A PC CRYPTOGRAPHIC COPROCESSOR BASED ON TI SIGNAL PROCESSOR AND SMART CARD SYSTEM

Milan Markovic[1], Zoran Savic[2], Željko Obrenovic[3], Aca Nikolic[4]
[1]*Mathematical Institute SANU, Kneza Mihaila 35, p.f. 367, 11001 Belgrade, Yugoslavia*
[2]*National Bank of Yugoslavia, Clearing and Payment Service, Pop Lukina 7-9, 11000 Belgrade, Yugoslavia*
[3]*Faculty of Electrical Engineering, University of Belgrade, Bulevar Kralja Aleksandra 73, 11000 Belgrade, Yugoslavia*
[4]*NetSeT Co., Karadjordjeva 65/4, 11000 Belgrade, Yugoslavia*

1. INTRODUCTION

Data encryption and other cryptographic functions are of the vital interest for computer networks to ensure the security of private and sensitive information. Most modern network security protocols are software-only based systems [1].

The existence of hardware security modules is ultimate for design of the computer network system with high performance and high level of the security. In that case, the encryption algorithms and the other security related functions (e.g. access control functions) are securely executed on the hardware element and sensitive data are never loaded on the user's computer memory. Without these modules, it would be not possible to achieve the trusted aplication concept with a full control of the system access and resistant to the Trojan Horse attacks [2]. It is proven that PC operating systems and other system software components have some security drawbacks, and this is especially critical for the software-only cryptographic tools.

This paper is dedicated to the conceptual description of NST2000 (NST - Network Security Technology), a SW/HW security coprocessor solution, intended to be used for realization of the proprietary or standard security protocols (e.g. SSL [1]) in the computer networks with "client-server" architecture. Also, NST2000 could be used as the security module for a local data archive protection. It is based on a signal processor from Texas Instruments TMS320 family, a peripheral controller PIC 16F87x and a smart card system (e.g. Gemplus GPK 4000 [3]). NST2000 consists of the software interface and the hardware board - PC-based security coprocessor module with a keyboard and the smart card reader control. A sample application that uses a first prototype of the NST2000 solution is described.

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35413-2_36

R. Steinmetz et al. (eds.), *Communications and Multimedia Security Issues of the New Century*

This paper is organized as follows. Section 2 is dedicated to a brief description of the main characteristics of the NST2000 solution. Hardware and software parts of the NST2000 are described in Sections 3 and 4, respectively. A sample application is described in Section 5, while conclusion is given in Section 6.

2. NST2000

As we stressed out, the hardware security modules, realized as coprocessors, represent strong points of the modern security solutions for the computer networks.

Such security products that implement time critical cryptographic functions are available in the market and range from PCI-based coprocessor modules, such as: CryptoSwift from Rainbow Technologies, HSP4000 from Baltimore Technologies, Luna VPN from Chrysalis-ITS, etc, to PCMCIA card-based solutions, such as: FORTEZZA card from Mykotronx, Luna 2 from Chrysalis-ITS, etc. The basic function of these products is in accelerating the cryptographic functions. Namely, these products are designed for the real-time realization of the symmetrical and asymmetrical cryptographic algorithms and other crypto functions according to common standards, such as PKCS [4].

NST2000 represents the SW/HW-based security solution for implementing the trusted system concept in the modern networks. The conceptual model of the NST2000 is shown on Fig. 1. The additional security features make this board, not only the cryptographic processor, but also the real security module that provides integrity protection of the critical applications and the system software. NST2000 is a point of confidence that implements strong security protocols isolated from malicious programs such as viruses and Trojan Horse programs. That confidence can be delegated to the rest of the software and the hardware components.

Besides all basic functions that are also implemented in all of the abovementioned security PCI and PCMCIA boards, NST2000 has implemented additional functions, such as:

- Microcomputer based control of the input data from the keyboard and from the smart card module,
- Random key generator based on the physical process,
- Direct interface to the smart card with identification and the other security parameters independent of PC driver software,
- Flexibility regarding a possible adding of new standard or proprietary cryptographic functions in the existing environment, and
- Programming interface for the secure applications, e.g. integrity protection of the PC operating system and the critical application.

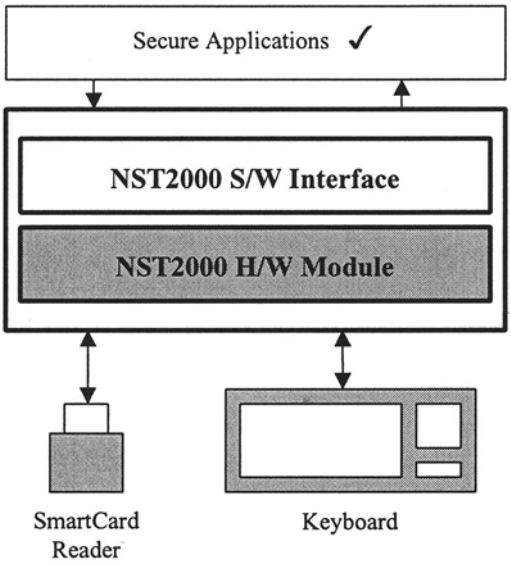

Figure 1. Conceptual model of NST2000. NST2000 represents the SW/HW-based security solution for implementing trusted system concept in contemporary networks. Secure applications use NST2000 via NST2000 software interface. NST2000 H/W module implements strong cryptographic functions. The keyboard and the smart card reader are connected to the NST2000 H/W module.

3. HARDWARE ARCHITECTURE OF NST2000

The NST2000 hardware consists of the following components:

- Signal processor of TI TMS320 family (TMS320C2xx 40 MHz, TMS320C54x 100 MHz or TMS320C62x 300 MHz family),
- Memory subsystem,
- Specialized I/O controller PIC 16F87x,
- Standard ISA or PCI bus interface for the host PC communication, and
- Smart card reader system.

A simplified block-diagram of NST2000 is shown on Fig.2. The main functions of the board processors and memory are:

- Implementation of the basic set of the cryptographic functions according to PKCS#11 standard [5],
- Implementation of the asymmetrical crypto algorithms (RSA or DSA) and hashing algorithms (SHA-1, MD5),
- Implementation of the standard (DES, 3DES, IDEA, RC4, ...) or the user defined symmetrical crypto algorithms,
- Hardware-based asymmetrical key pair generation,

- Communication with external devices (keyboard and smart card reader) through the specialized I/O controller,
- Communication with the Windows-based interface software running on the host PC workstation,
- Trusted path to the smart card memory for storing the cryptographic parameters such as symetrical and asymetrical keys, certificates, PIN codes, etc.
- Storage medium for all other sensitive data and programs such as cryptographic parameters (algorithms, certificates, symetrical and assymetrical keys, etc.), and
- Fast erasing of stored cryptographic parameters in emerging situation.

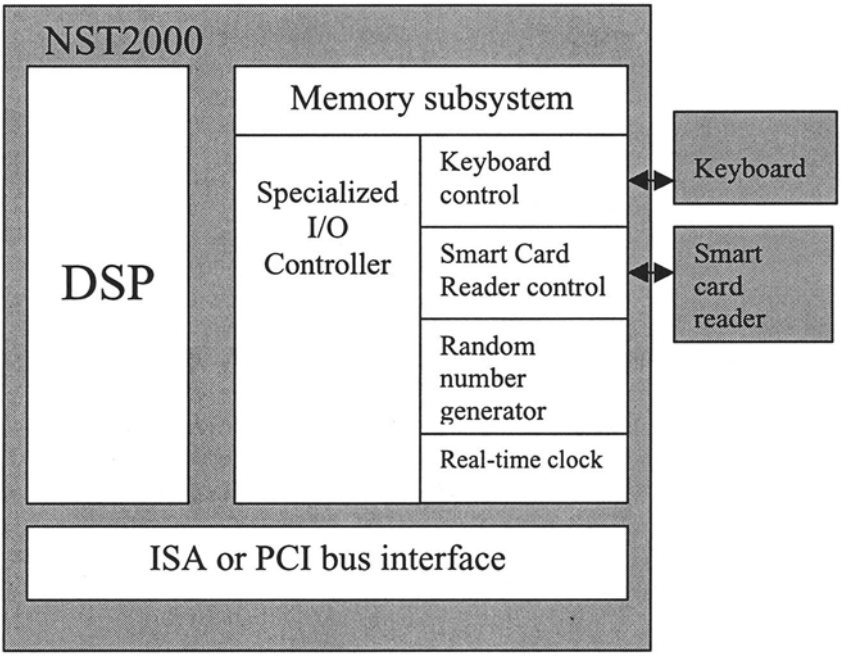

Figure 2. A simplified block-diagram of NST2000

NST2000 security module has three versions, depending on the applied signal processor.

TMS320C2xx (C2xx) [6] generation of signal processors brings the power of digital signal processing (DSP) to designers of high-performance, cost-sensitive, and emerging applications. Combining up to 40 MIPS performance with ultra-low cost, the C2xx generation offers the most optimal balance of price and performance of any DSP in the industry.

NST2000 version based on the C2xx signal processor is intended to use on the client side.

TMS320C54x (C54x) [7] family of DSPs offers the optimal combination of high performance, peripheral options, smal packaging and lowest power dissipation in the industry to give designers an edge in today's wireless and wireline communication markets. The C54x DSPs (30-200 MIPS performance) deliver the performance demanded by wireless communications aplications like cellular phones, pagers, PCS terminals, as well as emerging applications like Internet Protocol (IP) phones and portable information appliances. NST2000 version based on the C54x family of signal processors is intended for use on the server side for relatively small size network architectures.

TMS320C62x (C62x) [8] represents a family of fixed-point signal processors with the highest CPU performance in the market. TI delivers the revolutionary VelociTI™ advanced Very Long Instruction Word (VLIW) DSP architecture and a combination of the most up-to-date hardware and development tools that all C62x devices have in common. The C62x family of signal processors supports next-generation applications, such as 3G wireless base stations, high-end telecommunications and networking infrastructure, remote access servers, digital subscriber loop (xDSL) systems, imaging applications and multi-channel telephony systems. NST2000 version based on the C62x signal processors is intended to use for servers in the medium and large network architectures.

The specialized I/O controller PIC 16F87x and related hardware suports the following functions:

- Keyboard input control – the keyboard is connected to the PC via the NST2000 board,
- Smart card reader control – it is also directly connected through the NST2000 board,
- Hardware realized random number generator (RNG), and
- Reliable real-time clock for secure time-stamp generation thus preventing the replay attacks.

The keyboard control suppress monitoring of the all users data such as passwords, PINs or secret application parameters. The I/O controler implements the standard keyboard interface protocol and the input data are sent to the DSP processor. The inherently. insecure PC software has no influence to this process.

The similar idea is applied for the smart card data input. The interface to the smart card is realized in the I/O controler and the smart card reader connected directly to the NST2000 board. The serial communication between the smart card reader and I/O controler is implemented. This is the basis for implementation of the high secure authentication protocols.

The hardware based RNG is used for generation of the random initialization data and the message keys in some class of cryptographic algoritms as well as for random challenge generation in billateral challenge-

response authentication protocols. This component is used also in the process of the asymmetrical keys generation.

The reliable real-time clock is used for various protocol schemes for replay attack elimination and for secure time-stamp generation, and could be used in authentication protocols and digital signing functions. Beside this, it provides the reliable data for the message validity (e.g. certificates validation) and it is the basis for secure administration of the cryptographic parameters in the system. This real-time clock represents the "trusted time" which could not be altered via the software of the host PC computer.

As for the examples of the NST2000 timing performances in cryptographic functions realization, we have done some preliminary experiments on the 'C54x version of NST2000. In this sense, the minimal and maximal NST2000 execution times for private key operation of the RSA assymetric cryptographic algorithm for the range of the 'C54x family signal processors (from TMS320C541 (33 ns cycle) to TMS320C5420 (5 ns cycle time)) are given in Table 1. In these experiments, RSA modulus was 1024 bit length (n=1024) and private key exponent was 512 bits (d=512).

Table 1. RSA execution times for 'C54x signal processors

Cycle (ns)	Execution time (ms)	
	Maximum	Minimum
33	1411.05	1189.80
25	1068.98	901.37
20	855.18	721.09
15	641.39	540.82
12.5	534.49	450.68
10	427.59	360.55
8.3	354.90	299.25
6.25	267.25	225.34
5	213.79	180.27

Further experiments have shown that the RSA execution times, presented in Table 1, could be decreased for 10% if the modified Karatsuba-Offman algorithm [9] is used for multiplication. Also, the experiments based on the use of CRT (Chinesse Remainder Theorem) [9], have shown that the given RSA execution times could be further decreased from 3 to 3.5 times.

As for the symmetrical cryptographic algorithm example, the corresponding execution times for IDEA algorithm operations (creating encryption key (627 cycles), creating decryption key (13116 cycles) and IDEA encrypt/decrypt operation in ECB mode (2584 cycles)) are given in Table 2. As we already said, the experiments presented in Tables 1 and 2 are only preliminary experimental results obtained by using the NST2000 prototype. However, based on the presented experimental, results, we could

conclude that the NST2000 solution is very promising for realization of the complex cryptographic algorithms.

Table 1. IDEA ECB realization time for C54x signal processors

Cycle (ns)	Realization time (μs)		
	Creating Encryption key	Creating decryption key	IDEA ECB operation (encrypt or decrypt)
33	20.69	432.83	85.27
25	15.68	327.90	64.60
20	12.54	262.32	51.68
15	9.40	196.74	38.76
12.5	7.84	163.95	32.30
10	6.27	131.16	25.84
8.3	5.20	108.86	21.45
6.25	3.92	81.98	16.15
5	3.14	65.58	12.92

4. NST2000 INTERFACE SOFTWARE

The software part of NST2000 encapsulates functionality of the NST2000 hardware module, and provides the uniform interface to applications that use the NST2000 solution. The software interface includes the cryptographic functions interface according to PKCS#11 standard [5]. A cryptographic token is in this case the combination of the coprocessor board functions and the smart card functionality.

We have developed interface for native Windows NT applications and for Java applications (Fig. 3). The interface is implemented as Dynamic Link Library (DLL) module. Java applications use this module via Java Native Interface (JNI) adapter that just route function calls to the native interface. A security administrator interface is also a part of the NST2000 software interface.

It is important to note that all the cryptographic related functions are implemented and executed on the NTS2000 hardware module, not in the software interface. We have developed the SW part of the NST2000 solution according to the well-known "trusted application" concept. Namely, only the trusted application (SW part of the NST2000) can communicate with the NST2000 hardware board. This is provided by using a special authentication protocol between the trusted SW application and the hardware board. In the other words, a special internal cryptographic tunnel is established between the SW and HW parts of the NST2000. This prevents that no malicious programs (viruses and Trojan Horses) can compromise this kind of the trusted program-to-program communication. This concept eliminates potentially weak point that is inherent to the mixed smart card

and application software based implementation of PKCS#11 functions. The application is sure that, for example, a public key based digital signature verification process is implemented securely and that it can trust to their results. This is not true for software-only based signature verification process because some other malicious software can compromise it.

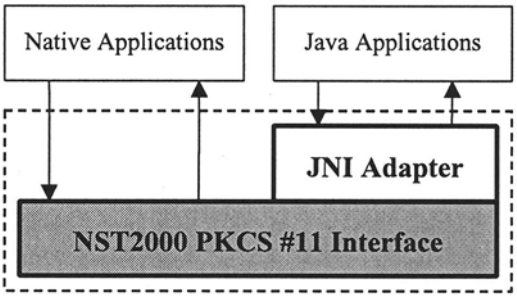

Figure 3. Simplified model of the NST2000 software interface. Software interface is based on PKCS #11 standard. Native applications can directly use this interface, and Java applications have to use Java Native Interface (JNI) adapter.

This software interface can be used for implementation of the NST2000 based TCP/IP secured protocols such as [8]: Secure Socket Layer (SSL), Transport Layer Security (TLS), Secure HTTP, S/MIME etc. The proprietary secured communications protocol could also be implemented.

5. SAMPLE APPLICATION: SECURE JAVA CLASS LOADER

To demonstrate functionality of NST2000 we have developed a secure Java class loader sample application. (Fig. 4). This sample application demonstrates functions of the NST2000 module and demonstrates the use of the NST2000 software interface, particularly the JNI adapter.

The protected Java byte code is stored on a hard disk in protected (encrypted) form. When the application wants to create an instance of the class, stored in the protected file, the secure Java class loader starts a deciphering process. The deciphering process uses cryptographic functions from the NST2000 board and keys from the smart card. As a result, the critical Java code can be executed properly only if the NST2000 board is installed on a specific PC and if the smart card with a proper cryptographic key is inserted into the smart card reader (which communicates directly through the NST2000 board). Figure 5 illustrates this functionality.

This concept can be used for integrity check of the critical applications and the system software components. An operating system can, for example,

check its critical components in a startup time. The only location where the application is in the open form is a PC RAM.

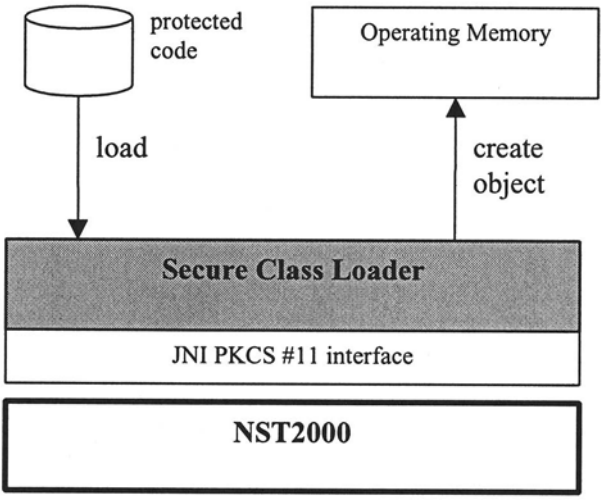

Figure 4. Conceptual model of secure Java class loader application. Secure class loader loads protected code from hard disk as a byte array. Then it decrypts array and checks its integrity, and creates new decrypted array. With newly created byte array secure Java class loader creates an object in operating memory. For encryption and integrity checking, loader uses NST2000 cryptography functions via JNI adapter.

6. CONCLUSION

In the paper, the SW/HW PC security module, called NST2000, is described and conceptually evaluated. The description of the realized sample application based on the first prototype of the NST2000 solution is given. As some other PCI-based or PCMCIA-based hardware security solutions, NST2000 implements a basic set of the standard cryptographic functions according to the PKCS#11 standard. Advantages of the NST2000 board regarding the other security solutions is in the following:

- The main advantage is the hardware based implementation of integrity protection functions for all the critical applications and the system software components. The NST2000 system represents the trusted resource. We assume that the trust that exist on the NST2000 functions can be delegated to potentially unsecure applications or operating system components. This is in compliance with the bedrock concept [2].
- NST2000 provides full flexibility for implementing the proprietary cryptographic solutions, or to the use of the predefined crypto

functions set. This is especialy important for the specialized applications for payment functions and e-government applications.

- NST2000 system supports additional functions, such as: keyboard input control, smart card control, hardware realized random number generator and real-time clock, which are very important for implementing the trusted system concept. The user data input control (keyboard and smart card data) is realized securely within I/O microcontroler that can not be tampered with viruses or Trojan horse programs as is the case for the software system drivers.

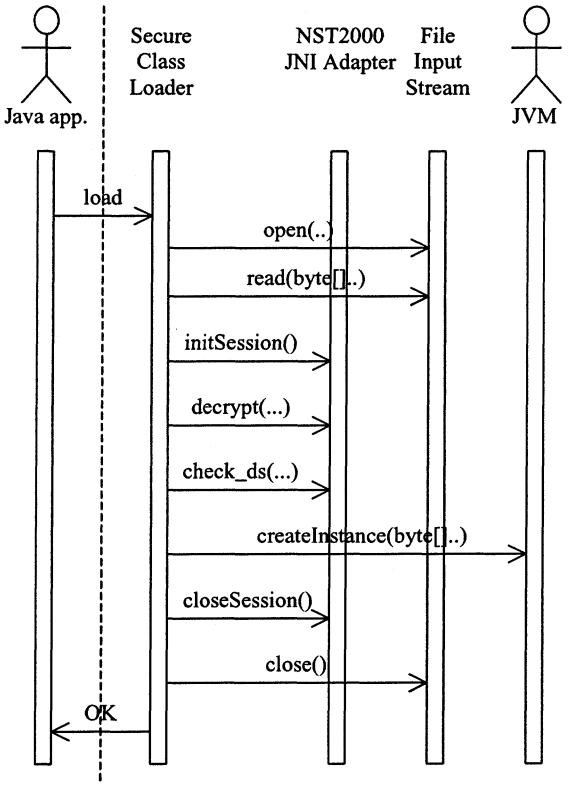

Figure 5. Simplified interaction model for the Java class loader application.

Based on the presented material, the NST2000 system is proposed for design of the highly secured computer networks systems. This solution is the main security component of the Secure Payment Gateway (SPG) computers which are to be used in the novel Yugoslav national payment network, called PLATNET. In the current experimental phase of the PLATNET network development, SPGs are based on the combinations of the smart cards and the software realized cryptographic functions. In the next step, in the final development phase of the PLATNET network, the

software realized cryptographic functions will be replaced with the suitable version of the NST2000 cryptographic coprocessor solution.

REFERENCES

[1] R.Oppliger, *Internet and Intranet Security*, Artech House, 1998, ISBN 0-89006-829.

[2] M.D.Abrams, M.V.Joyce, "Trusted system concepts," Computers and Security, VOL. 14, No. 1, Elsevier Science Ltd., 1995.

[3] Gemplus, *Gemplus GPK4000 Training Documentation,* 1998, http://www.gemplus.com.

[4] RSA Laboratories, *PKCS standards.*

[5] RSA Laboratories, *PKCS#11: Cryptographic Token Interface Standard*, Version 2.10, December 1999.

[6] Texas Instruments, *TMS320C2xx Fixed-Point Digital Signal Processors*, Product Bulletin, SPRT122C, 1997.

[7] Texas Instruments, *TMS320C54x Digital Signal Processors*, Product Bulletin, SPRT121F,1999.

[8] Texas Instruments, *TMS320C62x Digital Signal Processors*, Product Bulletin, SPRT136C, 1999.

[9] Knuth, D.E., *The Art of Computer Programming, Volume II, Seminumerical algrithms,* Addison-Wesley, 1997.