

## ENCRYPTION SOFTWARE – ARE YOU SITTING ON A SECURITY TIME BOMB?

Encryption software poses a real security risk. There seems to be a misunderstanding about how secure “encryption” software is. This is critical to organizations with data worth protecting whether it is sensitive customer or personal data, IP, or internal corporate governance data. Encryption software is a good solution where minimizing risk is not a primary concern. It is problematic where risk minimization is a priority or the priority is to prevent the occurrence of already identified weak points in your security architecture.

Hardware encryption provides far stronger protection where risk of exposure is a serious consideration. Until recently hardware solutions in the form of hardware encryption appliances were custom built, relatively expensive, and very often bulky and not amenable to deployment to end users.

Recently a small number of vendors have brought to market specialized ASICs which deliver tremendous value for money. They combine a small form factor and cost effectiveness with industrial strength encryption/decryption services. Each vendor has adopted different design methodologies and incorporated more or less useful features providing more effective solutions to the software encryption vulnerabilities mentioned below.

A January 2006 IT Architect article was entitled, **“Could 2006 be the year that security software vulnerabilities enable malware to compromise target computers?”** Andrew Conry-Murray discussed the threat, commenting that **security software is a prime target for attackers because it is usually the first inline.** He also points out that: “Anti-virus software isn't the only potential target. This past November a software vulnerability was discovered in **Microsoft's** anti-spyware beta that could potentially allow the execution of arbitrary code. Exploitable vulnerabilities were also uncovered in **Snort**, the popular open-source IDS software.”

**The 2006 SANS Institute list of the top ten vulnerabilities** in cross platform applications was headed by Backup Software, Anti-Virus Software, PHP-based applications (50% of Apache Servers run PHP), and Database applications. They also mention flaws found in front line security software applications from providers including, Computer Associates, RSA, HP, Sun, and Novell.

### **Symantec Press Release in August 2006:**

Symantec has warned of a new vulnerability discovered in its On-Demand Agent that, in the eventuality of an exploit, would permit a local attacker to decrypt files on the target machine. The security company has already addressed the situation and provided fixes for the vulnerability prior to its disclosure.

Although encryption software usually operates behind the front lines the threat is that once an attacker has gained access to a system, for example with administrator rights, they can access the keys and manipulate access control rights to decrypt data.



***The fundamental problem is that security software is written by corporate programmers who are always under the gun to get the job done and the product on the market! Time to market is as important to the security software developer as anyone else with shareholders and other investors to satisfy. Dollars wait for no man and once the feature set is stable the product is headed for the door.***

**Internal threats.....keys can be found in software.....and from your fellow employee!**

It is commonly understood that the majority of security breaches are internal. According to the FBI and the Computer Security Institute, 50-80% of all attacks happen from inside company firewalls. Of those companies surveyed, 73% of them reported that they'd experienced some form of internal security breach over the past year.

Within a consolidated data centre there is often little separation between the people that manage the data storage devices and the information that sits on it, providing them with unrestricted access to sensitive data. This means that an organization's security may be well ordered from a user perspective but wide open to third parties such as short term technical consultants and technical staff within the data centre.

### **Specific Software Encryption Problems**

- ***Operating System.*** Software encryption runs on top of an operating system with unknown security flaws. Although operating system vendors are constantly improving the security of their products the never ending feature creep and complexity is almost certainly generating more security vulnerabilities than are being fixed.
- ***Memory Space.*** Software encryption makes use of shared memory space. Software solutions do not facilitate their own physical memory. Software implementations are making use of externally available memory through services of an underlying operating system. When the memory that is used by the application is provided externally, there is no guarantee about the safety of that memory space. Most operating systems give some sort of random access memory space protection, but it must be remembered that this protection is only guaranteed by the robustness of the operating system and its freedom from flaws. Memory protection is even more difficult and lacking where secondary memory is used. Therefore, the security level of a software-based cryptographic module is upper-bounded by the security level of the mechanism(s) that protects the secrecy and integrity of the memory space it uses.
- ***Overall Ease of Manipulation.*** Software is based on a set of instructions that are stored in memory and are executed upon demand or prior instruction. Since the protection of secondary memory is not guaranteed, the integrity of the code itself cannot be guaranteed either. An adversary can modify the application code to cause it to malfunction or to cause it to leak critical information. Software code alteration can be done either manually, by changing specific instructions, or in an automated manner using hostile code such as a virus or a Trojan horse running

on the same platform or other platform which has adequate access privileges.

**Example:** A specific case, which is relevant to ARM processors, is that by using the GP-IO1 that is unique to ARM processors during power-up, the bus can be used to reflect memory contents and assist in modification of code that is stored on the host.

- **Reverse Engineering.** Software code can be read and reverse engineered. Although the use of publicly known algorithms and a first class implementation can reduce the risk of a full blown system crack it does not reduce the possibility of a “damage attack” where the primary goal is to inflict damage on the victim.
- **Key management and security.** Key management is a serious problem for software encryption because of the local operation of the encryption services. Keys proliferate exponentially and their management becomes a real burden for IT executives. The problem of securely hiding keys is also a very serious problem because the two most commonly used solutions are both weak. Enforcing user passwords with enough entropy to provide a sufficiently powerful key is a recipe for a helpdesk disaster. For example, a five character password delivers a 17 bit key. Encrypting the long term key with an internal key stored in the application brings us back to the issues of the memory spaces and operating system vulnerabilities.

### Specific Hardware Encryption Advantages

- **Key Management.** Keys are managed in a fundamentally different way and can provide protection against viruses, Trojans, and other malicious attacks if the hardware design is implemented correctly. A successful attack can gain access to software encryption keys. Hardware key management solutions significantly strengthen the safety of the encryption keys. A well designed hardware implementation has more effective key hiding solutions. Internal keys can be burnt as a part of the hardware implementation making them extremely difficult to extract. The internal key can also be stored in non-volatile memory, which is made inaccessible to other applications by hardware means.
- **Code Manipulation Protection.** Hardware-based solutions are safer in this respect as the code is burnt onto a chip. Physically burning the source code is probably the only proof way of causing it to be completely read-only, as source code of any cryptographic module should be.
- **Attack Resistance.** Brute force attacks are almost impossible because of the need to remount the disk for every new key guess.
- **Memory Space Protection.** Hardware-based solutions can contain their own internally managed memory space, which solves the problem of memory-space protection. Furthermore, hardware solutions can be applied, for memory illegal



access prevention, by hardware methods, which are inherently more secure than operating system services that are software-based in their nature.

### Other Benefits of Hardware Encryption

- **Speed of operation** delivers real time experience for users. The best hardware encryption implementations can encrypt and decrypt data at AES 256bit levels of security in real time and take advantage of storage protocols such as SATA.
- **Exceptional Bandwidth** delivers server side performance. Once again the best hardware encryption products can achieve bandwidth performance in the 3Gbps range enabling the real time read and write operation to as many as 5 or more SATA II drives.

The conclusion is that if you are seeking a robust, reliable, and, most importantly, industrial strength software encryption solution then you need to implement a system based on hardware encryption. With regulation getting more and more detailed in its attention to the way security protections are implemented we may well see software encryption be subject to invalidation if security flaws are found (which they will be) with the resulting exposure to law suits and the high cost of remedial action.

Enova Technology's X-Wall encryption ASIC family has a particularly rich feature set including NIST level algorithms up to 256 AES, very high bandwidth, interfaces to the leading storage control protocols, and an excellent price/performance profile. In addition the X-Wall family is able to provide a full range of authentication support including Common Access Card two factor authentications as well as RFID based tokens. Enova is currently finalizing the documentation for FIPS 140-2 certification.

The design has been validated by selection by world leaders in storage and mission critical applications after intensive field testing and in 2007 Enova will introduce a version with an integrated Trusted Platform Module (TPM) 1.2 that greatly extends the key management capability.

For further information please visit:

[www.enovatech.com](http://www.enovatech.com)

Or contact:

John Bourgein  
VP Marketing  
Tel: 925.276.8772  
Cell: 925.212.2599