International Journal of Science Technology and Management
Vol. No.6, Issue No. 01, January 2017
www.ijstm.com
ISSN (O) 2394 - 1537
ISSN (P) 2394 - 1529

CRYPTOGRAPHIC APPLICATION PROGRAMMING INTERFACES

S.Joseph Gabriel[1], M.Jawith Basha[2], P.Rizwan Ahmed[3]

[1]*Associate Professor & Head of Computer Science, Mazharul Uloom College, Ambur*

[2]*Research Scholar, Mazharul Uloom College, Ambur*

[3]*Assistant Professor & Head of Computer Application, Mazharul Uloom College, Ambur*

ABSTRACT

The significance of security devices that protect the numerous transactions, which take place in today's distributed virtual environment, cannot be underestimated. The importance of such devices will increase as our society continues to evolve into a cashless electronic society.

The continuing increase in the number and size of electronic transactions, the advances in the technology utilized and the growing sophistication of the adversary have led to significant resources being invested in the evaluation and analysis of security devices. There has been a transformation of the traditional security analysis from one focused on mathematical primitives and physical engineering solutions to a holistic approach that seeks to protect against subtle interactions between the cryptographic, logical and physical aspects of such devices that can collude to compromise the security thereof.

In the above setting, this dissertation investigates the electronic interface to security devices (i.e., the application programming interface or API) as a source of vulnerabilities. A number of innovative attacks are presented with significant implications for our financial transaction systems (e.g. ATM networks) that challenge previous assertions of their security, integrity and robustness. Finally, some design solutions and criteria are presented.

Index Terms— Cryptographic, DES, Electromagnetic Analysis

I. INTRODUCTION

Security devices are becoming increasingly pervasive in our modern society. Indeed, on closer examination of our interaction with both the physical and electronic world, one discovers numerous instances where we make use of electronic or virtual security devices (e.g. access tokens, electronic keys, gate remotes, subscriber identification modules (SIM cards), bank cards, credit cards and debit cards). Traditional security techniques have been augmented (and often completely replaced) by virtual methods. For example, traditional authentication by means of a written signature or face-to-face contact has evolved to include the use of digital signatures and biometrics. These changes are almost transparent, perhaps owing to the fact that the mechanisms employed often closely resemble the historical methods that we are accustomed to (e.g. an electronic key is conceptually identical to a physical key).

ISSN (O) 2394 - 1537
ISSN (P) 2394 - 1529

Perhaps the biggest change is the demise of our physical identities and the rise of our virtual personas. In our current Internet connected reality we can transact without a physical presence and through a distributed mechanism involving an eclectic collection of role players. These advances bring with them new security considerations, dangers and demands. Primarily the concern centres on our ability to authenticate and authorize a transaction, while at the same time ensuring the secrecy of the information (secrets) with which we achieve this. In the distributed scenario involving the use of infrastructure over which one has no control, one is particularly concerned about issues of trust in addition to the normal security requirements. The need to guarantee security in a hostile or untrustworthy environment is often also addressed through the use of security devices.

This work is predominantly concerned with the security of such devices – more specifically the logical security thereof. In order to reach the point were we can analyse the logical security in the general case and specific instances, we must first gain an understanding as to the design and workings of security devices.

II. TYPES AND EXAMPLES OF SECURITY DEVICES

There is a fairly wide range of devices that could be described as security devices. These range from the low end, low cost ICC (Integrated Chip Card or Smart card) and smart buttons, to the more powerful and costly, secure coprocessors and tamper responding security modules.

Example 1. Access control

The Dallas iButton is typically used as an access control device to control entrance to physical locations such as buildings and offices. It possesses a mechanism to authenticate itself to the control software, which can then authorize access to the bearer. In a sense, it is a key that opens a lock by electronic, as opposed to physical, means. A typical operating procedure would dictate that a token (button) only be issued to authorized people and the control system be configured to limit what access can be achieved with the button. The user is expected to take reasonable precautions to safeguard the device against accidental loss or theft. Theft of the device is an issue until the user can report it to the control system administrator. It is common practice for users to attach the buttons to their key rings (more evidence that they are perceived as conceptually equivalent) and hence the buttons are potentially transported, and exposed, to a large number of environments. The button represents a 'difficult challenge' to an attacker, who would attempt to steal or duplicate it. Thus, the security design requirements are to ensure that the secrets used to authenticate the device remain secure inside the device.
For our purposes, we shall limit our discussions to two categories of security devices, namely Integrated Chip Cards (ICCs), more commonly referred to as smart cards, and secure crypto coprocessors.

III. SECURE COPROCESSORS

A secure coprocessor can be known by a variety of names including:

- Tamper Resistant/Responding Security Module (TRSM),

ISSN (O) 2394 - 1537
ISSN (P) 2394 - 1529

- Crypto Accelerator,
- Network Security Processor (NSP),
- Host Security Module (HSM), and
- Hardware Security Module (HSM).

Essentially these devices provide a secure, trusted environment to perform sensitive operations. The exception is the crypto accelerator, which justifies its use purely on a performance basis by offloading the complex, time consuming cryptographic operations from the host system. These devices offer a degree of physical protection by being able to detect and respond to malicious attempts to recover key material or sensitive data. Typical measures include the use of a physical tamper envelope or membrane to detect physical intrusion, sensors for temperature and radiation as well as power supply monitoring and filtering. A detected tamper attempt causes the erasure of protected data. A thorough discussion on the physical security requirements is presented later.

A Simple Generalized Architecture

A simple design for a secure coprocessor consists of the following basic components:

- Communications interface
- Central processing unit (CPU)
- Crypto accelerators (including noise generating circuits)
- General purpose memory
- Memory for secrets

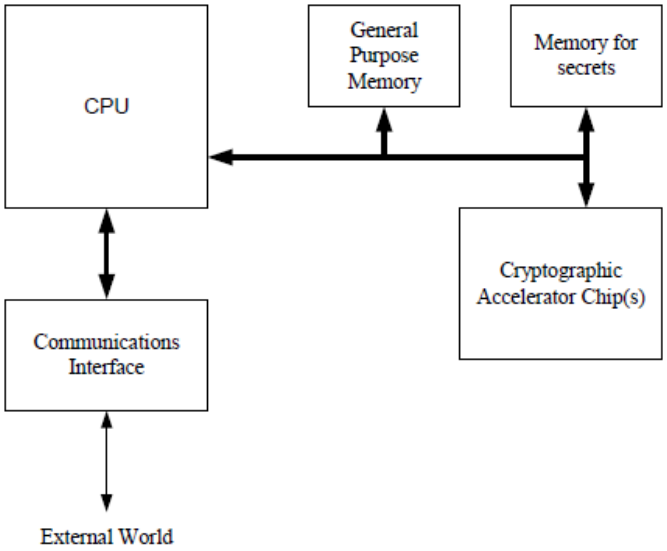

Figure 1-1: A simple design for a secure coprocessor

The CPU controls the general operation of the device. It communicates with the external world through the communications interface and makes use of memory to store program code, permanent information and temporary variables. Being a security device, the coprocessor is likely to contain some form of crypto accelerators, either as custom designed and built chips that implement the crypto or security algorithms in hardware or more general-purpose processors with optimised code implementations of the algorithms. The use

ISSN (O) 2394 - 1537
ISSN (P) 2394 - 1529

of cryptographic chips is largely driven by performance requirements and the fact that crypto algorithms are typically computationally intensive. Hardware generation of noise for the seeding of random number generation may be required owing to the fact that software noise is at best pseudorandom. Designing a noise source that generates genuinely random noise can be challenging task in itself. In addition, there may be a special area of memory used for storing secrets and other sensitive data, which may require special characteristics such as the ability to actively detect secrets while on standby power. While this description implies a hardware solution (i.e., a physical device), one can conceive of a software only solution that executes on some host system.

IV. TOWARDS DEFINING THE SECURITY GOALS

It is an axiom of security engineering that a system is only as strong as its weakest link. Despite this fact, it can be argued that real world systems typically fail to consider completely all aspects of security. Often a designer will focus on those areas about which he is knowledgeable or comfortable. This is perhaps a natural phenomenon. After all, one cannot consider topics or issues about which one is not aware. Towards the goal of describing a suitable security target, [We00] provides a useful and descriptive subdivision of security concerns relating to secure devices as follows:

- Physical security, a barrier placed around a computing system to deter unauthorized physical access to the computing system itself,
- Logical security, the mechanisms by which operating systems and other software prevent unauthorized access to data,
- Environmental security, the protection the system receives by virtue of location such as guards, cameras, badge readers, access policies, etc.

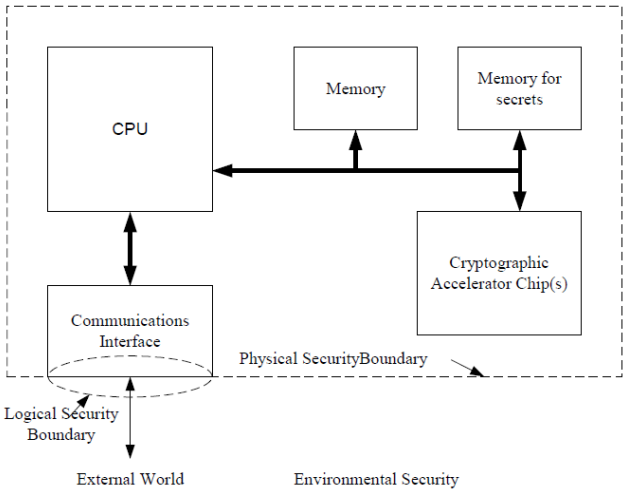

Figure 1-2: Security boundaries

ISSN (O) 2394 - 1537
ISSN (P) 2394 - 1529

V. CONCLUSION

The IBM attack is the attack of choice. The 'DES cracking machine' is actually irrelevant to this attack scenario. The effort required is trivial and the time required is minimal. This represents a serious threat if one can overcome the requirements of the attack and is able to mount it.

REFERENCES

[1] R. J. Anderson and S. J. Bezuidenhoudt, "On the Reliability of Electronic Payment Systems." IEEE Transactions on Software Engineering, v 22 no 5 (1996), pp 294-301. Available from http://www.cl.cam.ac.uk/ftp/users/rja14/meters.ps.gz

[2] R. J. Anderson and M. Bond, "API-Level Attacks on Embedded Systems." IEEE Computer Magazine October 2001, (2001), pp 67-75.

[3] R. Anderson and M. Kuhn, "Tamper Resistance - a Cautionary Note". Proceedings of the Second USENIX Workshop on Electronic Commerce, 1996, pp 1-11.

[4] R. J. Anderson and M. G. Kuhn, "Low Cost Attacks on Tamper Resistant Devices." Proceedings of the 1997 Security Protocols Workshop, Springer LNCS, v 1361 (1998), pp 125-136.

[5] R. J. Anderson, "UEPS – A Second Generation Electronic Wallet." Computer Security – ESORICS 92, Springer LNCS, v 648 (1992), pp 411-418.

[6] R.J. Anderson, "Why Cryptosystems Fail." Communications of the ACM, v 37 no 11 (1994), pp 32-40. An earlier version is available at http://www.cl.cam.ac.uk/users/rja14/wcf.html

[7] R. J. Anderson, "Making Smartcard Systems Robust." Proceedings of Cardis 94, 1994, pp 1- 14.

[8] R. J. Anderson, "Liability and Computer Security: Nine Principles." Computer Security – ESORICS 94, Springer LNCS, v 875 (1994), pp 231-245.

[9] R. J. Anderson, "The Correctness of Crypto Transaction Sets." Security Protocols – 8th International Conference, Springer LNCS, v 2133 (2000), pp 125-127.

[10] R. J. Anderson, Security Engineering – A Guide to Building Dependable Distributed Systems, Wiley, New York (2001), ISBN 0-471-38922-6