ASTES Advances in Science, Technology and Engineering Systems Journal Vol. 1, No. 5, 21-26 (2016)

Special issue on Recent Advances in Electrical and Electronics Engineering

www.astesj.com

ASTESJ
ISSN: 2415-6698

Strong Authentication Protocol based on Java Crypto Chip as a Secure Element

Majid Mumtaz[*,1], Sead Muftic[2], Nazri bin Abdullah[3]

[1]Lecturer, COMSATS Institute of Information Technology, Quaid Avenue, Wah Cantt, 46000, Pakistan

[2]CEO, SETECS Inc. Rockville, MD 20852, USA

[3]PhD Student, ICT/Dept. of Communication Systems, Royal Institute of Technology (KTH) Stockholm, Sweden

ARTICLE INFO

Article history:
Received : 10 September, 2016
Accepted: 11 October 2016
Online: 27 October, 2016

Keywords:
Authentication
Mobile security
Strong authentication
X.509 certificates
challenge/response
microSD Secure Element
FIPS 196
PKI

ABSTRACT

Smart electronic devices and gadgets and their applications are becoming more and more popular. Most of those devices and their applications handle personal, financial, medical and other sensitive data that require security and privacy protection. In this paper we describe one aspect of such protection – user authentication protocol based on the use of X.509 certificates. The system uses Public Key Infrastructure (PKI), challenge/response protocol, mobile proxy servers, and Java cards with crypto capabilities used as a Secure Element. Innovative design of the protocol, its implementation, and evaluation results are described. In addition to end-user authentication, the described solution also supports the use of X.509 certificates for additional security services – confidentiality, integrity, and non-repudiation of transactions and data in an open network environment. The system uses Application Programming Interfaces (APIs) to access Java cards functions and credentials that can be used as add-ons to enhance any mobile application with security features and services.

1. Introduction: User Authentication

Authentication of users is one of the most important security services for any application in the Internet environment. It guarantees that applications and their resources are used only by legitimate and authorized users. In addition, it represents important prerequisite for many other security services, such as data confidentiality (for exchange of cryptographic keys), data integrity (protection of digital digests), access control, non–repudiation, and so on. Authentication as the security service is also used to verify identities of other components of an IT environment, such as applications, servers, user workstations, messages, documents, E-mail letters, and other digital objects.

Because of its importance in any IT environment, it is essential that authentication is always performed correctly and with high degree of trust in its protocol and the outcome.

The essential goal of an authentication protocol is to verify the identities of parties and components participating in some application or transaction. This goal is usually accomplished by validating some secret value associated with the claimed identity. Alternative protocols include verification of some unique and intrinsic properties of individuals that provide their identities and participate in their validation.

This paper is focused on the authentication protocol specified in the National Institute of Standards and Technology (NIST) FIPS 196 standard [1]. The essence of that protocol is challenge/response procedure based on randomly generated number for each execution of the protocol, so that specific instance of the protocol and its results are non–repeatable. This approach prevents man-in-the-middle attack based on replay of protocol messages. Cryptographic protection of messages is based on public key cryptography, where both participants in the protocol – Identity Claimant and Identity Verifier, have a pair of asymmetric crypto keys and corresponding certificates. Using these credentials all messages of the protocol are cryptographically protected – digitally signed and enveloped, what guarantees successful verification of all messages and therefore successful completion of the protocol.

FIPS 196 standard does not specify specific details of the cryptographic protection of protocol messages. But, this aspect is complemented by another NIST standard – SP 800-63-1 [2]. This standard defines four levels of assurance, Levels 1 to 4, in terms of

[*]Corresponding Author Majid Mumtaz, CIIT Wah Campus Wah Cantt Pakistan, +92(0) 51-9314384 & mmumtaz@ciitwah.edu.pk

https://dx.doi.org/10.25046/aj010505

the strength of cryptographic algorithms, their parameters, authentication procedures, consequences of authentication errors, and misuse of credentials. Level 1 is the lowest assurance level, and Level 4 is the highest.

Brief description of the four assurance levels and their main characteristics is the following:

1.1 Assurance Level 1 – Low Assurance

At this level identity proofing is not required – simple user *login name* may be used for that purpose. Authentication mechanism is usually user *login password*, which is used as a shared secret with the Identity Verifier. Such authentication mechanism provides some assurance that the same Claimant who participated in previous transactions is accessing again the protected transaction or data. At Level 1, long-term shared authentication secrets (user password) are revealed to and shared with Verifiers.

The protocol relies on encryption of passwords for their transfer through a secure communication channel, usually SSL. This assurance level does not require cryptographic methods that block offline attacks by eavesdroppers. Authentication protocols that are implemented based on principles suggested for this level have several problems. The most important are: sharing of secret authentication credentials what gives the possibility of dishonest server administrators to impersonate users. Credentials stored at the servers are vulnerable and usually multiple credentials are used at different servers. The protocol that solves all these problems, described in section III.

1.2 Assurance Level 2 – Single Factor Remote Authentication

This level provides moderate assurance for authentication protocols. At this level identity proofing requirements are introduced, requiring presentation of identifying materials or information. For single factor authentication, Memorized Secret Tokens, Pre-Registered Knowledge Tokens, Look-up Secret Tokens, Out of Band Tokens, and Single Factor One-Time Password Devices are suggested. Successful authentication requires that the Claimant proves using a secure authentication protocol that he/she controls the token. In addition to Level 1 requirements, authentication assertions must be resistant to disclosure, redirection, capture, and substitution attacks. This implies that their cryptographic protection is needed. Certified and approved cryptographic techniques are required for protection of all assertions used at Level 2 and above.

Protocols implemented at this assurance level have reasonable good security, except that they are based on a single authentication factor. So, their assurance is not too high and these protocols are not suitable for highly sensitive applications and data.

1.3 Assurance Level 3 – Multi-Factor Remote Authentication

This level provides medium assurance in authentication protocol since at least two authentication factors are required. At this level identity proofing procedures require verification of identifiers. Authentication is based on proof of possession of the allowed types of tokens through a cryptographic protocol using strong cryptographic mechanisms that protect primary authentication tokens against compromise by all threats at Level 2 as well as Verifier impersonation attacks.

Authentication requires that the Claimant proves, using a secure authentication protocol, that he or she controls the token. The Claimant "unlocks" the token (the first factor) with a password or biometric (the second factor). Long-term shared authentication secrets are never revealed to any party except the Claimant and Verifiers.

Although authentication protocols at this assurance level are stronger than at Level 2, they still have weaknesses of shared secrets with Verifiers as well as multiplicity of such secrets with multiple Verifiers.

1.4 Assurance Level 4 – Multi Factor Remote Authentication

This level provides the highest degree of assurance in authentication protocols. At this assurance level in-person identity proofing is required what implies that identification data must be established by some trusted Registration Authority. The core requirement at this level is that only hardware cryptographic tokens must be used. The token is required to be a hardware cryptographic module validated at Federal Information Processing Standard (FIPS) 140-2 Level 2 or higher with at least FIPS 140-2 Level 3 physical security [3]. Level 4 token requirements can be met by using the PIV authentication certificate of a FIPS 201 compliant Personal Identity Verification (PIV) smart card [4].

The key characteristics and distinguished features of the strong authentication protocol described in this paper is that it provides the highest level of assurance at Level 4. In addition, another important feature of the solution is that, by suitable extensions of the FIPS 201 standard, the same cryptographic token (PIV card) can support other types of protocols at three other assurance levels. These features are available not only using Java smart cards with PC/Windows workstations, but also using Java crypto chips combined with smart phones. Therefore, the protocol is at Assurance Level 4 and it is available for PCs, for smart phones, and for other mobile devices and gadgets. In order to even prevent brute–force analysis using powerful computers by legal agencies, but without proper authorization, all data are randomized before encryption using ExOR with random 256 bit masks. This transformation makes analysis of encrypted data exponentially more difficult compared with data encrypted using standard crypto algorithms.

The remaining sections of the paper are organized as follows: in Section 2 related work and relevant alternative solutions are described and analyzed. In Section 3 all details of our protocol, its components and steps are described. Section 4 describes the management of security credentials as used in the protocol. Section 5 describes current implementation. Whereas Section 6 contains the results of evaluation and validation based on requirements of the Assurance Level 4. The last Section contains the conclusion and suggestions for further research and potential improvements.

2. Related Work and Standards

There are several research papers and standards dealing with strong authentication protocol using Java chips and mobile PKI. Although they address interesting problems, none of them describe a solution that is as comprehensive and also formally validated, as the protocol described in this paper.

Wireless Application Protocol (WAP) Forum was the first to specify Wireless PKI (WPKI) protocol for wireless environments [12]. In the WPKI protocol Web portal acts as a Gateway Server. It receives WAP client requests and transfers them to the Registration Authority (RA) and Certification Authority (CA) servers. The WAP client uses direct URL instead of X.509 certificate exchange. The entity that wants to communicate securely with a WAP client needs first to download the certificate from given URL and then verify client's signature on the authentication token.

The paper [13] describes an approach based on the use of mobile phones and SIM (Subscriber Identity Module) chips. The proposed solution utilizes security features, user identities, and public/private key pairs available inside SIM Module. The solution is dependent on a telecom issuing and personalizing SIM chips during client's subscription, so security of user's data and transactions depends on third party. Therefore, this system requires user's trust on services and actions of the third parties and also lacks protection of consumers privacy.

Research results reported in [14] proposed the use of enhanced PKI credentials as security tokens for mobile phones. The system comprises several components: a PKINIT component (i.e. an enhanced version of Kerberos); a client component; PANDA component (i.e. a device powered by Zigbee protocol) for communication and sensing locations; Delegation Server component is used to manage certificates and private keys for signing certificates; Referee Server component represents protocol bridge between a server and a client. Mutual authentication is performed by the Delegation Server and it is based on PKI. Upon receiving proxy certificate and Delegation Server's public key, a client signs it by his/her own private key and sends it back to the Delegation Server. In response, the Server returns a challenge to the client. The client encrypts the challenge and returns it back to the Delegation Server. Delegation Server performs its verification with the assistance of the Referee Server. Upon success, PKINIT is activated to issue Service-Granting Ticket (SGT) to the client. Authors claimed that the protocol provides authentication, digital signatures, non-repudiation, and secure distribution of keys to the client. The solution is comprehensive but quite complex, it has quite high deployment cost as it has a number of resources required to support different services at different levels. Our system provides the same security services, but with simpler structure and in transparent fashion to end user's.

Research results in [15] suggest the use of certificates for mobile phones. The authors claim that their authentication protocol is not only based on PKI certificates, but that it also provides secure solution to mobile applications. In addition, they claim user confidence that their credentials are password protected and kept secret. They measure the strength and protocol latency of their solution using security threats. Authors compare their solution with well-known authentication solutions by using formal verification approach and claim that their solution is more efficient and has the lowest latency for mobile phones. But, such claim requires practical testing in an enterprise environment and also requires tamper-proof technology.

Trichina [16] proposed a PKI system for SIM-based mobile payments in Finland. This proprietary solution was deployed with the help of telecommunication operator for secure mobile payments. Mostly financial organizations located in Finland can utilize the system according to operator guidelines. Network operator is responsible for issuance of PKI-SIM cards to customers. FINEID SAT applet module inside the SIM card generates digital signature and corresponding public-key certificate for customers. The big challenge to such system is privacy and customer confidence, as it is based on trust in third party services. A number of challenges are highlighted by [17] for such solutions especially when using online m-commerce applications.

Another PKI solution for mobile environments was proposed by Jeun and Kilsoo [18]. They first generate public/private key pair on a personal computer (PC) and manually transfer it into a mobile phone. Customers initiate PKI services by using SMS message requests to the server. In their system mobile devices rely completely on PC security, as PC generates public/private key pair and certificates on behalf of a customer. If PC is compromised, the complete customer's security is compromised. The solution has a number of challenges including insecure storage of public/private keys and their manual transfer to mobile phones.

Lee [19] proposed a WPKI based solution. In the solution an Elliptic Curve Digital Signature Algorithm (ECDSA) is utilized for key pair generation. He claimed that the generated certificate has reduced size as compared to the standard X.509 certificate. For validation of certificates, he uses Online Certificate Status Protocol (OCSP) instead of Certificate Revocation Lists (CRL). Although the solution is based on optimized protocols for certificate management, it has a number of limitations for mobile applications. A serious issue for every PKI-based solution is the protection of a private key. The best solution for tamper-proof storage and protection of private keys is to use either smart card chips or smart micro SD card chips. Compared with Lee's solution, our protocol use a tamper-proof technologies. In addition, its completeness, availability on multiple platforms, and compliance to standards have been proven using official validation and certification standards and methodologies.

3. Protocol Components and Steps

Two core components of our system are Strong Authentication Client and Strong Authentication Server. There are two implementations of the Client. One as Java *Web Start (JWS)* module, which is dynamically downloaded to and activated in the PC/Windows environment upon activation of the protocol. The other is a mobile application with versions for IOS and Android smart phones, called *m–Security*. Both versions are protected against malware and illegal code modification: JWS module is digitally signed, while for mobile applications software modules are encrypted before loading into mobile devices. For execution of such encrypted software modules special Java Class Loader is implemented as an extension of the standard Java Class Loader. Security Loader dynamically decrypts Java classes in the process of loading them into main memory before execution.

Strong Authentication Server comprises two servers: Web server and a classical network Strong Authentication server. Web server, when accessed through PC browser, dispatches JWS Strong Authentication module to the PC where client side functions of the protocol are performed. Network Strong Authentication server listens the socket and performs server side functions of the protocol. This server interacts with both, JWS client and also with m–Security client, during execution of the protocol.

The steps of the protocol are fully compliant with requirements for validation of HSPD-12 (PIV) products in order to be included in the GSA HSPD–12 Approved Products List [5]. These requirements are specified in the document [6]. All cryptographic operations are performed by the PIV smart card. The steps are the following:

Step 1: User either clicks on an icon for Cloud Login module or starts browser and visits security–enhanced application server. In both cases, login panel is displayed (Figure 2).

Step 2: User inserts PIV card into the smart card reader and enters his/her PIN using keyboard and simple smart card reader or using more secure smart card reader with the PIN pad.

Step 3: If PIN is correct, smart card will be activated and PIV authentication certificate is read form the card.

Step 4: Certificate is sent to the Strong Authentication Server, representing the first, identification message, in accordance with the FIPS 196 standard.

Step 5: Strong Authentication Server verifies user by verifying that

- User is registered in the IDMS and his/her status is correct (not suspended or terminated)
- Certificate is verified against CRL and through verification of the certificate chain to the top of the PKI
- The status of the smart card is verified against the database of valid PIV cards.

Step 6: If all verifications complete successfully, Strong Authentication Server generates random number, envelopes it using user's public key (extracted from the user's certificate) and sends it back to the user as the challenge together with its own certificate, in accordance with the FIPS 196 standard.

Step 7: Challenge is passed into user's PIV card, where it is decrypted using user's private key stored in the card, then enveloped using server's public key (extracted from its certificate), thus creating user's response.

Step 8: Response is returned to the Strong Authentication Server which opens the envelope using its private key and verifies user's response against its original challenge

Step 9: If the verification is successful, Strong Authentication Server contacts Policy Decision Point (PDP) Server to issue SAML/SSO ticket to the user

Step 10: SAML/SSO ticket is issued for the user and returned to the Strong Authentication Server, which sends it to the user together with a random session key, both protected using public key cryptography

Step 11: User stores SAML/SSO ticket into PIV card.

The final results of the authentication procedure are that

- User is authenticated with certainty, as the person in possession of the PIV card issued to that user
- User has SAML/SSO ticket in his/her smart card,
- PDP Server has the copy of the user's ticket,
- Shared secret session key is established between Strong Authentication Server and user's workstation.

4. Management of Security Credentials

The following security credentials are used in the protocol:

(1) user registration data, stored in an IDMS server and used in the form of the Distinguished Name object; (2) user X.509 certificate; (3) SAML/SSO ticket; and (4) PIV smart card. This implies that, in addition to Strong Authentication server, several other servers are used to manage those credentials. In particular, based on the list of four credentials, four such servers are used: (1) Identity Management System (IDMS) server managing user's registration data and their identities; (2) Certificate Authority (CA) server managing X.509 certificates; (3) Policy Decision Point (PDP) server managing authorization policies and tokens; and (4) Card Management Server for issuing and managing PIV cards.

Various aspects of security management are based on an innovative concept of security proxies [7]. Those are "intermediate" servers, connecting users with various security and application servers. Based on such concept, Strong Authentication Server is designed and implemented as a proxy for other security servers. In that way, users can access and use various security services through a single "contact point". Besides flexibility for users, this approach has also advantages in terms of user security, privacy and anonymity. The details about servers, their data, services, protocols and security, are beyond the scope of this paper. Their use and services are described for completeness of this paper. The architecture of the system is shown in Figure 1:

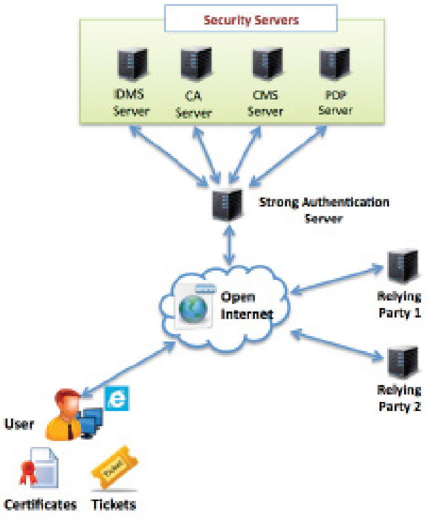
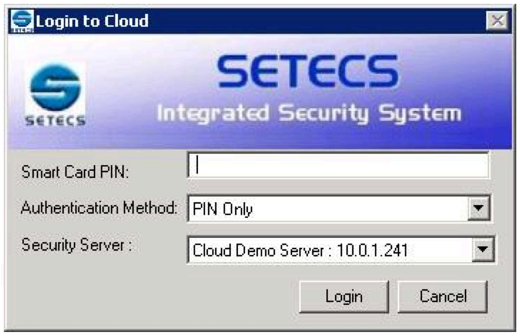


Figure 1: SA Server as Security Proxy

5. Current Implementation

The complete system is already implemented, tested, and certified. This section describes only its three main components: (1) PC/Windows based client; (2) mobile client (m–Security); and (3) Crypto Services Provider (CSP).


Login to Cloud

SETECS

Integrated Security System

Smart Card PIN:

Authentication Method: PIN Only

Security Server : Cloud Demo Server : 10.0.1.241

Figure 2: Login Panel of the JWS Client

5.1 PC/Windows JWS Client

As already described, this client is dynamically downloaded from the Web interface of the Strong Authentication Server into user's local PC/Windows workstation. Upon activation, it performs transparently all its functions. Users activate their PIV card by entering PIN in the Login panel (Figure 2). If smart card reader with PIN pad is used, then PIN is entered using the reader. Upon activation of the card, the steps of the protocol, described in section 3, are performed and the client simply displays success message.

5.2 Mobile Security Client

Strong Authentication Client for mobile platforms, besides strong authentication protocol, it also includes all functions necessary to manage security credentials that are needed in the strong authentication protocol. These functions are accessed using Graphical User Interface (GUI) menu shown in Figure 3. The functions are organizes in a logical sequence of steps.

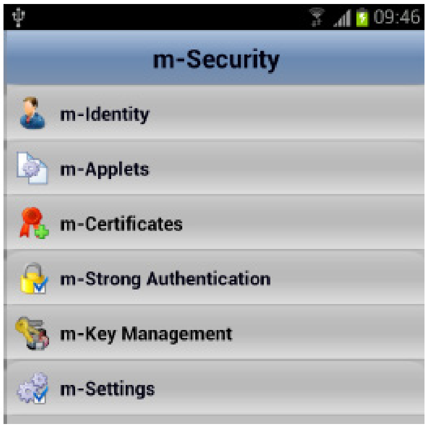


Figure 3: GUI of m–Security Application

Button *m–Identity* is used to register or update registration data. After that, the button *m–Applets* is used to download PIV and Security applets into JavaCard chip. During download, personal data are loaded into PIV applet, according to the PIV standard [8]. *m–Certificates* button generates in the chip two RSA keys, extracts public key, sends it in the form of PKCS#10 Certificate Request to the CA server, receives the PKCS#7 reply, and stores certificate in the PIV applet of the card. After that, the chip and the Client are ready to perform strong authentication, as already described. *m–Key Management* button is used to refresh session keys established during authentication procedure.

5.3 Crypto Services Provider

Both types of Clients, PC/Windows version and also mobile version, are using Crypto Services Provider (CSP) for all their cryptographic functions. CSP is the component of the security system that provides cryptographic services to both clients and also to all servers.

Several versions of the CSP have been designed and implemented. The details are described in [9]. That paper describes modules, APIs and validation procedure for the CSP, which is used by Strong Authentication clients described in this paper. Since all Strong Authentication Clients use crypto chips, some details of usage of the CSP, when Secure Element is a crypto chip, are here described.

With PCs standard Java cards are inserted into a smart card reader connected to the PC workstation. With mobile phones, there are two versions of embodiments of crypto chip. With one, the chip is embedded into microSD card, which is then inserted in the microSD slot in mobile phones that have such slot. For mobile phones that do not have microSD slot, external smart card reader is used, attached to the phone. Standard Java card is inserted into the mobile smart card reader. This solution is shown in Figure 4.

The card is inserted into mobile smart card reader that has PIN pad and LCD display to handle PIN and card data. The reader has audio interface, so it may be used with all types of smart phones. The card and the reader are inserted into a phone on the top, but to save space of the paper, they are shown next to each other.

Figure 4: Mobile Smart Card Reader and PIV Smart Card

Extensive research has been already performed related to managing and using Universal Integrated Circuit Chips (UICC) directly in smart phones, when such chips become broadly available in smart phones [10].

6. Evaluation and Validation

This section briefly describes the approach and results of the evaluation procedure and formal validation procedure that have been performed for the described system.

The protocol has been evaluated against NIST requirements for authentication protocols at Level 4 [2]. Besides its core requirements that the protocol uses hardware token and two factors authentication, the standard requires:

Level 4 requires strong cryptographic authentication of all parties and data transferred between parties.

The protocol uses strong cryptographic algorithms (AES and RSA) with long crypto keys (256 bits for AES and 1024 for RSA). Both algorithms are implemented in hardware. All messages within the protocol are encrypted and digitally enveloped, so they are all strongly protected.

The token secret shall be protected from compromise through the malicious code threat.

In the system there are two token secrets: user's PIV card PIN and user's RSA private key. Both are stored in the card and cannot be read, only used. RSA private key is even generated in the card and never leaves the card. PIN is protected by its blocking after three unsuccessful verification attempts.

Long-term shared authentication secrets, if used, shall never be revealed to any party except the Claimant and CSP; however, session (temporary) shared secrets may be provided to Verifiers or Relying Parties by the CSP.

The system does not use shared secrets. Session keys are exchanged cryptographically signed and enveloped using public key cryptography, so they are shared and can be used only with designated, legitimate Verifiers.

Strong, approved cryptographic techniques shall be used for all operations including the transfer of session data. All data shall be cryptographically authenticated.

The protocol uses AES (256 bits key) and RSA (1024 bits keys). Both algorithms are officially approved and validated [3]. Man-in-the-Middle (MitM) attacks are completely eliminated, as all messages are digitally enveloped by recipient's public key, so they can be opened only by the designated, legitimate recipients.

Level 4 assurance may be satisfied by client authenticated TLS (implemented in all modern browsers), with Claimants who have public key Hardware Cryptographic Tokens.

This requirement is out of scope of the protocol. It requires TLS based on client's certificate, so Web server of the Strong Authentication client must be configured to require client authentication in the TLS handshake process.

In addition to evaluation of the protocol for compliance with the NIST Assurance Level 4, the protocol has also been officially validated by the GSA, an agency of the US Federal Government. Validation was performed for the category "*PIV Authentication System*" of the GSA HSPD-12 Validation Program [11] and included in the official US Government HSPD–12 PIV Approved Products List [5].

7. Conclusion and Future directions

In this paper we have described our design and current implementation of the strong authentication solution for PC/Windows, mobile phones, smart gadgets, and other mobile devices. The prototype has been developed and evaluated according to industry compliance standards with lowest to highest authentication assurance levels. The designed solution provides transparent security, privacy and anonymity services to end user's.

As the next steps we are planning to integrate our system with different applications including vehicle tracking devices, health care appliances, and other embedded devices, especially Internet of Things [20]. In our future work we will integrate the solution with cloud-centric Internet of Things applications.

Another interesting area that we are already pursuing is use of the protocol for peer-to-peer authentication, without third parties. The innovative concept for validation of such transactions is blockchain. At the moment, there is a great need to provide strong authentication when accessing and using blockchain, but there are no even early solutions.

Finally, the third area of our research and development interest and our current activities is security of peer-to-peer transactions, also based on the use of the blockchain.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

My foremost appreciation and great thanks goes to Allah Almighty for his infinite wisdom and Grace. Special thanks and gratitude mainly goes to my advisor Dr. Sead Muftic for his great motivational counselling that impacted me technically and theoretically for completion of this task. It is by his support and encouragement that I was able to finish this good work.

References

[1] NIST, National Institute of Standards and Technology, "Entity Authentication using Public Key Cryptography", FIPS 196, http://www.nist.gov/manuscript-publication-search.cfm?pub_id=901429 [Retrieved March 2016]

[2] NIST, National Institute of Standards and Technology, "Electronic Authentication Guideline", NIST SP 800-63-1, http://csrc.nist.gov/publications/nistpubs/800-63-1/SP-800-63-1.pdf [Retrieved March 2016]

[3] NIST, National Institute of Standards and Technologies, "Validated 140-1 and FIPS 140-2 Cryptographic Modules", Item 1111, http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm [Retrieved March 2016]

[4] NIST, National Institute of Standards and Technologies, "FIPS 201 – Personal Identity Verification (PIV) of Federal Employees and Contractors", March 2006, http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.201-2.pdf [Retrieved June 2016]

[5] GSA, US Federal Government, "HSPD-12 Approved Products List", http://www.idmanagement.gov/approved-products-list [Retrieved June 2016]

[6] GSA/US Federal Government, "PIV Authentication System, Approval Procedure, v2.0.0", April 14, 2010, http://www.idmanagement.gov/ [Retrieved June 2016]

[7] N., bin Abdullah, "Internet Security and Privacy System based on the Concept of Trusted Proxies", Licentiate Report, Royal Institute of Technology, Stockholm, Sweden, 2015

[8] NIST, National Institute of Standards and Technologies, "Interfaces for Personal Identity Verification, Part 4: The PIV Transitional Interfaces & Data Model Specification", NIST SP 800-73-3, http://csrc.nist.gov/publications/PubsSPs.html#800-73 [Retrieved April 2016]

[9] S. Muftic, "Integrated Crypto Services Provider for Web and Mobile Applications", submitted to the IEEE Conference on Communications and Network Security, (2015).

[10] H. Zhao, "Secure Management of Multi–Application Mobile Platforms", Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, June 2013.

[11] GSA, US Federal Government, "FIPS 201 Evaluation Program", http://www.idmanagement.gov/about-fips-201-ep-program [Retrieved Jan 2016]

[12] "Wireless Application Protocol: Public Key Infrastructure Definition", WAP Forum. April 24, 2001.

[13] D. v. Thanh, T. Jonvik, B. Feng, D. v. Thuan, I, Jorstad, "Ubisafe: Simple Strong Authentication for Internet Applications using Mobile Phones", Proceedings of the IEEE "GlobeComm" (2008).

[14] K. Saravana, T. Vaisbnavi, "Rabin Public Key Cryptosystem for Mobile Authentication", Proceedings of the IEEE International Conference On Advances In Engineering, Science And Management, (2012).

[15] K. W. Park, S. S. Lim, K. H. Park, "Computationally Efficient PKI-based Single Sign-On Protocol (PKASSO) for Mobile Devices", IEEE Trans. Comp., **57**(6): (2008).

[16] E. Trichina, K. Hypponen, M. Hassinen. ″SIM-enabled Open Mobile Payment System Based on Nation-wide PKI″, ISSE 2007 Securing Electronic Business Processes, 355-366 (2007).

[17] S. Vishwakarma, P. Kumar Samant, A. Sharma, "Attacks in a PKI-Based Architecture for M-Commerce" IEEE International Conference on Computational Intelligence & Communication Technology, 52-56 (2015).

[18] Chun, I. Jeun and Kilsoo, ″Mobile-PKI Service Model for Ubiquitous Environment″ Comm. Comp. Info. Sci. **12**: 118-124 (2008).

[19] Y. Lee, J. Lee and J. Song, ″Design and implementation of wireless PKI technology suitable for mobile phone in mobile-commerce″, Comp. Comm., **30**(4): 893-903 (2006).

[20] R. Gupta, R. Garg, "Mobile Applications modeling and security handling Cloud-centric Internet of Things", Second International Conference on Advances in Computing and Communication Engineering, 285-290 (2015).