© 2012 **DSRC**

QUASI EXPERIMENT DESIGN FOR DETECTION OF SLEEPY OBJECTS OR NOISY OBJECTS

Pendyala Naga Lakshmi [1*], G.Subbalakshmi [2*] and CH. Raja Jacob [3*]

1. *M.Tech (CSE) Student, Dept of CSE, Nova College of Engg & Tech, Jangareddygudam, Dist: W.Godavari, AP, India*
2. *Asst. Professor, Dept of CSE, Nova College of Engg & Tech, Jangareddygudam, Dist: W.Godavari, AP, India*
3. *Assoc.Professor, Dept of CSE, Nova College of Engg & Tech, Jangareddygudam, Dist: W.Godavari, AP, India*

Keywords:
Access Control Algorithm, Authorization and lattice concept

Abstract: *In the era of Information technology, the terminology Data is a very highly difficult and the most wanted terminology of Computer Science. Data and Information seems to be the no difference words to the layman or the people of end user. Hence of this paper describes the Data, its security and if we consider its ability to cope with the error which in term we call it as the Robustness. Hence of this paper consist emphasizes on data security and its robustness .Robustness we can n't get have to achieve it .In this era of information technology, the most valuable and costly term is data. We derive a mathematical expression for the security of our algorithm, using which we show that the security of our algorithm can be increased independent of capacity, robustness and embedding induced distortion. The maximum security depends only on the length of the key sequence, which is limited only by the size of the host image. Using a joint security and capacity measure, we show that the proposed scheme performs better than current secure quantization based data hiding schemes.*

1. INTRODUCTION

News headlines about the increasing frequency of stolen information and identity theft have focused awareness on data security and privacy breaches and their consequences. In response to this issue, regulations have been enacted around the world. Data is a highly intensive terminology in this global information technology market. People spend billions of dollars to sell or buy the data either it may be in form of software. The term data should be secure if we go through the Internet working world , Because once we are keeping the data in the network it should and must be authenticated , if not should be privileged. Hence these context in mind I would like to write this paper ,hence of as of research is going on data security ,but comparing to these facts it is a continuous process ,so we will have to keep it as a research process like a maintains of a software.

**** Pendyala Naga Lakshmi***
M.Tech (CSE) Student, Dept of CSE, Nova College of Engg & Tech, Jangareddygudam, Dist: W.Godavari, AP, India

Covert communications, access control, ownership assertion and annotation are some applications of data hiding. There is another equally important parameter, namely the security of the hidden data. Several researchers have addressed this problem, but often not in conjunction with the other three parameters. In, the authors propose a lattice based embedding algorithm where security is achieved by randomly picking a set of host coefficients to embed. It is noted this algorithm can be made more secure by embedding smaller amount data . We addressed the trade-of between security and the other parameters and proposed a data hiding scheme using a lookup table to enhance the security of the embedded information. It was shown that the security of the LUT based algorithm is enhanced and that at lower WNR ranges, the LUT based scheme can achieve higher embedding rate than the odd-even embedding algorithm .

The probability of detection error could be considerably reduced below the traditional quantization based embedding, if we use a lookup table (LUT) of nontrivial runs that map quantized multimedia features randomly to binary data. However, this increased detection probability comes at the cost of increased distortion to the host.

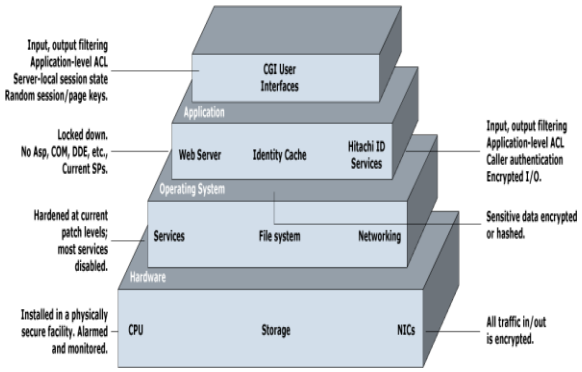

Fig. 1.1 Multi Layered Security architecture

2. RELATED WORK

The term “Security” describes the how much protection in terms of penetration testing. As of I used the terminology as the “Robustness”; this describes not only it’s safe guard as a stereotypic concept but also to its high end degree of data protection at each layer. The security architecture is not only considered as the typical but also highly and economically a high end concept. If the operating environment is not based on a secure operating system capable of maintaining a domain for its own execution, and capable of protecting application code from malicious subversion, and capable of protecting the system from subverted code, then high degrees of security are understandably not possible. While such secure operating systems are possible and have been implemented, most commercial systems fall in a 'low security' category because they rely on features not supported by secure operating systems (like portability, and others). In low security operating environments, applications must be relied on to participate in their own protection. There are 'best effort' secure coding practices that can be followed to make an application more resistant to malicious subversion.

The concept of data integrity from a security perspective deals with making sure that data is not subject to unauthorized alteration. If I can ensure data integrity, I can verify that the data hasn't been changed by some attacker. Data integrity is hugely important; in many applications, data integrity outweighs confidentiality. For example, for most users, an attacker's changing the balance of their bank accounts is of far greater concern than if a bad guy can see their balance. Although, I suppose whether the balance is changed upwards or downward has some impact on the user's concerns. ;)

Data integrity can be guarded by a variety of mechanisms. First, by keeping the attackers off of the system holding the information, data has some level of protection. Hardening systems, applying patches and utilizing host- and network-based intrusion-detection systems all help to keep the bad guys away from the data.

Another process that helps protect integrity is the regular back up. Systems, applications and stored data should all be backed up on a regular basis. For many systems, this should occur on a weekly or daily basis. If the data gets altered, backups are critical in restoring a trusted state.

However, these are only baby steps to ensuring real data integrity. To really protect integrity, cryptographic algorithms can help to create a secure digital fingerprint of the data. Integrity checking is often accomplished using hash functions, such as Message Digest 5 (MD5). On a Linux system (and some other flavors of UNIX), you can use the md5sum program to create

3. METHODS

The proposed method outlined in fig.1.1 comprised the interactive state phases. If we considered any of the layered architecture, as of coming to MVC in Java technology, it has a great lot high features, as of it shows state of transaction in among the phases. Access control in computer systems and networks relies on access policies. The access control process can be divided into two phases: 1) policy definition phase where access is authorized, and 2) policy enforcement phase where access requests are approved or disapproved. Authorization is thus the function of the policy definition phase which precedes the policy enforcement phase where access requests are approved or disapproved based on the previously defined authorizations.

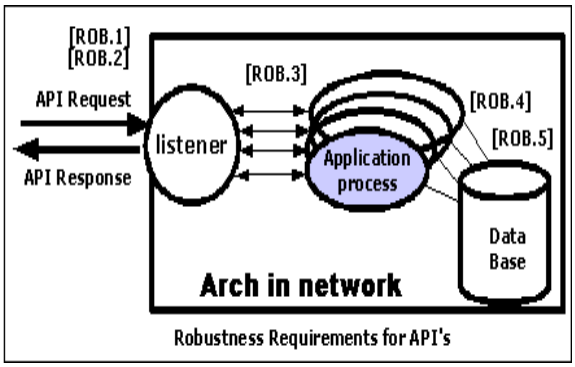

Fig.3.1 Showing the Architectural view of Data Robustness in Network

Access control also makes use of authentication to verify the identity of consumers. When a consumer tries to access a resource, the access control process checks that the consumer has been authorized to use that resource. Authorization is the responsibility of an authority, such as a department manager, within the application domain, but is often delegated to a custodian such as a system administrator. Authorizations are expressed as access policies in some type of "policy definition application", e.g. in the form of an access control list or a capability, on the basis of the "principle of least privilege": consumers should only be authorized to access whatever they need to do their jobs. Older and single user operating systems often had weak or non-existent authentication and access control systems.

"Anonymous consumers" or "guests", are consumers that have not been required to authenticate. They often have limited authorization. On a distributed system, it is often desirable to grant access without requiring a unique identity. Familiar examples of access tokens include keys and tickets: they grant access without proving identity.

Trusted consumers are often authorized for unrestricted access to resources on a system, but must be authenticated so that the access control system can make the access approval decision. "Partially trusted" and guests will often have restricted authorization in order to protect resources against improper access and usage. The access policy in some operating systems, by default, grants all consumers full access to all resources. Others do the opposite, insisting that the administrator explicitly authorizes a consumer to use each resource.

Organizations must take a holistic approach to protecting their information across the enterprise in physical, virtual and cloud infrastructures by:

I. Understanding where sensitive data exists
II. Safeguarding sensitive data in both structured and unstructured formats
III. Protecting non-production environments
IV. Securing and continuously monitoring access to the data
V. Demonstrating compliance to pass audits

As the news often tells us, data hacking is common, and hackers are becoming increasingly sophisticated in their methods. They consistently outpace IT professionals in identifying security weaknesses and successfully hacking systems to retrieve sensitive data in myriad ways. Once these breaches affect customers and reach the headlines, a company sustains reputational damage that affects the bottom line but can never be comprehensively measured in dollar amounts. But as sales numbers go down and stock value drops, the damage is nonetheless real, and frequently very significant.

The experience of the last 15 years, including dozens of now-famous security hacks, shows us that there is no certain method to keep data secure while a computer is in use. But well-planned security practices can prevent many hacks in a company, and diligence is crucial. Consistent internal education, current antivirus software and commonsense methods for preventing many potential data leaks all help narrow the security gap during a computer's life.

Computers, however, do not last forever, and to keep up with the ever-evolving demands of the business world, they must be replaced and their hard drives disposed of securely. When a computer is retired, there is a range of methods to protect the data on the hard drive. Determining which one is best for your business depends on the effectiveness and accessibility of each option.

If the data gets altered, backups are critical in restoring a trusted state. However, these are only baby steps to ensuring real data integrity. To really protect integrity, cryptographic algorithms can help to create a secure digital fingerprint of the data. Integrity checking is often accomplished using hash functions, such as Message Digest. On a Linux system (and some other flavors of UNIX), you can use the md5sum program to create the hash. A hash function takes a larger amount of data and crunches it down to a fixed length (usually on the order of a hundred bits or so). The hash function has a one-way nature. This means that given the hash result only, it's computationally very difficult to figure out what the original data was. Furthermore, it's extremely hard to find another set of data that has the same hash result.

I can use a hash function to create a fingerprint for each piece of data I want to protect. Then, I'll store the hash functions on a safe medium (such as a write-protected floppy disk, CD-ROM or another server). Periodically, I'll recalculate all the hashes of the active data to make sure they still match the hashes. If the data doesn't match the hash, I know it was altered. Then, I can roll it back to my previous value stored in my trusted backup.

Now, we don't want an attacker to manipulate our stored hash results. If they alter the data and the hash result, we won't be able to check the integrity of the data. Therefore, some applications apply the concept of a "keyed hash" or even a digital signature. For a keyed hash, the data is fingerprinted using a hash function that includes a secret key. If attackers manipulate the data and the hash, but don't have the secret key, they will not be able to calculate the proper hash. A digital signature uses public key encryption to digitally sign the hash to make sure it isn't altered. In a sense, both of these techniques are used to ensure the integrity of the hash itself.

4. CONCLUSION

As of Information technology is a dynamic changing word in this hot global market. Data security still and will also continue as hot cake in this current trend of Information technology. Robustness may not achieve in single algorithm, but it's a continuous process to achieve such a high trend. Hence of This paper gives emphasis on the two concepts of security is degree and level (may be robustness).

REFERENCES:

1. P. Bonatti, S.D.C. di Vimercati, and P. Samarati, "An Algebra for Composing Access Control Policies," ACM Trans. Information and System Security, vol. 5, no. 1, pp. 1-35, 2002.
2. F. Guo, J. Wang, Z. Zhang, X. Ye, and D. Li, "An Improved Algorithm to Watermark Numeric Relational Data,"
3. Information Security Applications, pp. 138-149, Springer, 2006.
4. L. Sweeney, "Achieving K-Anonymity Privacy Protection Using Generalization and Suppression," http://en.scientificcommons. org/43196131, 2002.
5. http://personal.stevens.edu/~ksubbala/Publications/nl-pka-kps-07.pdf
6. http://www.it-checklists.com/
7. http://www.wikipedia.com
8. patent/www.google.com

AUTHORS :

P.NAGA LAKSHAMI has received her Bachelor of Technology Degree inINFORMATION TECHNOLOGY from Sri Vishnu Engineering College for Women, Bhimavaram, W.G.District, Affiliated to J.N.T.U., Hyderabad. and Pursuing was M.Tech in COMPUTER SCIENCE & ENGINEERING from NOVA College of Engineering and Technology, Vegavaram, Jangareddygudam, West Godavari Dist, Affiliated to J.N.T.U., Kakinada, A.P., India.

Mrs.G.Subbalakshmi obtained her B.Sc in Computer Science from Andhra University. She received her M.Sc in Computer Science from Andhra University. She received her M.Tech in Computer Science and Engineering from JNTU Kakinada in 2011.

Currently she is working as Assistant Professor in the Department of Computer Science and Engineering, NOVA College of Engineering & Technology, Vegavaram, Jangareddygudem, West Godavari (Dist).

She is having 4+ years of teaching experience. Her areas of interest including Data Mining, Network Security, Information Retrieval Systems.

Mr.Ch.Raja Jacob, well known Author and excellent teacher Received M.C.A and M.Tech (CSE) from Acharya Nagarjuna university is working as Associate Professor and HOD, Department of MCA, M.Tech Computer science engineering , Nova college of Engineering and Technology, He is an active member of ISTE.he has 7 years of teaching experience in various engineering colleges. To his credit couple of publications both national and international conferences /journals . His area of Interest includes Data Warehouse and Data Mining, information security, flavors of Unix Operating systems and other advances in computer Applications.
(Mail id rchidipi@gmail.com)