Using Encryption Technologies to Protect Data

Alfred John Bacon, CISSP, CISA, CISM, CRISC

Petrobras
Alfred Bacon, CISA, CISM, CRISC, works at Petrobras as an information security and IT internal controls senior consultant, in the corporate finance department. Bacon has worked at Petrobras since 1994. He was responsible for the IT internal controls for S-OX certification in the Petrobras conglomerate. Currently, he consults on special projects in internal controls, such as GRC process control and access control. He still assists in the Petrobras Sarbanes-Oxley Certification.
Agenda

• Introduction
• Cryptosystems
• Symmetric and Asymmetric encryption
• Digital signatures
• PKI and Digital Certificates
• PKIs and management implications
• Database encryption
• Risks in the use of encryption
ISO27002 / COBIT 4.1 Visions

- ISO27002 Item 12.3 – Cryptographic Controls
  - 12.3.1 – Policies for the use of cryptographic controls (where to use and what for)
  - 12.3.2 – Cryptographic key management (ISO 11770 is mentioned)

- COBIT DS-5 – Ensure Systems Security
  - DS5.8 Cryptographic Key Management - Determine that policies and procedures are in place to organize the generation, change, revocation, destruction, distribution, certification, storage, entry, use and archiving of cryptographic keys to ensure the protection of keys against modification and unauthorized disclosure.

- Nothing about good or bad encryption, no guidelines about when and where to use encryption, only the generic guideline that the organization should have policies and procedures that determine where to use encryption and what for and that the cryptographic key management should be carefully defined and managed, i.e. that is an area of risk...
Why use encryption? What is your risk analysis? What is your threat model?

Main reason: I don’t want to depend exclusively on operational system or application access controls!

Interception during transmission (sniffing)

Unauthorized access by privileged users (system administrators)

A risk analysis is important to understand what type of encryption technology and architecture to apply!

Identity Theft

Subverting security controls (Operational System vulnerability exploitation, configuration errors, user errors)

Subverting application controls (SQL injection, exploitation of defaults, configuration errors, etc.)
Introduction

• Encryption allows information to be stored or transported with a guarantee of confidentiality. Encryption can also be used for control over authorization and non-repudiation.

• Modern cryptography applies sophisticated algorithms to “hide” the data, with the guarantee of access and recovery only by authorized parties. These algorithms make use of various complex mathematical functions.

• Fortunately, it is not necessary to understand the mathematical algorithms to use modern cryptography!

• The Romans used a simple mono-alphabetical substitution algorithm, known as the Caesar cipher: $A \rightarrow D$, $C \rightarrow E$, like this:

  – ABCDEFGHIJKLMNOPQRSTUVWXYZ
  – CDEFGHIJKLMNOPQRSTUVWXYZAB

• Today, far more complex algorithms are used, thanks to the abundant availability of IS processing power.
Enigma

- During World War II, the Germans used a sophisticated typewriter, called the Enigma machine, which was capable of operating dynamic poly-alphabetical substitutions, for all military communications.
- The British Intelligence managed to steal one of these machines and break its code, so as to be able to decode all German military communications.
- This story is told in books like “A Man Called Intrepid” and in movies such as “The Code Breakers” and “U-571”.

Modern Cryptosystems

• Modern cryptosystems make good use of available computational power to apply complex mathematical algorithms developed to ensure that their reverse engineering or cracking is statistically impossible or very improbable within the time-span of the information’s utility.

• Modern cryptosystems are so designed that their security does not depend on their being kept secret, to the contrary, they are secure because even after long public scrutiny, no specialist has been able to work out a way to break the algorithms involved.

• The security in modern cryptosystems is guaranteed by the cipher keys, not by the secrecy of the algorithms.
What can be accomplished with cryptosystems?

• Information can be kept confidential, using data encryption, which can include data at rest and data in transit.
• You can be sure of the integrity and origin of your information, using digital signatures.
• You can associate information to people, using digital signatures.
• You can authenticate users, with strong authentication.
• You can guarantee that information has not been tampered with, using digital signatures, associated with non-repudiation.
So what are the basic building blocks?
So what are the basic building blocks?
Symmetric ciphers

Same key used to encrypt and decrypt
Symmetric or Secret Key Cryptography

• Advantages
  – Secure (if key is large enough)
  – Widely used
  – Encrypted text is compact (low overload)
  – Fast (does not overload CPU)

• Disadvantages
  – Complex administration
  – Requires secret key sharing
  – One key per pair of users communicating, so lots of secret keys...
  – Does not support non-repudiation
Some well-known Symmetric Algorithms

- **DES (US Data Encryption Standard)** – ANSI X3.92 – cracked by brute force in 1999, in 24 hours, with a 100,000 super-cluster of PCs – 56 bit keys, data arranged in 64 bit, the de facto standard for many years, until mid 90s.
- **3DES** – ANSI X9.52 – block cipher – 3 subsequent operations of DES – 56 bit keys (3 x 56 = 168 bits), 64 bits blocks, de facto standard until AES become commonly supported in early 2000s.
- **DESX** – DES Extended, created by Rivest, from RSA, used in initial versions of Microsoft EFS, with 128 bit keys.
- **RC2** – stream cipher.
- **RC4** – stream cipher – used by most browsers, where speed is important.
- **IDEA** – described in standard ISO9979.
- **AES (Advanced Encryption Standard)** – Rijndael – present North American standard, chosen in public competition – block cipher, with 128, 192 or 256 bits blocks, and 128, 192 or 256 bit keys – faster and considered more secure than 3DES.
Public-Key or Asymmetric Cryptography

What is encrypted with one key, may only be decrypted by the other key.

RSA is one typical algorithm, Elliptical Curve is another.
Public-Key or Asymmetric Cryptography

• Advantages
  – Considered secure
  – Does not require secret key sharing
  – Easier to administrate
  – Fewer keys to administrate
  – Supports non-repudiation

• Disadvantages
  – Slower than symmetric cryptography
  – Encrypted text is much larger than the symmetrically encrypted text
  – Does not support good encryption of large blocks of data
Some well-known Asymmetric or Public-Key Algorithms

- **RSA key transfer** – based on large prime number factoring, supports secret-key exchange, confidentiality and digital signatures
  - Supported by most available encryption tools and most digital certificate service providers
- **Diffie-Hellman** – based on discrete number logarithms, used for secret key exchange
- **El Gamal** – extension of Diffie-Hellman algorithm, based on discrete number logarithms, used for digital signatures and cryptography
- **Elliptical Curve** – based on discrete number logarithms in an finite field, allows for smaller keys (160 bits in EC are equivalent to 1024 bits in RSA keys). Good option for mobile devices, where processing power is limited.
Message Digest Algorithms

• Message digest algorithms or *Hashes* are frequently used with encryption and possess certain special properties:
  – A hash cannot be executed in reverse mode, in order to recover the original information;
  – The resulting hash bears no relationship with the original text;
  – It is not trivial to guess the original text that generated a specific hash

• **SHA1** and **MD5** are examples of hashes
One-Way Hash Function

- Cryptographic function that receives a variable sized message and generates a fixed-sized digest or summary of the input.
- The hash is easy to compute and should be irreversible.
- Also used to provide data integrity guarantee, when used inside a digital signature algorithm.

This is a simple message

| Hash Algorithm | MD5 | 5d41402abc4b2a76b9719d |

©2012 ISACA. All Rights Reserved
Digital Signature Algorithms

Neiman Marcus’ Secret Cookie Recipe

“Hash Function”

Sender’s Secret Key

Digest

Encrypted Digest

Follow us @ISACA_News

#NACACS
Digital Signature Algorithms

“Hash Function”

Identical?
Sender confirmed!
Sender cannot repudiate!

Neiman Marcus’ Secret Cookie Recipe

Encrypted Digest

Sender’s Public Key

Encrypted Digest

Digests

Digest

Digest
Public Key Infrastructure or PKI

• “Public Key Infrastructure (PKI) is a set of hardware, software, people, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates” (Wikipedia)

• Digital certificates link people (or their digital identity) to a public key, signed by a Certification Authority (respectable, well-known third party), in accordance to a published Certification Policy.

• People or organizations are able to sign documents or decrypt encrypted documents, with the use of their secret keys.

• Digital certification is all about setting up a relationship of trust, between people, between organizations or between both.

• It is technically very simple to issue digital certificates (a Windows or Linux Server can do it), but it is not so simple to issue certificates for business use, with the necessary trustworthiness and reliability.

• Remember: Public Keys can be seen by everyone (after all, they are public) whilst Secret Keys must be kept safely, in a secure environment!
Parts of a PKI

- Public/ Secret pair of keys
- Digital Certificate
- Certification Authority (CA) and Registration Authority (RA)
- LDAP Directory and Certificate Revocation List (CRL)
What do you get with PKI?

- **Authentication** that guarantees that people or organizations are who they claim to be;
- **Confidentiality** that allows access to restricted information only to authorized people or organizations;
- **Authorization** that guarantees that only certain organizations or people have access to specific information;
- **Integrity** that guarantees that information has not been tampered with;
- **Non-repudiation** to be able to prove that information or a transaction has in fact been generated by a certain individual or organization.
Cryptosystems and PKI

Symmetrical-key Encryption

Public-Key Encryption

PKI = Public Key Infrastructure

Message Digest
Using the 3 systems with PKI

- e-Check created
- Message digest of check created
- INTEGRITY

010110
Using the 3 systems with PKI

- e-Check created
- Message digest of check created
- Sender’s PRIVATE KEY encrypts Message digest

NON-REPUDIATION (Digital Signature)
Using the 3 systems with PKI

- e-Check created
- Message digest of check created
- Sender’s PRIVATE KEY encrypts Message digest
- Digest attached to Check
- Encrypts check with one-time Symmetrical Key
Using the 3 systems with PKI

1. **e-Check created**
   - Message digest of check created
   - Digest attached to Check

2. **Sender’s PRIVATE KEY**
   - Encrypts Message digest

3. **Confidentiality**
   - Encrypts check with one-time Symmetrical Key

Follow us @ISACANews #NACACS
Using the 3 systems with PKI

e-Check created

Message digest of check created

Sender’s PRIVATE KEY encrypts Message digest

Digest attached to Check

AUTHORIZATION

Encrypts check with one-time Symmetrical Key

Encrypts Symmetrical Key with receiver’s PUBLIC KEY
Using the 3 systems with PKI

1. **e-Check created**
2. **Message digest of check created**
3. **Sender's PRIVATE KEY encrypts Message digest**
4. **Digest attached to Check**
5. **Encrypts Symmetrical Key with receiver’s PUBLIC KEY**
6. **Encrypts check with one-time Symmetrical Key**

©2012 ISACA. All Rights Reserved
Using the 3 systems with PKI

Digital Envelope arrives
Using the 3 systems with PKI

Digital Envelope arrives
PRIVATE KEY of receiver decrypts Symmetrical Key

AUTHORIZATION
Using the 3 systems with PKI

- Digital Envelope arrives
- PRIVATE KEY of receiver decrypts Symmetrical Key
- SYMMETRICAL KEY decrypts check

CONFIDENTIALITY
Using the 3 systems with PKI

Digital Envelope arrives

PRIVATE KEY of receiver decrypts Symmetrical Key

SYMMETRICAL KEY decrypts check

PUBLIC KEY of sender Decrypts Message Digest

 Produces Digest of Check and compares to original Digest sent

©2012 ISACA. All Rights Reserved
Using the 3 systems with PKI

Digital Envelope arrives

PRIVATE KEY of receiver decrypts Symmetrical Key

NON-REPUDIATION (Digital Signature)

PUBLIC KEY of sender Decrypts Message Digest

INTEGRITY

SYMMETRICAL KEY decrypts check
Where can this process be used?

- E-mails, signed and encrypted for internal or external use;
- Data transport on the intranet and on the internet;
- Web services (OASYS has standards for authentication, digital signature and encryption of SOA messages);
- Data files at rest on local drives.
Certificate Policies relating to Cryptography

- Policies associated to Digital Certificates:
  - Procedures for emission, registration and revocation;
  - Validity period of certificates;
  - Historical access to certificate keys (or how do you verify digital signatures signed by certificates that are no longer valid?);
  - Recovery of secret keys, used for confidentiality, when key owner is unavailable;
  - ANSI X9.57 defines that digital signature secret or private keys should be created, used and destroyed within a secure module (smart-card or token, for example);
  - Periodicity of Certificate Revocation Lists (more often, better);
  - Possibility of escrow keys for access to encrypted documents when secret key owner is unavailable (sickness or job termination).
Digital Certificate

Public Key

Secret or Private Key

Certification Authority

Certificate:
Data:
Version: 1 (00)
Serial Number: 7629 (0x1e95)
Signature Algorithm: md5WithRSAEncryption
Issuer: C=ZA, ST=Wes teni Cape, L=Cape Town, O=Thaute Consulting, OU=Certification Services Division, CN=Thaute Server CA/emailAddress=server.certs@thaute.com
Validity:
Not Before: Jul 9 16:04:02 1998 GMT
Not After: Jul 9 16:04:02 1999 GMT
Subject: C=ZA, ST=Wes teni Cape, L=Cape Town, O=Thaute Consulting, OU=Certification Services Division, CN=Thaute Server CA/emailAddress=server.certs@thaute.com
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
RSA Public Key (1024 bits):
 00:db:31:98:00:6c:6c:0b:62:1c:08:81:1e:0d:0b:1c:8b:bb:
 70:38:15:14:01:60:1f:18:1d:32:33:38:17:
 05:00:2b:0d:01:80:03:31:0d:0b:70:74:77:
 82:0d:21:07:40:0f:01:66:68:00:0d:10:fd:7b:80:0e:31:
 d2:75:0b:01:3e:8e:50:50:ea:7d:01:1a:10:bc:1b:
 e8:55:1d:0e:27:75:1b:7a:61:8f
Exponent: 65537 (0x10001)
Signature Algorithm: md5WithRSAEncryption
6b:22:eb:0e:6a:13:80:ee:20:de:45:00:be:ee:8e:06:77:
5a:de:9d:ea:63:0e:cb:00:6d:0d:08:5b:5e:61:12:0f:2f:
68:af

Follow us @ISACA @ISACANews #NACACS

©2012 ISACA. All Rights Reserved
X.509 Digital Certificate

“I officially recognize the association between this user and this particular secret key”
Digital Certificates

X.509 Certificate

Version
Certificate Serial Number
Signature Algorithm Identifier
Issuer Name
Validity Period
Subject Name
Subject Public Key Information
Issuer Unique Identifier
Subject Unique Identifier
Extensions

CA Digital Signature

X.509 Digital Certificate

Certificate:

Version: 1 (20)
Serial Number: 7629 (0x1e95)
Signature Algorithm: md5WithRSAEncryption
Issuer: CN=CA, ST=Western Cape, L=Cape Town, O-Thanwe Consulting Co., OU=Certification Services Division, CN=Thanwe Server CA/emailAddress=server-certs@thanwe.com
Validity
Not Before: Jul 9 16:04:02 1998 GMT
Not After : Jul 9 16:04:02 1999 GMT
Subject: CN=US, ST=Maryland, L=Passadena, O=Brno Baccala, OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
Subject Public Key Information:
Public Key Algorithm: rsaEncryption
RSA Public Key: (1024 bit)
Modulus (1024 bit):
16:94:8e:fe:ef:4d:8:fd:5b:cb:47:5e:1b:0c:7b:
Exponent: 65537 (0x10001)
Signature Algorithm: md5WithRSAEncryption
60:9c

©2012 ISACA. All Rights Reserved
Follow us @ISACAENews
#NACACS

©2012 ISACA. All Rights Reserved
Follow us @ISACAENews
#NACACS
How Digital Certificates can be acquired

- Digital Certificates are issued by a trustworthy entity called **Certification Authority**;
- Before a Digital Certificate can be issued, it is necessary to clearly identify the user, just as is necessary to issue a Passport or a driver’s license;
- The procedures whereby a user is identified before a Digital Certificate may be issued for this user are published in the Certification Practices Statement (CPS);
- The procedures used by a CA have a direct impact on the level of trust that users will have when they receive their certificates;
- Self-issued certificates cannot expect to be trusted!
How can you send encrypted e-mails to other organizations?

- My organization has digital certification, but how can my users communicate with users from other organizations?
- To communicate securely, it is necessary to have access to the other organization’s LDAP directory or to receive digitally signed e-mails from users from other organizations and recognize their digital certificates, so long as they are signed by a trusted CA;
- Another option is to set up your PKI under a well-known PKI hierarchy, that is mutually recognized;
- Federated Identity Management – A proposal for establishing trust between organizations so as to trust digital certificates and authentication procedures.
Access Control with Digital Certificates

• If you have someone’s public key, through access to their digital certificate, which can be registered with your organization’s LDAP or AD, then you can use it to authenticate the user, by verifying if they have the corresponding secret key.

• To authenticate users, you need to:
  • Make sure that the public key is associated to a trustworthy Certification Authority and that the certificate is still valid
  • Make sure that the private key is safely stored.
Authenticating with a Digital Certificate

Charlie requests access

Charlie’s ID and Certificate with associated Public key

Application Server

Certificate Revocation List

LDAP CRL

Charlie’s Private Key
Authenticating with a Digital Certificate

Charlie’s ID and Certificate with associated Public Key

Certificate Revocation List

Is the Certificate still valid?

Application Server
Authenticating with a Digital Certificate

Charlie's Private Key

Certificate Revocation List

LDAP CRL

Charlie's ID and Certificate with associated Public Key

Application Server

Signed Response

Encrypted Challenge
Authenticating with a Digital Certificate

Server checks Signature with Charlie’s Public Key

Charlie’s ID and Certificate with associated Public Key

Signed Response

Certificate Revocation List

LDAP CRL

Application Server

Charlie’s Private Key

©2012 ISACA. All Rights Reserved

Follow us @ISACANews

#NACACS
Authenticating with a Digital Certificate

If response is correctly signed by Charlie’s Private Key, then response is acknowledged and Charlie is Authenticated!

Charlie’s ID and Certificate with associated Public Key

Certificate Revocation List

Application Server

LDAP CRL
How to keep the secret key safe?

Where is it stored?

How does the user authenticate?

Password

Hard Drive

Virtual Smart Card

PIN

Smart Card

Crypto Operation

Follow us @ISACANews
#NACACS
Smart Cards and Authentication

- Smart Cards are ideal for storing secret keys:
  - The Secret Key is generated inside the smart card and never has to leave it;
  - 2 factor authentication is achieved (something you have and something you know);
  - It is easy to carry;
  - It can also be the company badge, whereby you can achieve a link between physical and logical security!
SSL (Secure Socket Layer)

- SSL allows a server to set up a secure communication channel to a client, where the server’s identity can be verified against a well-known, reliable CA.
- SSL allows a randomly generated secret key to be safely sent from a client, such as a browser, to the server and so symmetrically encrypted communication can be set up and used.
- If the client has a certificate, then double-way, mutual authentication can be set up.
- To be able to use SSL, it is necessary to install a digital certificate in the Web Server.
SSL  

Web Browser  

Randomly Generated Symmetric Key  

Server Certificate  

Server's Public Key  

Symmetric key encrypted with Server's Public Key  

2-way encrypted communication  

Web Server  

Randomly Generated Symmetric Key  

Server Certificate  

Server's Private or Secret Key  

©2012 ISACA. All Rights Reserved  

Follow us @ISACAnews  
#NACACS
Traditional Encryption

Randomly Generated for each session

Session Key
addy0c31034a7db4

Symmetrical Encryption

B's Public Key
a2399cfb1

Assymetrical Encryption

A's Private Key ➔ Kept safely by A
A's Public Key ➔ Accessible for public reference

Assymetrical or Public-Key Encryption

Encrypted Session Key
Vd4%%)u

Assymetrical Encryption

B's Private Key
bb6e51ca

B's Private Key ➔ Kept safely by B
B's Public Key ➔ Accessible for public reference
What about multiple users?

**USER A**
- Session Key: `addf0c31034a7db4`

** USERS B, C or D **
- Session Key: `addf0c31034a7db4`
- B, C and D’s Public Keys:
  - B = `a2399cfb1`
  - C = `28a451b1`
  - D = `103fe27ba`
- B, C and D’s Private Key:
  - B = d48f22783
  - C = 359afbb1d2
  - D = aff459dcb

- A’s Private Key → Kept safely by A
- A’s Public Key → Accessible for public reference
- C’s Private Key → Kept safely by C
- C’s Public Key → Accessible for public reference
- B’s Private Key → Kept safely by B
- B’s Public Key → Accessible for public reference
- D’s Private Key → Kept safely by D
- D’s Public Key → Accessible for public reference

(Ciphered file) Symmetric Encryption (Session Key) + (Encapsulated file) Symmetric Decryption (Session Key)
Management problems in practice

• Traditional public-key encryption system works fine with e-mails, with data in transport in general (in SOA, for example).
• Traditional public-key encryption system presents problems with data at rest:
  – How do you manage access (i.e. give/ take away) to network shared folders for groups that change over time?
  – How do you guarantee access to confidential files in notebooks or PCs whose users have left the organization or are generally unavailable?
• Various suppliers provide full-disk symmetric encryption, which, when associated to one-time password tokens or digital certificate authentication, operate as good protection against mobile device theft
• However, full-disk encryption does not address the risk of unauthorized access to local or network files by privileged users such as network sysadmins. Once the device is on the network, all files are accessible to network administrators.
• Does you organization employ 3rd party help desk services?
• How can you address the risk of unauthorized access to networked data?
Symmetric Encryption, using file system filter criteria

Security Server, where encryption policies are defined and distributed to client stations. Each local client maintains a policy file, where all symmetric keys are kept, one for each policy. Access to the file is given via Digital Certificate authentication.
Advantages with Symmetric Data encryption based on policies

• Easy to manage, user does not require any training;
• User only needs to know that Confidential files should be put in Confidential Folders, local or networked, according to the organization’s Data Classification procedure;
• It is easy to include and exclude users from encryption policy groups;
• Data is protected by the same system on local file systems and on network shares;
• The system does not depend on specific server software versions or brands (large companies are progressively using more storage systems based on non-Windows file systems, such as NAS – Network Attached Storage, whilst still using Windows clients);
• Several software suppliers on the market provide this particular encryption functionality, in particular endpoint software security vendors.
Important considerations

• With a file system filter-based encryption system, it is important to segregate network administration from security administration, or else you will not avoid the risk of unauthorized access by network administrators.

• It is also important to use strong authentication for users, such as one-time password tokens or digital certificates to avoid password-guessing risks.
Database Encryption

• Required by various laws and regulations, such as HIPAA;
• Included as one of the functionalities of various RDBMS vendors;
• Very difficult to implement in practice, because:
  – Key management is complicated (who can be trusted to look after the keys?);
  – The threat models are not well understood;
  – There are a lot of vendors on the marketplace selling packages for database encryption, who don’t explain which risks are addressed by their products.
Database Encryption

• Oracle 11g offers TDE - Transparent Data Encryption – the database encrypts everything “on-the-fly”, the application doesn’t have to be changed at all;
• Several vendors offer full file-level database encryption, even as a hardware appliance;
• Most other RDBMS have cryptographic function calls to encrypt columns (employee salary) or fields (board bonus);
• The problem remains about where to store the cryptographic keys...
• One solution many organizations use is a HSM – Hardware Security Module, where the keys can be stored and only accessed by the database, when performing encryption or decryption functions;
• And then you have to go back to the threat model – who do you want to protect the database from?
• If you use “transparent encryption”, what about SQL injection and other application level attacks, which will send the database the SQL instruction and recover clear-text data anyway, via the “transparent encryption” system?
Database encryption

• According to Rich Mogull:\(^2\):
  – “Before technology comes into play, questions must be answered, such as:
    • Do you want to protect data from database users?
    • Do you want to protect data from external attacks?
    • Do you need to protect all the data or just a single column, like credit card numbers?”…
  – “All major database management systems offer column-level encryption, but none support pulling the keys out of the database by default. My recommendation is to use native encryption capabilities wherever possible, but use third-party key management products to get the keys out of the database.”

Problems with database encryption

- If you choose to adopt selective column encryption, care needs to be taken as to indexes and types of data searches that involve data in encrypted columns;
- Keys ought to be stored outside the database, preferably in a HSM. Backup of the HSM should be provided for, whenever a key update is executed, according to the HSM vendor’s recommendations – remember availability is as important as confidentiality!
- Generally speaking, encryption will result in a performance impact;
- Oracle 11g (r2) TDE allows for hardware-based acceleration on certain CPUs, which will counter the performance overload of encryption.
An architecture for database encryption proposed by Kevin Kenan

An architecture for database encryption proposed by Kevin Kenan³

- The encryption keys are stored in an independent, safe vault;
- Provision is made for a systematic key renewal process, through the Key Manager, administrated by the Key Admin;
- The encryption process is carried out independently from the application and the RDBMS;
- The data that requires protection is stored in encrypted format and can only be retrieved by the same cryptographic process, accessing the same cryptographic engine and keys stored in the vault;
- The application needs to send specific instructions to the cryptosystem to be able to store and retrieve protected data, in a format unknown to would-be attackers;
- The entire process is explained in detail in the book “Cryptography in the Database: The Last Line of Defense”, by Kevin Kenan, together with example Java source code for an implementation.
Recommendations for Database Encryption, by Ron Ben Natan

- Threat models: File theft and abuse of access authorizations by DBAs.
- File theft: full database file encryption, such as Transparent Data Encryption or TDE.
- Abuse of access authorization: partial encryption of database, using internal or external extensions for encryption functions, good key management and transparent handling of encrypted data.

Recommendations for Database Encryption, by Ron Ben Natan

• Main considerations:
  – Key management – clear understanding of which keys are being used for what, where they are stored and how they are accessed;
  – Recovery – What happens if you lose your keys? What is your recovery strategy?
  – Backups and restores – data should be backed up in encrypted format, but recovery must consider the availability of the encryption keys;
  – Clustering – think out your encryption strategy considering the use of clustering;
  – Replication – If you replicate your data, replicate it in encrypted form and make sure you replicate the keys in a secure form;
  – Performance – remember that the use of encryption will impact performance, so follow these guidelines:
    • Encrypt selectively;
    • Never encrypt columns that are used as indexes
    • Do some benchmarking before going into production, so as to have an idea about performance impacts and the possible need for re-sizing
  – Disk space – Encrypted data will almost always consume more space than unencrypted data
  – Audit trail – make sure you have a visible and independent audit trail on the usage of keys and passwords
Public vs. Proprietary Algorithms

• Bruce Schneier\(^5\), in “Secrets and Lies”:
  — Public algorithms go through years of analysis by numerous cryptanalysts but continue to be secure
  — Proprietary algorithms are as good as their creators
  — When consulting with a doctor, how would you react if the doctor proposed: “I have just patented this completely new medicine, based on ground pretzels. Nobody has used it yet but I’m sure it’s the best for you”.
  — For exactly this reason, the market takes a long time to change to new encryption algorithms.

Public vs. Private Algorithms

• AES – Advanced Encryption Standard – was chosen as the successor to 3DES as the new North American standard for symmetric encryption and is now available from all encryption software vendors. It’s public and can be used by everyone. Thousands of cryptanalysts have tried to break it, without success.

• Remember how long it took hackers to break the DVD encryption algorithm, an example of a proprietary encryption algorithm…
Risk analysis in encryption

• Possibilities for Breaking Cryptography:
  – Software vulnerabilities
  – Management process vulnerabilities
  – Keyboard loggers that can capture passwords or passphrases, when PKI is not used
  – Breaking the algorithm by brute force or by exploiting a vulnerability
  – Statistical analysis (Domino Pizza Index in Washington) → Used to break the WEP key in Wi-Fi
  – Breaking the encryption keys by brute force → dependent on computational power, which, by the way, doubles every 18 months (Moore’s law…)

• Security Certification: Common Criteria or ISO15408 → third party examination of development process, with a security level certification. Security services such as FBI, NATO require security certification, which can be consulted on the internet. See certified data protection products on Common Criteria site: http://www.commoncriteriaportal.org/products/#DP
### Breaking Symmetric Encryption by brute force

<table>
<thead>
<tr>
<th>Bits</th>
<th>1% of key space</th>
<th>50% of key space</th>
</tr>
</thead>
<tbody>
<tr>
<td>56</td>
<td>1 second</td>
<td>1 minute</td>
</tr>
<tr>
<td>57</td>
<td>2 seconds</td>
<td>2 minutes</td>
</tr>
<tr>
<td>58</td>
<td>4 seconds</td>
<td>4 minutes</td>
</tr>
<tr>
<td>64</td>
<td>4,2 minutes</td>
<td>4,2 hours</td>
</tr>
<tr>
<td>72</td>
<td>17,9 hours</td>
<td>44,8 days</td>
</tr>
<tr>
<td>80</td>
<td>190,9 days</td>
<td>31,4 years</td>
</tr>
<tr>
<td>90</td>
<td>535 years</td>
<td>321 centuries</td>
</tr>
<tr>
<td>108</td>
<td>140,000 millennia</td>
<td>8 millions of millennia</td>
</tr>
<tr>
<td>128</td>
<td>146 billions of millennia</td>
<td>8 trillions of millennia</td>
</tr>
</tbody>
</table>

Note. These figures are only valid if the seed that generates the keys is truly random!
Equivalence for brute force breaking

<table>
<thead>
<tr>
<th>Asymmetric Key</th>
<th>Symmetric Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>512 bits</td>
<td>64 bits</td>
</tr>
<tr>
<td>1792 bits</td>
<td>112 bits</td>
</tr>
<tr>
<td>2304 bits</td>
<td>128 bits</td>
</tr>
</tbody>
</table>

Note: the Brazilian Government official PKI works with 1024 and 2048 bit RSA keys
About encryption software

• Key manipulation ➔ if a smart-card is used, all key manipulation is done inside the smart-card or token and an API provides access to all smart-card functions.

• Some countries require software vendors to provide back-doors in encryption software to provide access to law agencies during criminal investigations.
  – How can you guarantee that all law officers are honest and ethical?
  – How do you guarantee that hackers don’t find out about these back-doors and use them themselves?

• End-point security
  – How do you guarantee that a Trojan horse doesn’t sign off on the wrong document?
  – How do you guarantee that a Trojan doesn’t send clear copies of encrypted documents to a third party?

• These points need to be addressed by selecting a well-known software vendor, preferably with security certified products (Common Criteria, NIST or FIPS certified).
Audit of Encryption Systems

• What are the controls over digital certificate emission, usage and revocation?
• Are the certification policies followed? Are the records available?
• If symmetric encryption is used, how are the keys generated, kept and discarded?
• Is there a cryptography usage policy, such as a Data Classification standard which says where encryption is mandatory? If there is a standard, has it been communicated to users? Is the standard followed in the organization, are the records available?
Management decisions

• Organizational Data Classification standard – what needs to be encrypted has to be defined by business needs;
• How to encrypt confidential data – an IT and IT Security decision process;
• PKI – buy commercially available certificates for employees or set up organization’s own PKI?
• If you set up a PKI, do it inside a well-known PKI hierarchy, so that others may trust your PKI;
• Be careful with key management and storage – they are the basis for your data security;
• Mandate a segregation of duties between your IT help desk and network administration and the security team. The security team alone should take care of your key management process.
Questions?

Email: alfred@petrobras.com.br
Collaborate – Contribute – Connect

- www.isaca.org/knowledge-center

- The Knowledge Center is a collection of resources and online communities that connect ISACA members – globally, across industries and by professional focus - under one umbrella. Add or reply to a discussion, post a document or link, connect with other ISACA members, or create a wiki by participating in a community today!