Virtual Invisible Disk Design for Information System Security

Faisal Nabi
Internet Security Group, Department of Computer Science, Colorado State University, Fort Collins, CO 80523-1873, USA (Email: nabi996@cs.colstate.edu)

(Received Jan. 4, 2006; revised and accepted May 5 & Nov. 8, 2006)

Abstract

Information is one of the important enterprise assets for any organization or institute in order to facilitate day-to-day operation and effective decision-making. Access to the right information by the right people is increasingly vital in gaining a competitive advantage. However, managers of information system are facing real challenges with the growth of security risks created by intruders using modern hacking techniques. For this reason, it is imperative that these issues should be addressed in order to ensure security for the information system. This research focuses on three of the major issues related to information security: Confidentiality, Data Integrity and Availability of Services and Secure Information Storage. In this paper we present the Model of a Virtual Invisible Disk (VID) and discuss a timeout key session technique. The authentication method for the VID is based upon incorporating biological information into a secret key to Access the VID that can assist in overcoming the constantly growing security challenges.

Keywords: Biometrics, confidentiality, cryptography, data integrity, virtual invisible disk

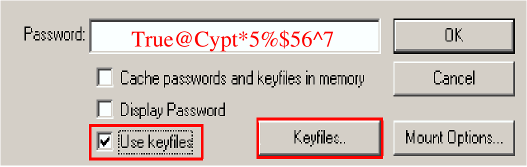
1 Introduction

Information is one of the most important assets of any modern organization or institute. Organizations of all shapes and sizes need to embrace information systems & technology, if they wish to survive, let alone thrive, in a highly competitive environment. Effective operational control & strategic direction are increasingly dependent on the availability and exploitation of information. It is vital that adequate security and control procedures are introduced to ensure that all the information embedded within organization system retain adequate integrity, confidentiality and availability [3]. The issue of information security has been recognized since the advent of first multi-user computer system for sharing information resources. As we begin the 21st century, this need has become even more significant as countries join together to securely share information at the global level. Information security is a daunting challenge, since organizations are often dealing with information content that ranges from the simple to the complex (e.g., intelligence information, financial information, travel information, health records, citizenship records, e-government, etc.) in an interoperable and constantly changing environment [8]. The Internet has provided organizations with access to valuable information and services, and an instant communications pathway with partners around the globe. However, with such rich source of information and communication lies great risk. It is often stated that the only sure way to remain secure is never to connect to the Internet. This is not a viable option for the modern business. Organizations that want to take advantage of the vast resources available on the Internet must find some happy medium of accessing and providing access while ensuring that the adverse consequences of getting on the Internet are minimized. Internet data/Information storage is only useful if information is safe, private and retrievable. Some of the issues in making this scenario a reality include securing data keys, ensuring quality of service and legal matters of trust. Making files tamper proof and "non-repudiable" is an important feature for Information/e-commerce. The integrity of documents such as invoices, receipts and health records must be guaranteed. The system must detect discrepancies between original and reconstructed files and be able to trace any suspected tampering. Securing Information is not only about storage space. It's about intelligent approach for the systems that make it secure and flexible. Securing information on the internet works by breaking data files and objects into encrypted, erasure-resistant fragments that are replicated and stored randomly across a global network of data storing into Virtually Designed Secure Disk onto servers. Keeping in view fundamental issues regarding Information Security. The issue that first springs to mind is that one must maintain confidentiality, keeping sensitive information secret. Other key issues such as data integrity require addressing. This is the assurance that any transmitted data has not been modified. An increasing problem facing organizations now is the availability of data and services, meaning that systems from complex enterprisewide Internet works to small organization resources are up and running when they are needed.

The remainder of this Research is organized into three sections. In Section 2, we characterize and discuss problems which lead to three major issues regarding security (1) Confidentiality (2) Data Integrity (3) Secure Availability of Information & its Storage, with the examples. In Section 3, we present the Virtual Invisible Disk (VID) technique, which can provide solutions to the 3 major issues regarding IS security. We discuss the VID access key creation and identity management technique as well as key management and distribution strategy in a secure environment. In Section 4, we conclude and discuss the significance of this technique and its integration into modern systems security. We further discuss future research work in this section. This is based on enhancing additional functionality integration into the Virtual Invisible Disk (VID).

2 Specification & Characterization of the Problems in Different Scenario IS Security

With the advent of PKE encryption schemes such as PGP & True Crypt tools etc, most public and private keys are generated using passwords or pass phrase, leaving the password generation steps vulnerable to brute force attacks. If a password is selected that is not of significant length, that password can be susceptible to what is termed a brute force attack, in an attempt to generate the same keys as the user. Thus PKE systems such as RSA may be broken by brute force, not because of any deficiency in the algorithm itself, but because of deficiencies in the key generation process. However, Password/PassPhrase based Key generation is very much likely refers to chances of Brute Force attack. As mentioned given below examples.



Password: True@Cypt*5%$56^7

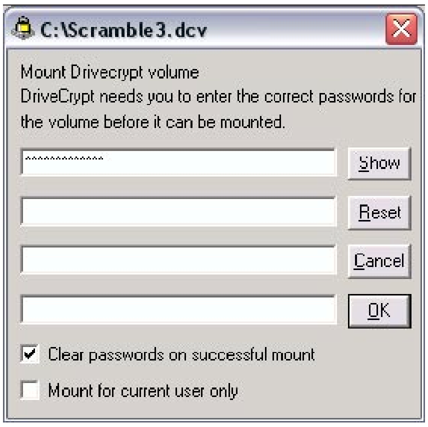
□ Cache passwords and keyfiles in memory

□ Display Password

☑ Use keyfiles

Figure 1: True Crypt using passwords/pass phrase for key file generation

Key generation using 4 lines The generation, transmission, and storage of keys are all part of the key management process and all cryptographic systems must address key management issues. Any cryptographic system is only as strong as the key is secure. Consider a situation in which you have only chosen to use the strong cryptographic algorithms, you have verified that there are not any flaws in the vendors' implementations, and you have generated your keys with great care. How secure is your data now? It is still only as secure as your private or secret key. These keys must be safeguarded at all costs, or you may as well not even use encryption. Since keys are simply Strings of Data, they may well be stored in a file somewhere in your system's hard disk. For authentication, private keys for SSH-1 are stored in the *identity file* located in the *.ssh* directory under a user's home directory. If the file system permissions on this file allow others to access the file, then this private key may be compromised. Once others have your private or secret key, reading your encrypted communications becomes trivial. However, in some vendor implementations, your keys could be disclosed to others because the keys are not stored securely in RAM. As it is understood, any information processed by a computer, including your secret or private key, is located in the computer's RAM at some point. If the operating system's kernel does not store these keys in a protected area of its memory, they could conceivably become available to someone who dumps a copy of the system's RAM to a file for analysis. Most operating systems, including Windows, can be configured to write debugging information and contents of the system memory to so-called memory dump files when an error occurs (system crash, "blue screen," bug check). Therefore, memory dump files contain sensitive data. TrueCrypt cannot prevent cached passwords, encryption keys, and the contents of sensitive files opened in RAM from being saved unencrypted to memory dump files. Note that when user open a file stored on a TrueCrypt volume, for example, in a text editor, then the content of the file is stored unencrypted in RAM (and it remain unencrypted in RAM until the computer is turned off). Also note that when a TrueCrypt volume is mounted, its master key is stored unencrypted in RAM. in the such case TrueCrypt which also use automatically encryption or decryption right before the loading/saving without any intervention of user & Data storage on an encrypted volume (in memory/RAM) can be decrypted using Password/Key file based on Pass



C:\Scramble3.dcv

Mount Drivecrypt volume

DriveCrypt needs you to enter the correct passwords for the volume before it can be mounted.

****

☑ Clear passwords on successful mount

□ Mount for current user only

Figure 2: Drive Crypt use passwords/pass phrase for key generation using 4 lines

Phrase which mean entire file system is on great Risk (i.e. File name, Folder name, contents of every file & free space. These memory dumps are called *core dumps* in UNIX, and they are commonly created during a denial of service (DoS) attack [4]. Thus a successful hacker could generate a core dump on your system and extract your key from the memory image. In a similar attack, a (DoS) attack could cause excess memory usage on the part of the victim, forcing the key to be swapped to disk as part of virtual memory.

In recent years there has been the development of the Secure Sockets Layer (SSL) protocol for Internet security. SSL provides a straightforward method for adding security to existing applications and network infrastructures [5]. SSL is application protocol-independent and provides server authentication, data encryption, and message integrity, which create a securedchannel to prevent others from tapping into the network. SSL is layered beneath application protocols such as HTTP, telnet, FTP, and gopher, and layered above the connection protocol TCP/IP. This protocol uses both Symmetric and Asymmetric Algorithms [9]. If you are using an algorithm that is known to be secure but the algorithm is being applied incorrectly, security holes can result. An excellent example of a security hole resulting from misapplied cryptography is Netscape's poor choice of random number seeds used in the Secure Sockets Layer (SSL) encryption of its version 1.1 browser. This is big security flaw. We'll see that this particular bug is an almost classic example of one of the ways in which vendors implement broken cryptography, and as such it continues to remain relevant to this day. We will limit this discussion to the vulnerability in the UNIX version of Netscape's SSL implementation, although the PC and Macintosh versions were similarly vulnerable. Before we can explain the exact nature of this security hole we will need to cover some background information, such as SSL technology and random numbers. SSL is a certificate-based authentication and encryption scheme developed by Netscape during the fledgling days of e-commerce. It was intended to secure communications such as credit card transactions from eavesdropping by would-be thieves. Because of U.S. export restrictions, the stronger and virtually impervious 128-bit (key) version of the technology was not in widespread use. In fact, even domestically, most of Netscape's users were running the anemic 40-bit international version of the software. Most key generation, including SSL key generation, 'requires some form of randomness as a factor of the key generation process. Arbitrarily coming up with random numbers is much harder than it sounds, especially for machines. So we usually end up using pseudo-random numbers that are devised from mostly random events, such as the time elapsed between each keystroke you type or the movement of your mouse across the screen. For the UNIX version of its version 1.1 browser, Netscape used a conglomeration of values, such as the current time, the process ID (PID) number of the Netscape process and its parent's process ID number. Suppose the attacker had access to the same machine as the Netscape user simultaneously, which is the norm in UNIX-based multi-user architectures. It would be trivial for the attacker to generate a process listing to discover Netscape's PID and its parent's PID. If the attacker had the ability to capture TCP/IP packets coming into the machine, he could use the timestamps on these packets to make a reasonable guess as to the exact time the SSL certificate was generated. Once this information was gathered, the attacker could narrow down the key space to about 106 combinations, which is then brute force attacked with ease at near real-time speeds. Upon successfully discovering Netscape's SSL certificate seed generation values, one can generate an identical certificate for itself and either eavesdrop or hijack the existing session [4].

We have seen, with reference to the above given examples, that the key management problem (e.g., storage, generating, transmission), misapplied cryptography (incorrectly applied strong algorithm) can lead to major flaws in information security such as (1) Confidentiality (2) Data Integrity (3) Secure Availability of Information. As far as SSL is concerned, the idea behind it only provides a secure mechanism while the information is in transit from your computer to the server you are conducting the transaction with. After your credit card information safely arrives at the server, then the risk to that information changes completely. At that point in time, SSL is no longer in the picture, and the security of your information is totally based on the security mechanisms put in place by the owner of the server. If they do not have adequate protection for the database that contains your information, then it could very well be compromised. For example, let's say that the database on the server is Super Duper Database v1.0 and vulnerability has been discovered in that particular version that allows any remote user to craft a specific *GET String* to retrieve any table he or she may want. As you can see, SSL has nothing to do with the vulnerability within the database itself, and your information could be compromised. Here we give two case examples, which occurred due to inferior security approaches on the server side concerning sensitive data. One case example is the arrest of the Briton, Gary McKinnon, accused of hacking into computer systems at the PENTAGON and NASA, which threw the spotlight on the world of Hi-Tech crime. The problems he allegedly perpetrated cost more than £500,000.00 to track & correct. The second example is the case of Israel Atomic Research Centre in June 2005, which involved the theft of formula and data from the computer systems of the research laboratory. The data stolen related to Hydrogen Bombs. The case involves a data security breach in which top-secret data was stolen because of poor security strategy deployment on research computer systems. Securing the enterprise perimeter is not enough, it is also important to secure the data itself. In what follows, we will present the secure technique of Virtual Invisible Disk. This can present a technique to overcome three major issues regarding security: Confidentiality, Data Integrity, and Secure Availability of Information & Storage.

3 Virtual Invisible Disk

Virtually Invisible Disk design is an idea that can be considered to overcome the above-mentioned three major issues regarding Information Security. It is important before going into the discussion, how this idea can overcome these identified issues we will present some introduction and then further explanation about VID. This is an idea in the field of computer science, which is being introduced here, applying a special technique to create virtual invisible disk inside a virtual "Outer Disk/Container", onto the Operating System Disk, on Windows XP/2000/NT or UNIX. In order to access this designed approach, the idea defines a double key technique utilizing/incorporating biological information into secret crypto key for an "Outer Disk/Container" & VID. The first key is to give access to the visible "Outer Disk /Container", while other is to access the VID.

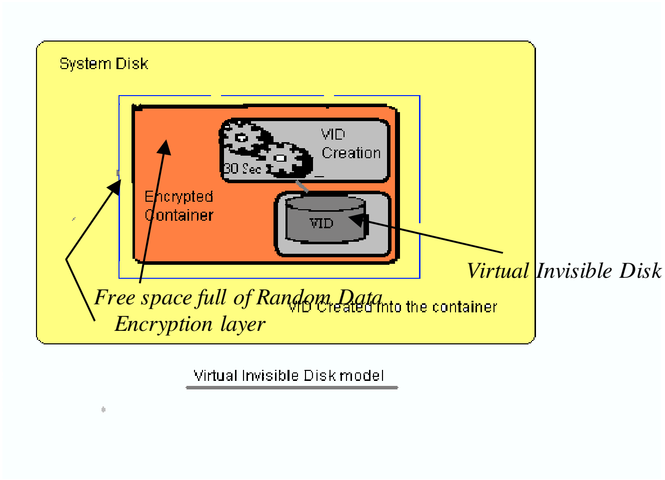
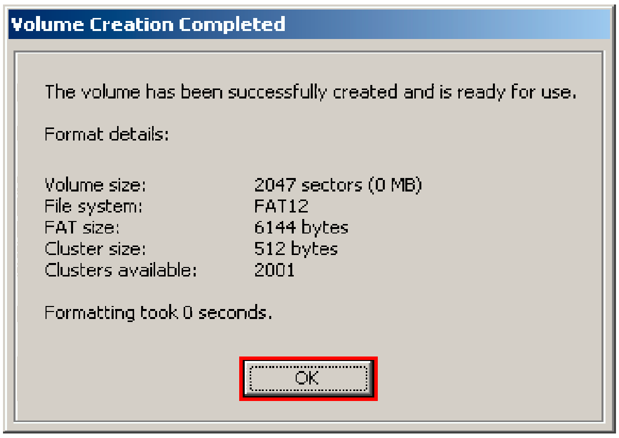

System Disk
VID
Creation
30 Sec
Encrypted
Container
VID
Virtual Invisible Disk
Free space full of Random Data
VID Created into the container
Encryption layer
Virtual Invisible Disk model

Figure 3: Virtual invisible disk model

This invisible disk key allows one access to the working disk that is invisible in the unused area of "Outer Disk/Container", while another key would allow giving access to the pre-setup volume (Outer Disk) in which data is stored that you want to convince others, was the only data in the Outer Disk. This technique is very effective especially in case of when someone forces you to use your biometric key. This results in the aggressor only seeing the fake generated data, while being unaware that there is another invisible disk that securely stores confidential data on the VID, encrypts "sector by sector" without generating temporary files and prevents the unauthorized access. This VID would not even show its properties and covered area by it, so no one can come to know whether there is another disk on the system's disk which contains all genuinely valuable data, secure from the eyes of intruders and aggressors. Now, we present information about how the technique works and the concepts behind it. In order, to create VID first need to make an Outer Disk/container, this will be able to specify the filename for an Outer Disk/container and its location. It is also important to set the Outer Disk/container filename format size of disk (disk size is dependent on the size of Outer Disk/container formatting system). The space used by the Outer Disk/container file is not overwritten on the disk and it creates hidden encrypted directory for security reasons, because directory on the hard disk is uniquely identified using two pieces of information: its filename, and the path along the directory tree that you traverse to get to it, also file table (FAT) which is used to keep track of which clusters are assigned to each file (As displayed in the given below example of True Crypt which reveals secret information to the Intruder, is a flaw in the vendors' implementations).



Volume Creation Completed

The volume has been successfully created and is ready for use.

Format details:

```
Volume size:          2047 sectors (0 MB)
File system:          FAT12
FAT size:             6144 bytes
Cluster size:         512 bytes
Clusters available:   2001
```

Formatting took 0 seconds.

Figure 4: This file will be a TrueCrypt container which reveals all secret information

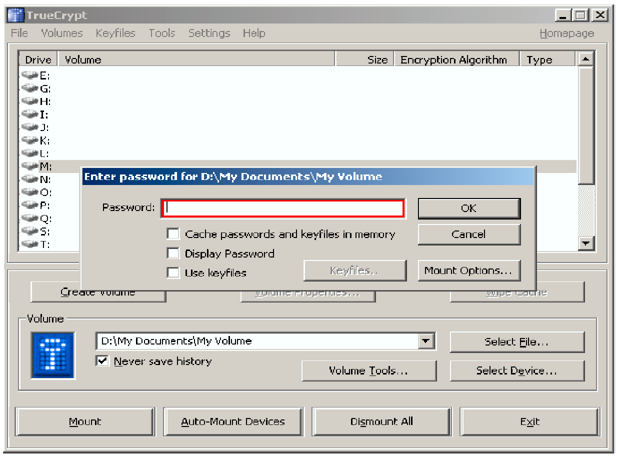
The operating system (and hence any software applications) can determine where a file's data is located by using the directory entry for the file and file allocation table entries. Similarly, the FAT also keeps track of which clusters are open and available for use. Therefore if it is hidden, in other words invisible then Accessing the entire length of a file which is done by using a combination of the file's directory entry and its cluster entries in the FAT could not be located by anyone, moreover, secure cluster engineering is the technique which would make it very much secure as that I stated above it generates hidden encrypted file's directory and dose not show property of where the container & then VID is generated therefore, it is impossible for intruders and aggressors to identify, Which means cannot be displayed even by using the DOS command line, so no one will be able to tell how many sectors have been allocated, at this stage user needs to assign a key which is incorporating biological information into a secret key for this Outer Disk/container to access, from the options of key method which is based on a multifactor biometric authentication method based on cryptosystem keys containing biometric signatures, for higher security it creates every volume differently. To do so it collects random data. By keeping the mouse on the left side of the Outer Disk/container screen for given calculation time (as stated in the VID Model) and ensuring it remains there until the random process complete, guarantees the randomness of this data. For encryption set up options can be chosen for different encryption algorithms. Typical choices might include the following.

Algorithms
AES 256, Triple AES,
Blowfish, Triple Blowfish,
RSA, Tea 16, Tea 32,
IDEA, DES, Triple DES.

By doing so the process of the encrypted Outer Disk/container completes, before creating VID it is very important to ensure that automatically generation of the fake data process must be completed, as mentioned to deceive the intruders. Once a standard Outer Disk/container is created (for instance with a size of 800 MB), It needs to mount it and let the process complete fill it with generation of fake data that user would not mind others finding in the case that the user is forced to use the key. It is important to bear in mind that the space which is using for fake data will reduce the available space that user will have on the invisible disk. For example if user is recommending to putting 60 MB of "fake" data generation on the created Outer Disk /container, the user will only be left with a potential Invisible Disk of around 740 MB.Therefore, at this stage by keeping the mouse at the right side of the designed encrypted Outer Disk /container for at least 30 seconds (as stated in the VID Model) it will start the VID creation process and it will start calculation of free space available for creation of the VID and mark the Outer Disk/container by default protected read only. This calculation is made by counting the biggest defragmented block of data starting from the end of the disk. This means that if user has a defragmented disk, the user will be able to use all the free disk space for the Virtual Invisible Disk.The Invisible Disk is stored in the free space at the end of an encrypted Outer Disk/container, as that the free space is full of random data it would be not possible to determine whether there is an Invisible Disk or not. When the mouse is kept for 30 seconds at the right side of the screen, a key window will pop up and user can apply its biological information based secret key to access VID. The purpose of this multi-time of setting in the Outer Disk/container, in which VID is created, is to ensure security.

3.1 Key Creation & Identity Management

The key session creation for VID is very important part of its security. Therefore, keeping in mind the above-discussed threats in Section 2, the key generating/choosing problems with passwords/pass phrase (as stated in the above examples PGP & True Crypt case which leads to weak security for key generation).



Enter password for D:\My Documents\My Volume

Password: |

□ Cache passwords and keyfiles in memory

□ Display Password

□ Use keyfiles

D:\My Documents\My Volume

☑ Never save history

Figure 5: Passwords/pass phrases & key files weak security

The concerns of brute force attack, biometric methods are utilized for verification. Since conventional biometric authentication methods have a serious problem - it is easy to steal the biological information of another person, such as fingerprints, iris patterns, and facial forms. Accordingly, it is essential to protect authentication systems from an attack like impersonation with an artificial finger. To solve this problem a multifactor biometric authentication method based on cryptosystem keys containing biometric signatures. It has two new aspects; one is to propose a method for embedding two kinds of data - personal biological information and a confidential item corresponding to a conventional secret key - into a cryptosystem key to detect any impersonation even if the biological information is stolen, and the other is to demonstrate the safety of the method, a factor which has been referenced by Okamoto [7].

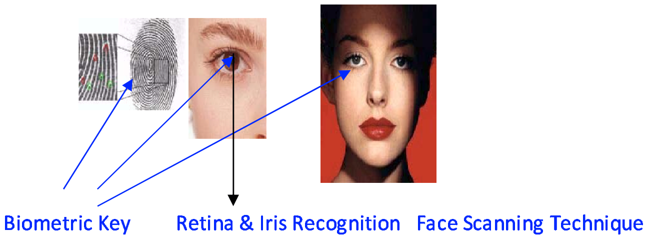


Figure 6: Biometric methods of key verification

Usually in the registration process on the system first acquires biological information from the individual to be registered. Of the resulting data, featuring points are extracted using a given algorithm. Next, the system records the featuring points as a template in a database for later personal identification. The database is called the biological information registration database (DB).

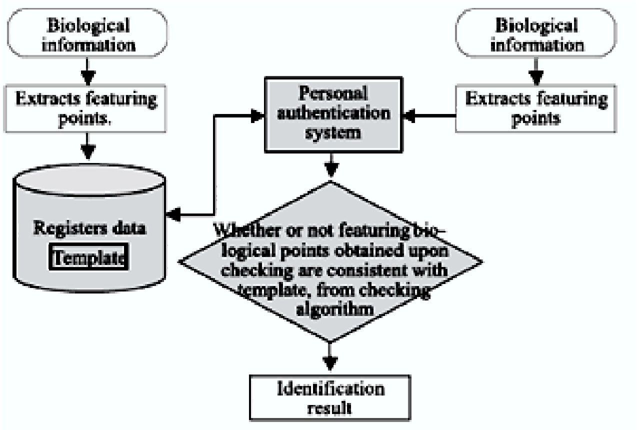

Biological information
Extracts featuring points.
Registers data
Template
Personal authentication system
Biological information
Extracts featuring points
Whether or not featuring bio-logical points obtained upon checking are consistent with template, from checking algorithm
Identification result

Figure 7: Personal authentication system based on biometrics

Therefore, a new system is being developed to carry out biometric personal identification by unlocking digital signatures or cryptographic communications using a corresponding secret key. This system first checks the biological information based on the algorithm mentioned above, and after the personal identification is successful, it makes the secret key recorded in VID effective to make calculations for the signature.

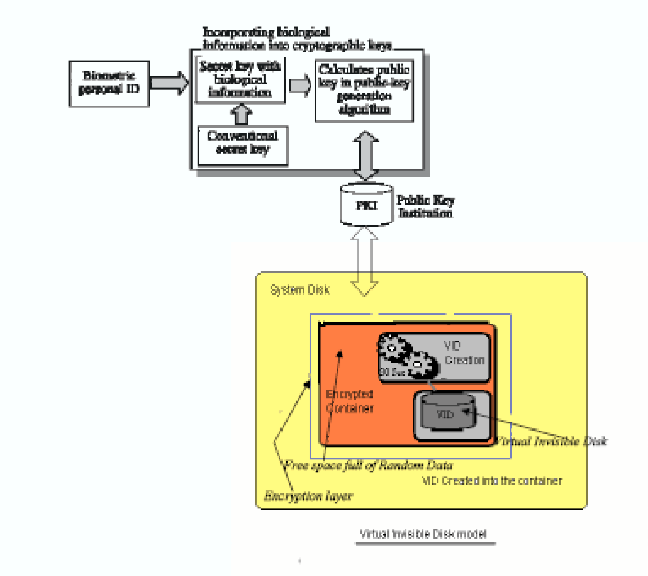

Incorporating biological information into cryptographic keys
Biometric personal ID
Secret key with biological information
Calculates public key in public-key generation algorithm
Conventional secret key
PKI
Public Key Institution
System Disk
VID Creation
30 Sec
Encrypted Container
VID
Virtual Invisible Disk
Free space full of Random Data
VID Created into the container
Encryption layer
Virtual Invisible Disk model

Figure 8: Incorporating biological information into a secret key to access the VID

The cryptographic keys consist of secret and public keys used in the public-key cryptographic scheme. Incorporating the biological information into the secret key means that the same data are automatically embedded into the public key generated in an algorithm defined by the system. After this, the cryptographic keys consisting of the secret and public keys into which the biological information is embedded are referred to as the biometrics-cryptographic keys.

As shown in above Figure embedding biological information in the cryptographic keys has the following advantages.

1) Privacy protection of personal information.

2) Zero knowledge, which means that no biological information is given directly to an inspector.

3) Economical system which doesn't need to build up its own biological database.

Assuming that δ is generated uniquely as a personal identifier, $s1$ is given by processing δ through a hash function, i.e., $s1 = H\ (\delta)$ is defined.

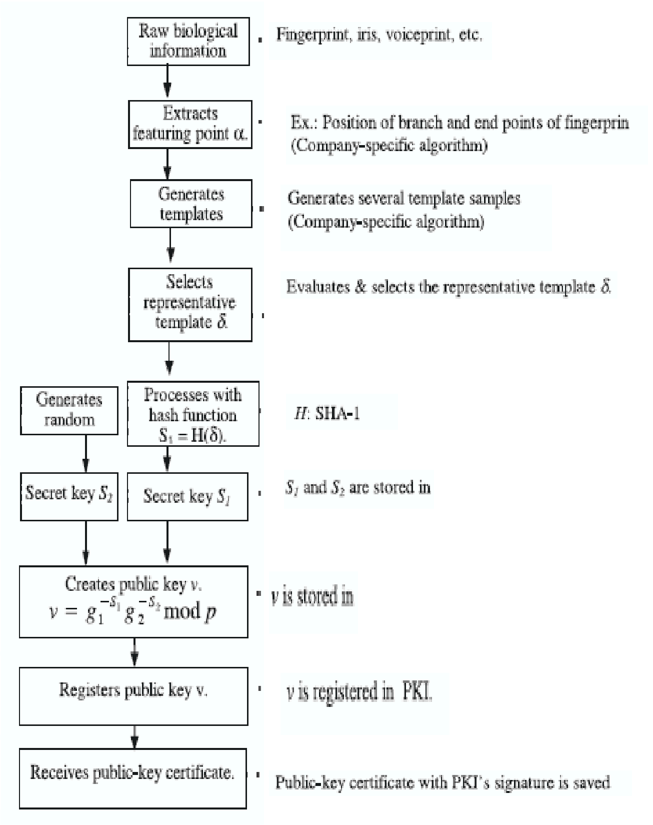

Raw biological information
Fingerprint, iris, voiceprint, etc.
Extracts featuring point α.
Ex.: Position of branch and end points of fingerprin (Company-specific algorithm)
Generates templates
Generates several template samples (Company-specific algorithm)
Selects representative template δ.
Evaluates & selects the representative template δ.
Generates random
Processes with hash function $S_1 = H(\delta)$.
H: SHA-1
Secret key S_2
Secret key S_1
S_1 and S_2 are stored in
Creates public key v. $v = g_1^{-S_1} g_2^{-S_2} \bmod p$
v is stored in
Registers public key v.
v is registered in PKI.
Receives public-key certificate.
Public-key certificate with PKI's signature is saved

Figure 9: Generating cryptographic keys having biological information

The $s1$ is a secret key having an individual's biological information. A candidate for the hash function H is $SHA-1$. In this case, the input data size is smaller than 264 bits, which is enough for the bit string size of the template. The data processed and provided by the hash function are 160 bits long, that is, a compressed key length of 20 bytes. If the biological information and template generating algorithm are stolen, the risk of forging $s1$ occurs. Therefore, the second secret key $s2$ is defined as a random number generated by conventional mathematical calculation, and the two secret keys $s1$ and $s2$

guarantee the security. The public key v is derived from a pair of secret keys $s1$ and $s2$. The created key v is registered in a public-key infrastructure (PKI).A public-key certificate with PKI's signature issued after the registration saved in the VID with δ, $s1$, $s2$, & v.It is also very important to keep in mind that *Database Security* is still of great importance as this prevents unauthorized access. Therefore for high security reason it is by default stored at secure location in the VID.

3.2 Key Management & Distribution Strategy

Another unique strategy that has been applied in this research VID functionality is a time out key session technique. This unique technique can provide higher-level security for the sensitive information, by restricting the key file. The purpose of a key file is to allow others access to an encrypted volume without having to reveal the key for the volume itself. One mounts all the encrypted volumes that they wish the other users to be able to access. It can also be impose further restrictions on the key file, such as the expiration date (key is valid only for a certain number of days), and/or the time during which it should be possible to use. *For example: Key may be used only during office hours, including break timings, such as from 8:30h to 1:30 & 2:30 to 17:30h.* The key file expires after a certain amount of days, by doing so it will allow access to the disk only between certain times as per given time frame and its number of days validity. The purpose of this level of approach is to provide top level of security for sensitive areas, e.g. in e-commerce, banking, e-government or military applications. The significance of these areas can be measured by the earlier provided case examples (NASA or Atomic Research Centre) regarding sensitive information and data security. Therefore, by applying this strategy we can defend sensitive data/information resources from within and outside against unexpected efforts to steal the valuable data.

4 Conclusion

Confidentiality and security are widely regarded as prerequisites for information storage and transmission, particularly in respect of the types of financial, e-government or military applications. There is clearly a need to apply techniques and approaches that engender confidentiality, information integrity, security and trust for both host and end-users of the system. This research describes a unique approach involving applying the technique of a "Virtual Invisible Disk". Utilizing & incorporating biological information into a secret key to Access the VID provides frontline defense for Virtual Invisible Disk. This can overcome the earlier stated issues & provide great confidence from the host system to end-users.

Future Research Work

The future research work is on the integration of additional functionality of VID with the combination of steganographic technique while exchanging sensitive information for different sector areas, such as the military or e-government.

Acknowledgments

The authors wish to thank the anonymous referees for their careful reading of the manuscript and their fruitful comments and suggestions.

Faisal Nabi would like to express special thanks to Dr. T. Okamoto, Yukio Itakura, Shigeo Tsujii (Institute of Information Security, Japan & Chuo University Research and Development Initiative) for their kind co-operation in this research.

Faisal would also like to thanks to Dr. Dale Grit & Madam Carol for their so much support & co-operation which made me able to conduct this research work being part of "Network Security Research Group at Department of Computer Science", Colorado State University, USA.

Special Note:

This research is based on the following message where by the idea conceived and the inspiration to develop the VID Concept

786
We shall dispose of you, O ye two dependents
(man & jinni),
O Company of jinn & men,
If ye have power to penetrate (all) regions of
the heavens (Sky/Space) & the earth,
Then penetrate (them)!
Ye will never penetrate them save with (our)
sanction,
Which is it, of the favour of your Lord, that
ye deny?
(Ref. Surah Ar-Rahman, Holly Quran)

Intellectual property, patent ship, copyright are reserved for this Research (VID) by USA Law, to the name of Junior Scientist Faisal Nabi.

References

[1] V. Ahuja, *Secure Commerce on Internet*, Academic Press Ltd, London, pp. 78, 1997.

[2] W. Cheswick and S. Bellovinm, *Firewalls and Internet Security Basics*, Addison-Wesley, 1994.

[3] Dhillon and j. Backhuose, "Current directions in IS security research; towards socio-organisationl perspectives," *Information Systems Journal*, vol. 11, pp. 127-53, 2001.

[4] D. Dong, *Using SSL As An Encryption" Tool*, June 7, 2002©.

[5] A. Furche and G. Wrightson, *Computer Money: A systematic overview of Electronic Payment System*, pp. 456, dpunkt Verlag Fuer digital technology Gmbh, Heidelberg, FDR, 1996.

[6] S. Garfinkel and G. Spafford, *Practical UNIX and Internet Security*, pp. 363-404, O'Reilly and associates, 1996.

[7] http://www.syngress.com/book_catalog/241_sscp/sample.pdf.

[8] C. E. Phillips, T. C. Ting, and A. Steve, "Information sharing and security in dynamic coalitions," in *ACM SACMAT'02*, California, USA, 2002

[9] S. Tsujii, Y. Itakura, H. Yamaguchi, A. Kitazawa, S. Saito, M. Kasahara (2000), "Public-key Cryptographic scheme having a structure in which biological information is embedded into a secret key," *IEICE Symposium (SCIS2000)*, D07, Jan. 2000.

Faisal Nabi Junior Scientist in the field of Information & Computer Security was born in Karachi City. He had his initial schooling, college & University degree level education from Pakistan. He went UK for higher education Oct 2001 Joined University of Luton, (Great Britain) M.Sc by Research in Electronic Commerce. Then he was selected for PhD in Network Security Student at Colorado State University, USA. He is specialized in e-commerce/information security. *Interest Area of Research:* Cryptography, Steganography, Virtual Invisible Secure Disk design, Secure Architecture, Web Security.

Started research R&D as an assistant Researcher with Professor: Carsten Maple 23 Nov 2003, Institute of Applied Research for Applicable Computing, University of Luton. In April 2004, he was appointed as Researcher at Deveraux & Deloitte Research Centre, UK. Faisal's Research work has been published in International Journals of IEEE/MCB level. Faisal has also received Research Award in the "*First Cyber Security Conference 29 Sept 2004 " at Sheraton Hotel, held in Karachi in conjunction with NR3C National Response of Cyber Crime.*