International Journal of Advanced Science and Technology
Vol. 12, November, 2009

CryptoNET: Integrated Secure Workstation

Abdul Ghafoor Abbasi and Sead Muftic

Department of Computer and System Sciences,
The Royal Institute of Technology (KTH)
Kista, Sweden
aghafoor@dsv.su.se, sead.muftic@dsv.su.se

Abstract

In most of the current applications security is usually provided individually. This means that various applications use their own security mechanisms and services, applied only to their own resources and functions. Furthermore, procedures to configure security parameters are usually inconvenient and complicated for non-technical users. As an alternative to this approach, we have designed and implemented Secure Workstation, which represents an integrated security environment and protects local IT resources, messages and operations across multiple applications. It comprises five components, i.e. four most commonly used PC applications: Secure Station Manager (equivalent to Windows Explorer), Secure E-Mail Client, Secure Documents System, and Secure Browser. These four components for their security extensions use functions and credentials of the fifth component, Generic Security Provider [5]. With this approach, we provide standard security services (authentication, confidentiality, and integrity and access control) and also additional, extended security services, such as transparent handling of certificates, use of smart cards, strong authentication protocol, SAML based single-singe-on, secure sessions, and other security functions, to all PC applications with the same set of security modules and parameters.

Keywords: *Encryption, strong authentication, Single Sign On, SAML Ticket, secure data processing environments.*

1. Introduction

Most of secure applications today were usually developed first with their basic functionality and security was added later, as an add–on extension or as additional, optional feature. If some already developed and operational application is to be enhanced with security, then the usual approach today is to invoke application programming interfaces (APIs) of some crypto library [1] [2] or some, so called, crypto services provider [3][4]. However, security tools and libraries today are not broadly available, sometimes not fully functional, and usually very complicated to use. Furthermore, security functions are usually applied only to resources and functions of the specific application. In addition, if an application offers some security services, then end-user has to configure various options and parameters prior to use of these security services. The procedures for that are usually inconvenient, especially for non-technical users.

Various existing solutions and commercial products were analyzed (outlined in Section 2) and found that most of them protect only information stored in files and messages in transit. However, we designed, implemented, and tested, an Integrated Secure Workstation (ISW), which also strongly protects its resources and operations against downloaded mobile code,

malicious software, intruders, insiders' attacks, and incorrect operations. Protection in our environment is based on a simple principle that all software modules, IT resources, messages, and operations are maintained and manipulated in the encrypted form. This is the essence of our approach and the core of our solution.

Our ISW is secure environment comprising four most popular PC applications and their associated security protocols. These applications are: Secure Station Manager (equivalent to Windows Explorer), Secure E-Mail Client, Secure Documents System (security extensions of Open Office), and Secure Browser (security extensions of standard browsers). Security protocols are: Strong Authentication, SAML–based Single-Sign-On, Secure Sessions, and some application–specific security protocols. All our applications and security protocols use functions and credentials of the single Generic Security Provider (GSP) [5], which also transparently uses smart cards, if they are configured and attached. The ISW may also be connected to various servers of our global security infrastructure, so it supports standard network security protocols, such as certification protocol [6], single-sign-on protocol [7], strong authentication protocol [8], and secure asynchronous sessions.

2. Security Analysis of Popular PC Applications

In this section we analyse security features and principles of some the most popular and widely used PC applications. With respect to security, we classify various PC applications in three groups:

- Applications that provide protection of PC resources against intruders, malicious code, theft, destruction, etc. Popular such applications are McAfee [9], Norton [10] or Symantec [11];
- Proprietary products, open source or commercial, that provides mainly encryption and/or access control to local resources. Examples of such products are eCryptf [12], Ubuntu File Browser [13], AxCrypt [14] or Crypt Manager [27];
- Standard PC applications, available on every desktop, with some security extensions: Web browsers (with SSL), E-mail clients (with S/MIME), and applications handling files and documents (with possibilities for encryption or creation of digital signatures). Examples are security extensions of E-mail clients to send/receive signed/encrypted e-mails, SSL for browsers, or digital signing of PDF documents in Adobe Acrobat.

Protection of Files: File or directory encryption functions, if available in file browsers, use symmetric key cryptography. These applications store symmetric keys either in the same folder or file they protect or in a separate encrypted private directory [15]. Some commercial products, like McAfee and Symantec, provide Endpoint Encryption Suites, which automatically encrypt files and devices using AES-256 symmetric key algorithm. In addition, this type of products sometimes also provides local access control and key management functions for sharing information in distributed environments. Another example, eCryptfs [12], provides security features like encryption of files, key management and access policies. This product stores cryptographic metadata in the header of each file, so that encrypted files can be copied between hosts without keeping track of the cryptographic keys. In general, currently available commercial and open source products do not provide strong and comprehensive security using advanced security functions such as asymmetric key cryptography, support of certificates, cryptographic encapsulation technique (PKCS#7), or strong authentication protocol.

Security in E-mail Clients: Popular E-mail clients, like MS Outlook, Eudora or Thunderbird provide end-to-end security for E–mail letters using S/MIME. These applications do not provide enhanced security features, like protection of their address books, key-management, transparent handling of certificates, use of smart cards, strong authentication protocol, single sign on, and protection against spam. Thus, E–mail is usually used to transfer malicious content, spam, viruses, etc.

Security with Browsers: Browsers are another application with serious security weaknesses and privacy threats. Current browsers do not protect browsing history, cookies, passwords, and data filled in Web forms. Furthermore, some browsers automatically download ActiveX controls from web servers [16], which are major source of vulnerabilities, viruses and worms. Eavesdropping, man-in-the-middle, spyware, malicious scripts are additional threats in most of the current browsers. Moreover, the integration of smart cards and strong authentication are not properly addressed.

Protection of Documents: Currently, most widely used document processing applications are MS Office [17] and OpenOffice [15, 18]. Both provide the possibility to encrypt documents using symmetric key. The key is stored internally in the protected file. This represents security vulnerability, since an attacker can discover the key by applying dictionary or brute force attacks. Document Security Systems [19], a commercial product, provides security functions like: illegal scanning, copying, digital imaging, protection of personal identification, authentication and authorization. Jakarta Slide [20] and JLibrary [21] provide security functions and services like security locks, constraints on documents, authentication and authorization. Protection of documents using advanced cryptographic techniques was explained in [22]. That research addressed security issues of documents stored at a local station and shared in group environments. In addition, the solution structures documents in sections accessible only by authorized group members. The enforcement of authorization policies and protection of sections are achieved by using Role–Based Access Control and symmetric key cryptography. The system was implemented as an extension of OpenOffice using XACML [23], Policy Decision Point, and Policy Enforcement Point. However, most of the current document manipulation applications provide weak security and do not support certificates, strong authentication, single sing on, secure session, or transparent integration with smart cards.

As conclusions, it may be emphasized that all examples of current security features are: (a) limited in scope (b) available only locally in individual applications (c) applicable only to resources of specific applications, (d) not extendable or replaceable with stronger solutions and finally (e) complicated to set-up and use. Contrary to those existing solutions, we have designed and implemented secure user workstation that not only overcomes all the listed shortcomings of existing applications, but at the same time addresses and also effectively provides solutions for problems treated by the first two groups of security applications. Our solution eliminates the possibility of intrusions, prevents stealing of valuable data or files, effectively protects against viruses, and transparently encrypts local resources.

3. The Concept of Integrated Secure Workstation

Integrated Secure Workstation comprises four secure end–user applications: Secure Station Manager, Secure E-Mail Client, Secure Documents System, and Secure Browser. It protects local IT resources, messages and operations across multiple applications using enhanced security functions. It implements security functions using Generic Security Provider (GSP),

which provides standard and extended security functions and features. Furthermore, GSP transparently connects to security hardware tokens, like smart cards, if available, which enables users to perform login, create signatures, and store security credentials using smart cards.

ISW also supports security protocols with components of our security infrastructure: Certification Authority Server, Identity Management Server (IDMS), Strong Authentication Server, SAML Policy Server, and Single Sign–On Server. Furthermore, in each domain, ISW supports network aspects of its applications by connecting to Secure E-Mail Server, Secure Web Server, and Secure Library Server. Security features of those application servers are not discussed in this paper in detail, but each application server is capable to process appropriate client's requests. To make it simple and understandable, we categorized security functions and features into two groups: common security functions and application–specific security functions. Common security functions are used across multiple applications, while application–specific security functions are used only by individual applications.

3.1. Common Security Functions

3.1.1. Local Authentication: At start-up, the system activates generic login module, shown in Figure 1. This module loads Generic Security Provider required for applications security. During this process Secure Workstation creates connections with IDMS Server and Local Certification Authority Server, if these servers are accessible. The module is generic in a sense that it uses user name and password, if smart card is not available; otherwise it requires PIN to authenticate to the card. Local authentication based on smart cards is compliant to the FIPS 201 standard [24], i.e. it supports PIN–only or PIN plus fingerprint authentication. After successful authentication, Secure Workstation checks for presence of user certificates. If they are not present and connection to Local CA Server is established, the Station will automatically request/receive three user certificates. If CA Server is not available, three user certificates will be created as self–signed certificates. After completion of local authentication procedure, it displays a generic graphical interface which is easy to use and managed as shown in other figures.

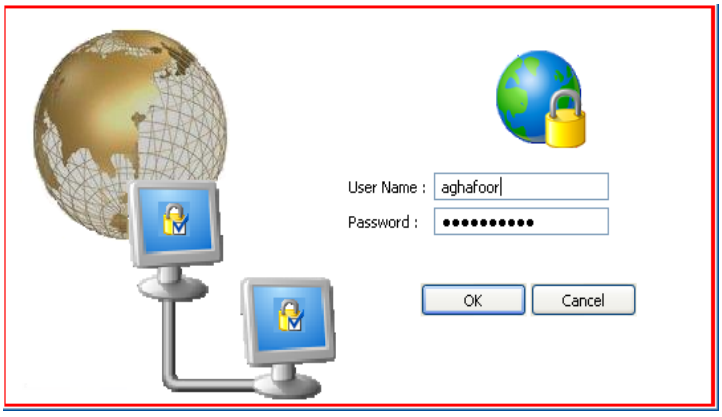

Figure 1. Generic Log-in module for local authentication

3.1.2. Handling of Certificates: Each user in a domain is registered by the IDMS, the responsibility of a Security Manager. He/she inserts user's registration and bio-data in the IDMS. One of important functions of Integrated Secure Workstation is transparent handling of certificates. The ISW fetches current user's registration data from the IDMS and creates Distinguished Name, which is used for generation of the three self–signed certificates. These

are: digital signature, key exchange, and non-repudiation certificates (the roles of certificates are explained in next sections). If Certification Authority (CA) Server exists in a domain, then the ISW requests and receives certificates from the CA Server. It stores certificates in a smart card, if it is connected; otherwise it stores them in a local certificate database. In addition, various certificate management functions are available with Secure Station Manager, as explained in the Section 3.2.1.

3.1.3. Strong Authentication and Single-Sign-On Protocol: Digital signature certificate is used to perform strong authentication with the Single Sign On (SSO) Server, which is connected to the XACML Policy Server. Extended strong authentication protocol follows the procedure defined in [8]. Extended security functions of strong authentication protocol are: verification of certificates by the Local CA Server and processing of SAML attribute assertion by the IDMS Server. After successful authentication, the SSO Server forwards the request to the XACML Policy Server, which issues SAML ticket and sends it back to the ISW. SAML ticket enables user to log-in to multiple application servers in a domain without repeating authentication process with each server.

3.1.4. Secure Asynchronous Sessions: Sometime ISW requires some application–specific services from application servers. For this purpose the ISW establishes Secure Asynchronous Session with application server(s). Session creation is based on the SAML authentication and key-exchange functions. Initially, ISW presents SAML ticket to Policy Enforcement Point (PEP) which is security gateway to each application server. The PEP checks the authenticity of the ticket from SSO by using SAMLAutheticationRequest and Response protocol. After successful authentication, the ISW exchanges key-exchange-certificate with application server, which is used to securely exchange session-id and session-symmetric-key. Furthermore, ISW stores session-symmetric-key in a smart card, if it is connected. Otherwise, it stores it in key-file, which is enveloped using key-exchange-certificate of the current user. The ISW uses session-symmetric-key and digital signature certificate to create secure messages in the standard format – PKCS#7 SignedAndEnvelopedData. The purpose of session-id is to facilitate ISW and PEPs at application servers to perform mutual asynchronous communication.

3.2. Application–Specific Security Functions

3.2.1. Secure Station Manager: Station Manager is an application, equivalent to Windows Explorer, but extended with security. Using Station Manager, users can perform standard file management functions like copy, cut, paste, rename file and folder, open file, etc. In addition, this application interacts with Local CA Server to generate, fetch, verify and list certificates, as shown in the background panel of Figure 2. Furthermore, it also creates encrypted AuditLog and decrypts it upon current user's request for inspections of its entries.

Station Manager generates local-resource-symmetric-key to encrypt and/or sign local files and IT resources, using standard cryptographic format – PKCS#7, as shown in the front data panel of Figure 2. Station Manager may also store local-resource-symmetric-key in a smart card, if it is connected to the computer. Otherwise, it stores it in key-file, protected by key-exchange-certificate of the current user.

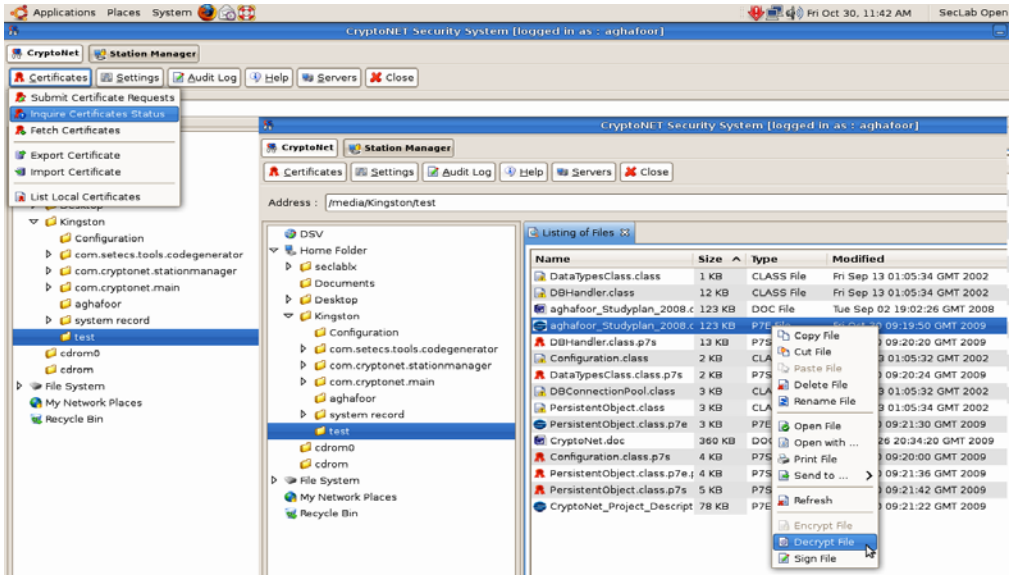

Figure 2. Certificate Management Functions of Secure Station and View of Protected Files and Actions in Data Panel (Listing of Files). Running on Linux environment

Station Manager is also capable to send local files to other user(s), registered in domain, in a protected form. It fetches the list of registered users from the IDMS and certificates of that selected recipient(s) from a Local CA. Then, it cryptographically encapsulates selected file(s) in the PKCS#7 SignedAndEncryptedData format and uploads them to a Library Server. Prior to uploading, Station Manager performs common security functions, explained in section 3.1. The recipient, using also ISW, downloads file(s) from the Library Server and opens them, after verification and decryption.

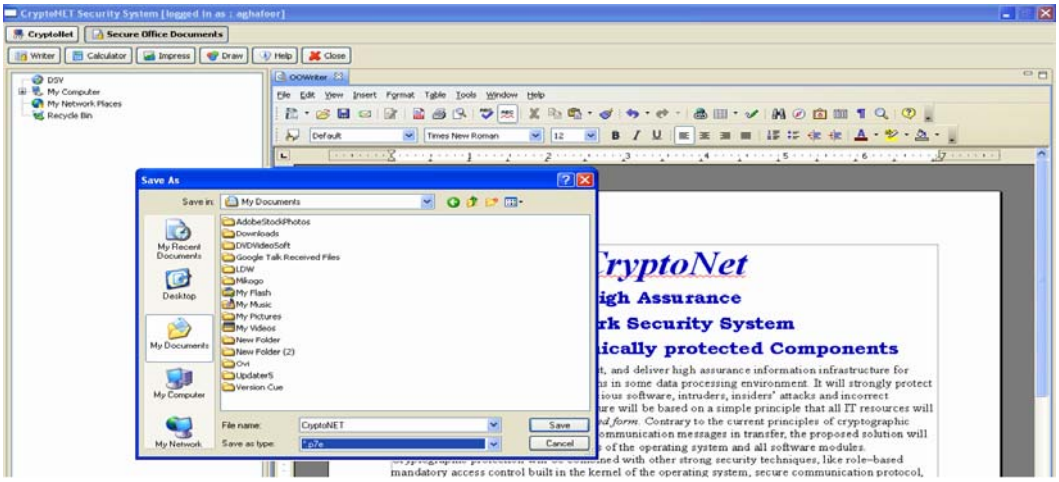

Figure 3. Secure Documents System based on OpenOffice with Security Extensions. It saves documents in encrypted format with *.p7e extension

3.2.2. Secure E-Mail Client: Secure E-Mail Client, which is a component of the Integrated Secure Workstation, performs standard e-mail functions: sending and receiving secure e-mail using S/MIME standard. It uses standard mail transport protocols: SMTP and POP3. Secure E-Mail Client stores contacts into the address book. For the protection of contacts, Secure E-Mail Client creates an address-book-symmetric-key and stores it in a smart card, if it is connected with system. Otherwise, it stores address-book-symmetric-key into key-file protected by the certificate of the current user. Secure E-Mail Client encrypts address book entries before storing them into address book and decrypts them before displaying on the display panel. Furthermore, this application is connected with Secure E-Mail Server to upload and download protected address books and address-book-symmetric-key for recovery and portability purposes.

Secure E-Mail Client uses only Signed and/or Enveloped e-mail letters. This approach reduces threats of viruses, spam and malicious code. In addition, in order to be authorized to receive those mails, each Secure E-Mail Server applies authorization polices, specifying other, authorized “Sending To” and “Receiving From” Secure E-Mail Servers. The complete functionality of Secure E-Mail Server is explained in [25]. In addition, Secure E-Mail Client creates secure session with Secure E-Mail Server by using Single Sing–On Protocol, explained in Section 3.1.3.

3.2.3. Secure Documents System: Secure Documents System offers standard functions to end-users like manipulation of word documents and spreadsheets, image editing and presentations. This functionality is based on the OpenOffice, which was extended with security features, as shown in Figure 3.

Secure Documents System transparently stores protected files using PKCS#7 SignedAndEnvelopedData format. This module also facilitates upload and download of documents to and from the Library Server, which is actually a repository of documents in a distributed environment. Furthermore, it provides options to securely distribute documents within the group of users.

Secure Documents System uses security features explained in the section 3.1 for distribution of documents. The format of distributed documents is also PKCS#7 SignedAndEnvelopedData. In addition, this application also manages documents in grouped environments and divides document into different sections, accessible only to authorized users. Enforcement of authorization polices and key management is performed by the Library Server.

3.2.4. Secure Browser: Secure Browser, as the component of the Integrated Secure Workstation, performs standard functions to exchange information using HTTP protocol with standard Web servers through our Secure Web Proxy Server. Secure Browser creates secure session with the Secure Web Proxy Server which further creates connection with standard Web Server. Secure Browser accepts data only in the PKCS#7 SignedAndEnvelopedData format which protects the PC from viruses and malicious code. Furthermore, Secure Browser generates a browser-symmetric-key to encrypt history, cookies, passwords, and automatic form filling data. Secure Browser may store browser-symmetric-key in a smart card, if it is connected to the computer. Otherwise, it stores it in a key-file, protected by key-exchange-certificate of the current user. After receiving them, Secure Browser decrypts Web pages by using browser-symmetric-key.

4. Implementation

Our implementation of the described ISW is based on the concept of generic security objects [26], implemented in Java in the form of Eclipse plug-ins. Eclipse plug-in architecture supports dynamic integration and generalization of its components. Since it is implemented in Java, the ISW is multiplatform software and can be installed in Windows and Linux environment. The plug-ins of our framework are all packaged for run–time execution in the encrypted format, therefore not vulnerable to any software attacks. These encrypted modules are loaded by our specially created Secure Class Loader, which decrypts and verifies each module before loading them in main memory for execution.

5. Conclusions

Our Integrated Secure Workstation provides comprehensive set of security services for PC environments and selected applications. The main principle was to cryptographically protect local IT resources, potations and messages. It transparently handles security functions and services. The design of ISW is based on the concept of generic security objects, which can be used by any application included in the Secure Workstation. Our future research topics are security issues of application servers which will enable CryptoNET to provide a complete secure framework for network applications.

References

[1] OpenSSL, http://www.openssl.org/docs/ visited on January, 2009

[2] RSA Security, Inc. “BSAFE: A Cryptographic Toolkit”, Library Reference Manual Version 4.0 http://www.rsa.com/products/bsafe/documentation/ cryptoc_411_ reference.pdf

[3] SUN Corporation, Java Cryptographic Extensions (JCE), www.sun.com visited on February, 2009

[4] Microsoft Corporation, Cryptographic Services Provider (CSP), www.microsoft.com visited on Feburary, 2009

[5] G. Abbasi, S. Muftic, G. Schmölzer, “A Model and Design of a Security Provider for Java Applications”, accepted in The 4th International Conference for Internet Technology and Secured Transact (ICITST-2009), London, UK, November 2009,

[6] Adams, S. Farrell, T. Kause, T. Mononen,, RFC 4210, Internet X.509 Public Key Infrastructure Certificate Management Protocol, September 2005

[7] Liberty Alliance Project, Case Study “SSO for All” http://www.projectliberty.org /liberty/content/download/4064/27328/file/ssocircle_libertycasestudy2.08.pdf downloaded on March, 2009

[8] A. G. Abbasi, S. Muftic, G. Schmölzer, “CryptoNET: Secure Federation Protocol and Authorization Policies for SMI”, Accepted in International Conference on Risks and Security of Internet and Systems 2009 (CRiSIS 2009) and will be held on October, 2009, Toulouse, France

[9] McAfee SECURE, “Data Sheet”, downloaded form http://www.mcafee.com /us/local_content/datasheets/ds_endpoint_encryption.pdf downloaded on September, 2009

[10] Norton 360, All in one Security http://www.symantec.com/norton/360

[11] Symantec, White Paper: Enterprise Security, “Critical System Protection and Endpoint Encryption for the PCI Data Security Standard”, downloaded form http://www.symantec.com/business/products/whitepapers.jsp?pcid=pcat_security&pvid=endpt_encryption_1# on September 2009.

[12] eCryptfs, “eCryptfs – Enterprise Cryptographic Filesystem”, https://launchpad. net/ecryptfs visited on May, 2009.

[13] Ubuntu File Browser, http://www.ubuntu.com/ visited on July, 2009

[14] AxCrypt, “Introduction and Features”, Axantum Software AB, Sweden, http://www.axantum.com /AxCrypt/Features.html visited on October, 2009

[15] E. Filiol, J. Fizaine, White paper, "Open Office v3.x Security Design Weakness", Laboratorie de virology et de cryptologie operationnelles, France, March, 2003

[16] Information Security Awareness, "Browser Threats" http://infosecawareness.in/isea/women/browser-threats Last modified: 2009-05-08

[17] Microsoft Corporation, "Using Microsoft Office 2003 security features", http://www. microsoft.com/protect /products/yourself/office2003.mspx visited on May, 3009.

[18] Sami Rautiainen "OPENOFFICE SECURITY" published in the processing of the 13th Annual Virus Bulletin International Conference (VB2003), Toronto, Canada 25-26 September 2003.

[19] Document Security Systems, Inc. http://www.docum entsecurity.com/ visited on July, 2009

[20] Jakarta Slide, http://jakarta.apache.org/slide/archit ecture.html visited on July, 2009.

[21] Open Source Project, JLibrary, "Tutorial: Security management" http://jlibrary.sourceforge.net/1/tut6.h tml, downloaded on July, 2009.

[22] M. Alhammouri, S. Muftic, "A Model for Creating Multi-level-security Documents and Access Control Policies", published in the proceeding of SSI´2006 - 8th Intl Symposium on System and Information Security, Sao Jose Dos Campos, Sao Paulo, Brazil, November, 2006

[23] M. Alhammouri, S. Muftic, "A Design of an Access Control Model for Multilevel-Security Documents" published in the proceeding of The 10th International Conference of Advanced Communication Technology (ICACT 2008), pp 1476-1481, Feb 2008, Phoenix Park, Korea.

[24] FIPS 201 "Personal Identity Verification (PIV) of Federal Employees and Contractors", Computer Security Division Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8900, March 2006.

[25] A. G. Abbasi, S. Muftic, G. Schmölzer, "CryptoNET: Design and Implementation of the Secure Email System" published in the proceeding of IEEE International Workshop on Security and Communication Networks, Trondheim, Norway, May 2009

[26] Ciobanu Morogan Matei, Licentiate thesis: "Generic security objects", Computer and Systems Sciences year, KTH, published by Data- och systemvetenskap in year 2000

[27] Crypt Manager, User Guide, http://www.ubuntugeek. com/crypt-manager-an-encrypted-folder-manager-for-ubuntu-linux.html visited in October, 2009

Authors

Abdul Ghafoor Abbasi received the Master of Science (MS) degree from National University of Sciences and Technology, Islamabad in 2004. Currently, he is pursuing his PhD in the filed of Network Security at The Royal Institute of Technology, Stockholm, Sweden. He started his professional carrier as Lecturer in 2002 and served at public sector colleges and universities. Prior to start his higher studies, he worked at NUST School of Electrical Engineering and Computer Sciences, Islamabad as a lecture. He is member of Security Lab at DSV Department, Kista, Sweden, Information Security and Distributed Computing Group, SEECS, Islamabad, IEEE Graduate Student, Stockholm research association, and IPID Sweden.

Sead Muftic has been working in the area of computer security for more than 30 years. He is professor of Computer Security at the Department of Computer and Systems Sciences (DSV), The Royal Institute of Technology, Stockholm, Sweden and also research professor at The Michigan Technical University USA). Dr. Muftic was the member of the Permanent Stakeholders Group (PSG), an expert advisory group to ENISA (European Networks and Information Security Agency), director of the EU COST–11 Security project, consultant to VISA, World Bank, Siemens and other international organizations. Dr. Muftic is the author of three international books and about 100 research and scientific papers published in journals or presented at international conferences.