New Zealand Security of Information Technology Publication 204

AUTHENTICATION SERVICES AND MECHANISMS

CHAPTER 1

INTRODUCTION

101. Over the course of history, people have often needed to establish the integrity of a document or correspondent. This process, in modern parlance, is known as authentication. A variety of authentication methods have been used. Seals, chops, and written signatures all provide a means of proving the integrity and authenticity of documents. Identity cards, letters of introduction, personal introduction by a mutual acquaintance, or knowledge of a special phrase or word have all been used as a means of correctly identifying an individual. While these techniques are not all directly useful for authentication purposes within a network or computer system, a number of equivalent electronic measures have been developed.

102. In modern information technology, authentication is one part of an overall security strategy. Authentication often integrates with other security mechanisms, such as encryption and security audit, to provide technical protection. This must be supported by physical, personnel, and procedural security features in order to achieve complete protection.

103. The major use of authentication in information technology (IT) is to provide assurance that a user logging into a computer system or network is the valid holder of the identification code presented. Computer users are typically authenticated on the basis of information known to them, generally a password, or something owned by them, such as a key, token, or smartcard.

104. Another use of authentication in the IT environment is to provide assurance about the origin and authenticity of a message in a manner analogous to the use of a signature on a letter. This technique is often called a digital signature mechanism.

105. Authentication measures are also used to confirm that user login and access is being carried out on the correct system. This prevents systems and users being compromised by an unauthorised user or process *spoofing* the system by inserting false data into a valid data channel or diverting access to an alternate system.

106. An authentication code is only as trustworthy as its supporting mechanisms, and therefore strong protection should be provided for authentication mechanisms and their associated authentication databases. Unless a high assurance trusted system is used, there is a risk that a technically skilled hacker, possibly a staff member, can obtain access to the authentication database. Once such access has been gained, an attacker may

either modify the database directly or copy it to an unsupervised environment to carry out an exhaustive attack on the authentication codes.

107. The type and quality of authentication mechanism appropriate for any particular system will be determined by the level of threat to the system and to the authentication mechanism itself. The *NZSIT 103: Security Evaluation Criteria for Government Computer Systems* provides a procedure for selecting an appropriate minimum standard of protection for various system configurations. Risk analysis can also be used to identify specific authentication requirements.

108. Authentication can be achieved in a number of ways. This publication reviews the major mechanisms for authentication and describes their strengths, weaknesses, and areas of applicability. The mechanisms considered are:

a. passwords,

b. cryptographic authentication,

c. message authentication codes,

d. trusted third parties,

e. security tokens, and

f. biometrics.

109. Additional details of any aspect of authentication may be obtained on request from the GCSB.

CHAPTER 2

PASSWORDS

Introduction

201. Where security policies require users to be accountable for system use, or where access control rules are in place to limit access to specific files, users must be uniquely identified. This can be achieved by using an identifier which may be the user's name, position, or some form of structured access code, and a password for authentication.

202. A password is the most commonly used authenticator, as it is cheap to implement and simple to administer. Authentication is assured so long as the password associated with each user identifier is known only to the authorised user. Unfortunately, passwords are vulnerable to misuse and attack in a number of ways:

a. passwords may be written down where a potential attacker can find them;
b. passwords may be observed at the point of entry;
c. a fraudulent sign-on screen may be displayed to capture the user identifier and password;
d. an attacker may be able to identify passwords through interception of compromising emanations from computing equipment or network interception of login sequences;
e. the authentication subsystems may be subverted; and
f. the password file may be extracted or directly attacked using a dictionary-based attack.

203. There are a number of ways in which passwords schemes can be strengthened against the above attacks. One of the most effective ways of strengthening a password authentication scheme is to ensure that only strong, ie well structured, passwords are used on the system. Even if there is only one weak password on the system, the whole system may be exposed to attack.

Password Protection

204. The effectiveness of a password scheme depends upon how well users protect their passwords, and this will in turn depend upon the users' level of computer security awareness. In addition, if staff are required to sign a password protection declaration (see Annex A) when they are first issued with a user code and password, they are more likely to take seriously the requirement to protect their password.

Password Structure

205. The composition of a password can be crucial to the success of any password-based authentication scheme. The length and structure of the password should provide a sufficiently large number of possible combinations to ensure that any attempt at breaking the password through trial and error is detected before the password is broken, while at the same time making it easy for an average user to remember it. Where the access control system allows only a limited number of tries before generating a security alarm and disconnecting the prospective user, a small password may be sufficient. Where no limitation on login attempts is provided a much stronger password must be used to protect against a connected computer continuously generating login attempts at high speed.

206. Passwords will normally use either the full ASCII printable character set or can be numeric only - as in the case of PIN numbers. The minimum length of any password in both cases should be chosen taking into account the expected life of the password, the average time taken to attempt a login without detection or disconnection, and the number of attempts (if any) to which an attacker is limited by the access control system. The simplest form of password is the common personal identification number (PIN) used in conjunction with bank cards. The PIN is normally a four digit number, and card

terminals usually allow a maximum of three unsuccessful logon attempts before the card is retained.

207. Users will remember a password they have selected more easily than one that has been generated by a system. Care must be taken, however, to ensure that user-selected passwords are chosen so as to minimise the chance of a successful guess or dictionary attack. Passwords to be avoided include:

a. structured passwords, such as combinations of initials and dates;
b. nicknames, pet names, spouse's name, car registration; and
c. words which are found in a dictionary.

208. The best way of ensuring users do not adopt inherently insecure passwords is to use access control software which reviews the password before acceptance to eliminate known words, detect and prevent re-use or cyclic use of passwords, and enforce such other rules as are required by the system's security policy. Using the initial letters of the words constituting a phrase will provide an easily remembered password; for example, "The evil that men do lives after them" would give a password of TETMDLAT; alternatively words may be concatenated to produce a password which is easy to remember but will not appear in a dictionary, such as 'mousecat'.

209. Generating passwords by computer allows the creation of strong passwords which cannot easily be guessed, but these may be meaningless to users and therefore difficult to remember. The risk that a difficult password will be written down and subsequently discovered by an attacker is high. Some systems address this problem through generation of meaningless but 'english-like' passwords composed of one or more nonsense syllables.

210. If passwords are generated by computer, the method used must not be predictable. Password generation systems, therefore, require a good quality random or pseudo-random source. Random number generators commonly provided in computer systems are normally suitable for password generation purposes.

Attacks and Countermeasures

211. **Password Discovery**. The most basic form of attack on a password authentication scheme is unauthorised use of a known password. All users should be aware of the need to keep their passwords secure by not writing them down, avoiding overview during entry, and not sharing them. In addition, simple passwords such as family names, street numbers, and car registrations should be avoided.

212. **Default Passwords**. A vulnerability on many computer systems is the presence of user identifiers and passwords that were set up before or during system installation. Combinations such as field/engineer, super/man, a/a, and many others are commonly used by vendors to perform system installation, but may not have been removed once the system became operational. Hackers

use lists of such 'standard' passwords when trying to gain unauthorised access to systems. System administrators can counter this type of attack by careful review of the password file as soon as the system becomes operational, and at regular intervals thereafter.

213. **Obsolete Userids**. Another problem similar to default passwords is the retention of userids and passwords for staff who have left or no longer require access to the system. This can be countered by regular review of the password file or automated audits of account inactivity. Ideally, administrative procedures for staff postings and resignations should include notification to the appropriate system administrators.

214. **Trial and Error**. An attacker may try a range of possible passwords, and, if their terminal is a microcomputer, may even be assisted by a program which tries logons automatically until a successful password is found. This scheme can be simply foiled by limiting the number of logon attempts, typically to 3, and issuing some form of alarm if the maximum number of failed attempts is reached. This form of defence will often lock out the terminal until the operator or system administrator manually resets it.

215. **Dictionary Attack**. Even though a password file may be encrypted, it is still vulnerable to attack. An attacker who gains access to the system may be able to attack the file in situ or copy it to an uncontrolled environment where the attack can be carried out. A dictionary attack makes use of a computerised database of possible passwords to attempt a match with an entry in the password file. Such databases can be simply created by encrypting the spell-check dictionaries commonly found in word processors. This attack can be countered in a number of ways:

a. **UUCP/FTP Restrictions**. The simplest way to counter a dictionary attack is to stop the attacker gaining access to the password file in the first place. Where the attacker is an outsider, restrictions on the use of file export commands in such features as FTP and UUCP may avoid export of the password file.

b. **Shadow File**. The use of a shadow password file in UNIX and similar environments will allow access restrictions to be placed on the file without disrupting use of the file for login purposes.

c. **Non-Dictionary Passwords**. Dictionary attacks work because staff select passwords which may be found in a dictionary. The obvious counter to this attack is therefore to ensure that passwords are not real words found in dictionaries.

216. **Password Crackers**. All encrypted password files are vulnerable to cryptanalytic attacks if they can be accessed directly or copied out to an uncontrolled environment. The level of vulnerability will depend upon the attacker's knowledge of the encryption mechanism and the strength of the cryptographic scheme used. The GCSB can provide on request advice on the strength of specific systems.

217. **Network Interception**. A major problem with contemporary computer systems is the vulnerability of passwords to interception during a remote logon sequence. The host computer will typically issue a request for user identification and password authentication, and the userid and password will be entered into the terminal and transmitted in plain text back to the computer. Anyone monitoring such line traffic can obtain a valid userid/password combination. This is a major problem on local area networks, where any connected terminal can be configured to monitor all network traffic. Encryption of the password prior to transmission will avoid this problem, a strategy used by automatic teller machines and the some local area networks. Another countermeasure is to use one-time passwords in conjunction with hand held authentication devices; this scheme is discussed in detail in Chapter 6.

Other Issues

218. In the event that a userid/password combination is compromised, departments need to ensure that the period of unauthorised access is minimised. If a compromise is detected, the relevant password should be revoked immediately. However, as password compromise may not be detected, systems may be configured to force regular password change, typically monthly.

219. The chance of detecting unauthorised access can be increased by advising the date and time of the last successful login for each user. It is also useful, for systems which maintain audit trails, to produce a regular printed session summary for each user, and request formal confirmation of the reported usage.

Password Summary

220. Where passwords are used for authentication, the following minimum standards of use should provide adequate security in most situations:

a. **Password Length**. Passwords should be a minimum of 6 characters in length and contain at least one alphabetic character and one numeric character. However, if a limitation on the number of password login attempts is enforced and sufficient protection can be given to the password file, passwords need only be four characters in length and do not require any minimum number of alphabetic or numeric characters.

b. **Password Expiry**. Passwords should expire after 30 days use. However, when positive acknowledgement of system usage and inactive account monitoring are in effect, six monthly password expiry should be adequate.

c. **Password Login Attempts**. A limitation on password login attempts is strongly recommended.

d. **System Accounting**. A regime of positive acknowledgement of system usage is strongly recommended through the use of audit trail reports.

e. **Inactive Account Monitoring**. It is recommended that accounts which have not been used for one month should be disabled.

CHAPTER 3

CRYPTOGRAPHIC AUTHENTICATION

Introduction

301. It may be necessary to use some form of cryptographic mechanism to provide authentication of a user, message, or system. Cryptographic authentication is based on the fact that the cryptographic processes can only be carried out successfully by authorised parties who have been issued with the appropriate cryptographic keys.

302. The process of encryption transforms some unit of data (the plaintext) using some mathematical process (the algorithm) in conjunction with a secret value (cryptographic key), into an output unit of data (the ciphertext) which is meaningless on its own. The ciphertext can be made meaningful only through a reverse process, decryption, using the same algorithm and cryptographic keys. Authentication can be assured, therefore, by controlling the knowledge of the cryptographic algorithm and keys.

303. There are two classes of cryptographic algorithms commonly used in authentication: symmetric and asymmetric. Symmetric algorithms use the same secret key for encryption and decryption, while asymmetric algorithms use different but mathematically related keys for each function. Digital signature schemes use asymmetric algorithms to provide electronic document authentication.

304. Advanced techniques for authentication based on zero-knowledge cryptographic functions are currently being developed, whereby a second party can authenticate a first party without needing to know in advance any information regarding them. These techniques have yet to come into significant commercial use.

Symmetric Authentication

305. Symmetric key systems use the same key to encrypt and decrypt the data. It is therefore necessary to ensure that all authorised parties have the cryptographic key and that no attackers can acquire or calculate it. Knowledge of the key then provides proof of identity. The security of the encryption process, and therefore the trust that can be placed in the authentication mechanism, depends upon the strength of the algorithm, which is a function of

the complexity of the process in which plaintext is transformed into ciphertext; the length of the key; and on secure generation, transmission, storage, and management of the key. An example of a symmetric key algorithm is the DES algorithm which uses a 56-bit key and is widely employed in commercial encryption equipment.

306. Symmetric key algorithms can provide authentication by encrypting a full message or just a cryptographic checksum. A minimum key management requirement is that two communicating parties share the same key and have access to a secure channel, such as a trusted courier, for communicating keys when they are changed. This requirement imposes constraints as key distribution has to take place in advance of any communications.

Asymmetric Algorithms

307. Asymmetric systems use a scheme whereby one key is used for encryption and a mathematically related but different key is used for decryption. The mathematical basis of the key relationship is such that development of a pair of keys is relatively simple, but deduction of one key by knowledge of the other is extremely difficult - and generally computationally infeasible. RSA (developed by Rivest, Shamir, and Adleman) is a popular asymmetric system typically using 1024-bit keys.

308. The use of asymmetric keys allows a special mode of use, known as public key encryption, which is unavailable to symmetric key systems. In an asymmetric system, one key may be kept secret by the owner while the related key can be made publicly available. Authenticity can be achieved by the originator of a message producing a checksum of the transmitted text and encrypting it using his or her secret key. If the checksum can be decrypted by the recipient using the originator's public key, then it could only have been encrypted by the originator.

309. Public key cryptographic techniques provide flexible key management options which do not require advance exchange of secret keys. An independent arbitrator can, given the public and private keys, state that a particular message was originated by a known individual, and provided the message has been acknowledged using an agreed dialogue, verify that it was received by the addressee.

310. Asymmetric key algorithms need to use a much longer key than symmetric systems, and therefore tend to be substantially slower. On the other hand, key management is greatly simplified and secure channels for transmission of keys between the sender and recipient of a message are not required. These considerations mean that public key systems are used where there is a requirement to encrypt small volumes of text (such as symmetric session keys or message checksums).

Digital Signature Standard (DSS)

311. For digital signature schemes to work properly, all communicating parties must agree on a standard message format, covering the position of message digest or checksum fields and digital signature blocks. All communicating parties must agree on the cryptographic algorithm to be used and procedures for key management must be developed and agreed.

312. The American National Institute of Standards and Technology (NIST) has promulgated a digital signature standard based on the ElGamal public key algorithm. The principal features of the DSS are:

a. a hash function is used to generate a message digest (while the DSS does not specify any particular hash function, the Secure Hash Algorithm is commonly associated with DSS);

b. the sender uses a private key up to 1024 bits long to compute a digital signature from the message digest;

c. the signature is transmitted with the message;

d. the recipient uses the sender's public key to determine whether the message truly originated from the sender; and

e. the recipient records the above as evidence for any subsequent third party enquiry.

Smartcards

313. Smartcards can be used to support a digital signature algorithm such as the DSS. Although smartcard processing is relatively slow compared to that of a PC or mainframe system, it is adequate for use in a digital signature system where the frequency with which signatures are to be created or verified is low. Smartcards may also be used to retain private keys securely in memory areas which are accessible only to the cryptographic processor on the card. Further details of smartcards are given in Chapter 6.

CHAPTER 4

MESSAGE AUTHENTICATION CODES

General

401. Authentication of message contents is of particular importance where there is a risk of inadvertent or deliberate modification. This chapter describes the various cryptographic processes which may be used to generate a secure message authentication code (MAC) for messages transmitted over computer

networks. The production of a MAC is an essential part of a digital signature process.

402. Generation of a MAC is achieved by transforming a message in a way which produces a relatively small datum entirely dependent upon the message content. In use, the MAC is appended to or incorporated in the transmitted message: the recipient repeats the MAC generation process over the received message, and verifies that the calculated MAC is identical to that supplied with the message. The MAC may also be referred to as a 'message digest' as described in paragraph 313.

403. The design criteria for a MAC algorithm are:

a. the process must be computationally highly efficient, in order to minimise the overheads in the messaging system;

b. the algorithm must be sufficiently complex to make it computationally infeasible to generate a false message matching a known MAC, in order to minimise the risk of deliberate message modification;

c. the algorithm should operate as a one-way function, i.e. it must be impossible to deduce any part of the message from the MAC; and

d. the algorithm must be strong enough to ensure that the smallest possible change in the message (alteration of one bit) is accurately detected.

Birthday Attacks

404. A Birthday Attack is so called because it can be demonstrated in the following way:

How many people must there be in a room for there to be a greater than even chance that one person will have a birthday on a specified date? The answer is 183. But how many people must there be in a room to have a better than even chance of two having the same birthday - a surprisingly low 23.

405. Finding two messages that hash to the same value is similarly much less costly than finding a message that hashes to a preset value. The birthday attack creates two messages with the same hash value and, by getting one signed, can incorrectly 'prove' that the second message was signed. In order to assure integrity, therefore, the hashing scheme must not only detect minor changes, but it must also be resistent to birthday attacks.

CRC-16 & CRC-32

406. There are two simple authentication codes, called cyclic redundancy checks (CRCs), which are in popular use. CRC-16 is a 16-bit authentication

codes used in the SDLC communications protocol. CRC-32 is used in communications protocols such as HDLC and ZMODEM. Both have been sanctioned by the CCITT for error detection use in communications systems. Neither is particularly strong in cryptographic terms, but they are fast and, as every bit in the message contributes to the hash value, relatively small changes in the message will produce a change in the hash value. They may be of use in some Government applications requiring a minimal level of authentication.

ISO 9797

407. The International Standards Organisation (ISO) has produced a standard, ISO 9797, which specifies the production of a MAC. The standard involves use of some symmetric cryptographic algorithm (eg DES) but does not specify any particular one. MAC generation is achieved as follows:

a. the message text is if necessary padded out to a suitable length using binary zero values, in order to make the message length a multiple of the cryptographic key length *n* (an alternative method in the standard allows for appending a binary 1 then padding with binary zeroes);

b. the data is divided up into *n* -bit blocks;

c. an initial key is randomly or pseudo-randomly generated;

d. the first block of the message is encrypted using the initial key, producing a block of ciphertext the same size as the input;

e. the ciphertext resulting from the previous encryption process is exclusive-or'd with the next block of data and used as input to the next round of encryption as in the above subpara;

f. this process is repeated for each block in the message until the last block has been encrypted; and

g. the ciphertext resulting from encryption of the final block of the message is the MAC (the standard allows for an *m* -bit MAC by using the m leftmost bits).

408. ISO 9797 MAC generation will be effective in most Government applications.

MD2 and MD5

409. A series of 128-bit hashing algorithms have been developed by Ron Rivest, of which the most useful ones are Message Digest 2 (MD2) and Message Digest 5 (MD5). MD4 may also be encountered, but this has now been

superseded by MD5. MD5 is more complex than MD2, and so more secure but less efficient. Both algorithms are specified as hashing algorithm alternatives in the Privacy Enhanced Mail standard for the Internet. Either should be effective in most Government applications.

RIPE-MD

410. A modification of MD4 has been developed as part of the European Community RACE project, and is known as RIPE-MD. It is another 128-bit algorithm and is suitable in most Government applications.

ANSI X9.9

411. The American National Standards Institute (ANSI) have published a message authentication standard for financial institutions which can produce 32-bit, 48-bit, or 64-bit MACs. The standard allows for selective authentication, ie authentication of parts of a message, as well as full messages. The X9.9 authentication algorithm is identical to that defined in the ISO 9797 standard, using a 64-bit cryptographic algorithm and 32-, 48-, or 64-bit MACs. X9.9 is suitable for those Government applications requiring a moderate level of authentication.

Secure Hash Algorithm (SHA)

412. The Secure Hash Algorithm is a US Federal Standard which specifies the production of a 160-bit hashing value, and is the recommended algorithm for use with the Digital Signature Standard. SHA is similar in design to MD4, but has additional features which make it more secure than either MD4 or MD5. It is suitable for all Government message authentication use.

Further Details

413. Further details on the hashing algorithms described in this chapter and their suitability for specific Government applications can be obtained on request from the GCSB.

CHAPTER 5

TRUSTED THIRD PARTIES

Introduction

501. Users in a distributed system often need to authenticate hosts or messages. Management of network security services may need to be centrally provided through the use of what are known as authentication servers or Trusted Third Parties (TTPs).

502. A TTP can provide a number of authentication services:

a. authentication of a user to a system, to provide trusted assurance that the user is genuinely the claimed individual;

b. authentication of a system to a user, to provide trusted assurance of the system's identity;

c. generation and distribution of shared cryptographic session keys;

d. authentication of subordinate authentication servers, in a hierarchical network;

e. arbitration of disputes concerning message authenticity; and

f. facilities to revoke authentication certificates and session keys.

User Authentication by a Trusted Third Party

503. Cryptographic techniques are an essential part of the TTP functions. The integrity, timeliness, and confidentiality of messages between a security server and participating host systems must be totally assured for the authentication server to be trusted.

504. The TTP must first identify and authenticate a user by means of password verification, tokens, smartcards, or digital signature techniques. Once the user is authenticated the TTP may generate session keys directly and communicate them to the user and the system to which the user is seeking access, or may grant a ticket or certificate which contains session keys and authorises the holder to gain access to the target system for a specified period of time.

Kerberos

505. Kerberos is an example of a TTP system. It uses DES encryption and is based on the use of a Kerberos authentication server (KAS) and a Ticket-Granting Server (TGS) both of which are trusted entities on a network. A user shares a prearranged secret key with the KAS for initial authentication. Once the user is authenticated, the KAS issues a ticket which is valid for a period of time equivalent to one login session and which entitles the bearer to obtain

other tickets containing session keys from the TGS. Tickets granted by the TGS are marked with the date and time of issue and the duration of the ticket's validity, and contain the user's identifier and symmetric session keys for subsequent host computer access.

ANSI X9.17

506. The ANSI X9.17 standard on key management incorporates the use of a Key Translation Centre (KTC) which provides authentication services for key exchange. The authentication process uses symmetric encryption (DES) to provide trusted paths between the KTC and the network entities and uses the MAC specified in the ANSI X9.9 standard for message authentication.

ISO TTP Services

507. ISO currently have working drafts of Trusted Third Party services for Open Systems Interconnect (OSI) networks, and the ISO 11770-3 standard specifies the role of a specific form of TTP called a Certification Authority (CA).

508. The basic operation of an ISO TTP involves:

a. authentication of entities using the services of the TTP;

b. signing and sealing and electronic document prior to transmission;

c. registration of the document;

d. certifying time and date stamps of a document; and

e. certifying the contents of a document.

509. The TTP will generate security certificates for network entities before any use of asymmetric key pairs. The certificate will have a specified validity period and will contain the CA's digital signature. A hierarchy of CAs may exist to provide authentication services in larger networks.

CHAPTER 6

SECURITY TOKENS

Introduction

601. An authorised user may own a device which can be used to supplement or replace password authentication. These devices, known generically as

security tokens, interface to electronic reader circuitry and may incorporate electronic circuitry themselves.

Token Types

602. Tokens may be completely 'dumb'. For example, a key which opens an authorised terminal may be sufficient to authenticate the terminal user. The degree to which authentication by this type of mechanism may be relied upon depends upon the ease with which the token can be copied, and the degree of care that the user takes to ensure it is not available to another unauthorised user.

603. Tokens may implement a certain amount of intelligence. The most popular tokens of this type are hand-held authentication devices (HHADs). Where HHADs are used to authenticate a user, the system to which access is being sought interrogates the token either directly through hardware or indirectly through the user using a 'challenge and response' dialogue.

Challenge/Response HHADs

604. Challenge/Response HHADs are used in conjunction with host software (and possibly hardware) to provide a strong level of authentication in the system to be accessed. This type of device is similar to a pocket calculator but also contains a secret key and a cryptographic algorithm. The HHAD is physically tamper-resistant and is protected by a PIN which the user has to enter in order to access the challenge/response process.

605. The authentication process is as follows:

a. the user initiates the login sequence by entering a valid user identifier (and possibly a password);

b. the system to be accessed selects a random number of, typically, six to eight digits which is the challenge value displayed at the user terminal;

c. the user enters his or her PIN number to open the HHAD;

d. the user then enters the challenge number from the screen into the HHAD, which displays a corresponding response;

e. the user enters this response value into the PC or terminal; and

f. the host system checks that the response is correct for that user and the challenge sent.

606. HHADs provide a number of security measures. Entry of a valid response is proof that the user has possession of the HHAD and knows the

correct PIN. Challenges are one-time passwords and therefore, if the line is monitored, do not allow subsequent system access through re-use. Some HHADs incorporate a duress feature, where entry of a specified alternate PIN can produce a response which may be used to alert operations staff that the HHAD user has been forced to access the system against their will.

Time Dependent HHADs

607. A typical time dependent HHAD is approximately the size of a credit card, and contains a battery, a calculator-type display, and clock logic circuitry. Functionally the device combines a clock and a calculator to generate a time-dependent value. The value displayed by the HHAD is cryptographically derived from a stored key specific to the device which encrypts the current date and time. The displayed value thus varies on a minute-by-minute basis in an non-predictable fashion.

608. The user enters the displayed value in lieu of a password, and the host access control system calculates the expected value based on the key associated with the device and user, taking into account the time of day. Possible variations in time are handled by incrementing or decrementing the clock at the host system within a small tolerance limit.

609. HHDA authentication methods provide a higher level of security than a password system. Authentication is based on something the user has (the HHAD) and something the user knows (the PIN). Where device-independence is not an issue, or where it is desirable to limit a user to a specific set of locations, authenticators which interface directly to a PC or terminal are available: these are discussed in the following sections.

Smartcards

610. A smartcard is an advanced version of the popular credit-card, and incorporates a microprocessor, memory, and program storage. It requires electrical contact between the card reader and a set of gold logic circuit contacts on the face of the card. Non-volatile smart card memory uses erasable programmable read-only memory (EPROM) or electrically erasable programmable read-only memory (EEPROM).

611. The smartcard may incorporate software to implement any desired function (including, for example, a cryptographic algorithm) subject only to the amount of program storage memory available on the card. A smartcard may also be used to retain a password or key which may be accessible through the interface logic or may only be accessible to the processor on the card. Cards which incorporate cryptographic processing are generally referred to as advanced smartcard access control systems (ASACS).

612. Smartcards are tamper-resistant and hard to duplicate. They can be initialised with secret keys and PIN numbers which are used by software on the card but cannot be copied out of the card. The limited amount of software and data storage on the smartcard and the relatively slow speed at which the processor and interface operate tend to restrict the amount of processing which can be carried out.

PCMCIA Devices

613. PCMCIA devices are similar to smartcards in concept but are slightly larger. They are designed to connect to a PC through a special form of external bus-connector which provides power to the logic circuitry on the card and allows high speed transfer of information to and from the PC. PCMCIA devices were originally designed for information storage in random access memory (RAM) or on a hard disk assembly built into the card, or to provide communications facilities to laptop PCs. PCMCIA devices are emerging which incorporate the same security functions as advanced smartcards.

Optically Scanned Tokens

614. Exotic photo-sensing devices are available which can be used to read a pattern of light and dark squares displayed on a computer screen and calculate a response value from the image. These devices are to all intents and purposes the same as challenge/response HHADs.

Dongles

615. A dongle is a token which stores user profile information electronically. A dongle would typically contain memory only and is protected against tampering and illicit attempts to read or modify the stored information. Dongles are often connected onto parallel ports while allowing unimpeded throughput of printer data. Special software interrogates the dongle and can use the values returned to authenticate a user.

CHAPTER 7

BIOMETRICS

Introduction

701. Biometric technology can be used to authenticate a person by measuring some personal attribute. Attribute measurements are stored in a reference table or database and checked each time the user seeks to gain

access. Biometric authentication techniques are, on the whole, still in an early stage of development.

702. Two issues are important in authentication: an authorised user should not be erroneously denied access, and an unauthorised user should not be permitted to gain access. In the case of the authentication techniques outlined in earlier sections, error rates are typically extremely low or non-existent as fixed values are being exchanged according to a specified formula or algorithm. In the case of biometrics, however, characteristics are being measured which may legitimately change, and allowances should be made for these variations to minimise false rejections. Two common measures of quality often associated with biometric devices are the false acceptance rate (FAR) and the false reject rate (FRR).

Fingerprints

703. Fingerprints have been accepted as uniquely identifying individuals for more than one hundred years. Fingerprint scanners are now on the market which are small enough and cheap enough to be used as authentication devices to control access to computer systems. These scanners incorporate imaging technology to capture fingerprint details, processing capacity to analyse the fingerprint and extract key parameters, and logic to transfer these values to an access control system. The system compares stored parameters against those supplied and grants or denies access as appropriate.

Voice Scans

704. The characteristics of a user's voice can be used to identify an individual even where some variation occurs over time or due to illness. The technology to capture speech and identify individual characteristics is mature although current commercial devices are still fairly expensive. The user speaks a known word or phrase into a microphone, and a dedicated processor extracts key characteristics which can then be compared against expected values.

Retina Patterns

705. The blood vessels in the back of the human eye vary from one individual to another in much the same way as fingerprints. Retina pattern matching devices require the user to look into an optical unit, which maps the retina with the aid of a beam of light and derives characteristics with which to authenticate the person. This technology requires bulky precision optics and is more suited to premises security rather than computer access applications.

Palm Patterns

706. The shape of a hand varies from one individual to another in the same way as a fingerprint and can be similarly used to authenticate a person. The equipment tends to be physically bulky and is thus better suited to controlling access to premises security rather than computer access applications.

Handwriting Analysis

707. The movements of a pen while a user is writing a given word or signature can be accurately measured in terms of distance, angle, pressure, and velocity. These parametric values vary widely from one user to another for a given written word, and can be measured relatively simply and cheaply using an item of equipment containing position sensors and accelerometers.

708. Capture of the written word is performed by a dedicated unit typically containing a special pen and writing surface. Values derived from the process of writing a word or signature are stored and compared against a reference table for user authentication. The master value computed for the signature is arrived at by averaging a number of attempts, and a deviation index value is also calculated and stored.

Keyboard Characteristics

709. When morse code was commonly used for communications, operators could recognise a morse key operator by their individual style or 'fist'. A similar characteristic has been found to apply to individual use of a computer keyboard. User authentication can be achieved by matching the timings between character pairs against the user's stored profile. This technique is suitable for implementation in an intelligent terminal, but cannot be used to authenticate remote users across a network.

Related annex at http://www.gcsb.govt.nz/nzsit/204/204nxa.htm