryption is suitable for use on desktops and servers to protect selected information.

Disk encryption (or full disk encryption) can be done at a hardware or software level, and encrypts the operating system, the swap file, the temporary files, and all information files and directories. In this way, the threat of compromise via operating system exploitation (as with file system encryption) is avoided. Additionally, full disk encryption supports pre-boot authentication. Due to these reasons, full disk encryption is suitable for portable devices such as laptops.

Supporting Encryption/Hash

Encryption is done using either shared key (also known as symmetric) or private/public key (also known as asymmetric) encryption. Normally, shared key encryption algorithms are used to encrypt bulk information, since they are significantly faster than the private/public keys. Private/public key encryption is commonly applied to protect the shared session keys, which, in many implementations, are valid for one communication only and are subsequently discarded. An example of private/public key encryption would be using a secure protocol, such as HTTPS, for a web based transaction on the Internet.

The usual mechanism to protect information tampering is by hash. An example of this is password encryption; which ensures that a password never gets passed in a readable format. Instead, the password is encrypted by a

hashing algorithm as it is entered by the user. The hash value generated by this encryption process is then compared with the hash value of stored password, and if the two hash values match, the entered password is accepted.

Key Management

IT support staff need to understand how these processes work in order to ensure that they are implemented correctly. Regardless of the type of encryption that is being used, a critical issue is that of key management. The compromising of a secret key will lead to the compromising of all information encrypted with the key. The longer a secret key is used, the more exposure it receives, and the greater the chance that it may be compromised. So, for keys that are used to encrypt protected information, the length of key life should be short (no more than 90 fays or a semester).

Other critical factors in the key management process include mechanisms by which keys are generated, escrowed, updated, shared, revoked, and destroyed. The use of key management technologies such as Kerberos or Public Key Infrastructure (PKI) can assist with these issues.

Given that the key management process is complicated and may require exhaustive co-ordination with IT support staff, the use of encryption is not suitable for every application. For instance, in database systems, where the configuration of the key management factors can be controlled (either manually or automatically), then encryption is viable since IT support staff can choose the appropriate level of encryption for each of the data elements in the database table.

In email systems, the use of encryption is viable in some instances, but not in others. For example, IT support staff can control the key on the email server, and University desktops or laptop client systems under their control, in which case, encryption is viable. But IT support staff has no control over email clients outside of the University that may also receive the email message, in which case, encryption is not a viable solution.

Requirements for Strong Encryption of Protected Information

Cryptographic algorithms must be fips validated at: http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2010.htm. Contact the IT Security Office for additional information.