Trust Establishing Model in IoT using PKI and Timestamp

Ivan Vulić*, Radomir Prodanović**, Gradimir Vukčević***, Stefan Sretenović**
* University of Defence, Belgrade, Serbia
** Centre for Applied Mathematics and Electronics, Serbian Armed Forces, Belgrade, Serbia
*** The School of Electrical and Computer Engineering of Applied Studies, Belgrade, Serbia
ivan.vulic@mod.gov.rs, radomir.prodanovic@vs.rs, gradimirv@viser.edu.rs, stefan.sretenovic@hotmail.com

***Abstract*— IoT system gathers data from physical devices integrated into a data exchange network. Upon analysis, intelligent systems or humans make decisions based on the analysis performed. The quality of decision made depends on the trustworthiness of the data sources. One of the key elements in the decision-making process is trustworthy interaction between IoT entity and time information. In order to make timely decision, it is necessary to be certain about the time of the data gathering. The authors apply PKI technology in order to secure trust between all parties in the communications and timestamping technology to verify digital data in specific time in trustworthy and verifiable way.**

I. INTRODUCTION

The Internet of Things (IoT) is a paradigm where everyday objects can be equipped with identifying, sensing, networking and processing capabilities that will allow them to communicate with one another and with other devices and services over the Internet to accomplish some objective [1].

Based on obtained data, IoT system or users, make decisions on further actions. Na osnovu dobijenih podataka, IoT sistem ili korisnik donose odluku o daljem delovanju. Quality of a decision made and service provided depends on the data obtained, that is on the trust in the IoT entity and data.

Trust in the IoT entity should be such that it should be certain that it is an entity as represented, and that a change of data may be confirmed during transmission process. With this trust, prevention of malicious user presentation as a part of IoT system and delivery of malicious data is prevented.

Prior to processing IoT entity data intelligent applications and services should be certain in impossibility of revocation that transaction originates from a specific entity. This trust is necessary in order not to create doubt in processed data and decisions originating from them. Quality decision making is affected by the moment of the data collection. Decision made by the data collected in an inappropriate moment for sure is not a good decision and affects the final result. Therefore, trust in the moment of data or content creation is of importance for relevance and quality of a decision.

IoT is a complex environment with interaction of various entities within the same environment or with interaction of entities of different environments. It is a challenge how to achieve a trust between entities in IoT environment. In order to be certain that data originate from the right entity, it is necessary to establish trust that all entities participating in the interaction trust that data originate from the real source and that they are generated at the original generation time. Established trust is not permanent and it is necessary to enable trust management. Authors think that the trust in the system must be considered from the time perspective, also, as it is time bound.

In this paper, authors rely on PKI and timestamping technology in the process of establishment of trust between IoT entities. In the proposed model, the authors introduce trust token as a tool for presentation of entities to one another and a guarantee that the content has not been altered or generated after the specific time. The proposed trust model enables time bound trust. Rest of the paper includes: the section defining trust in IoT and early defined trust models; section three and section four describing applied public key infrastructure technologies and trusted timestamping; section five describing proposed trust model in IoT and model analysis and section six providing a conclusion.

II. TRUST AND IOT TRUST MODELS

A. Defining Trust in the IoT

Trust, in its common understanding, is a human feeling affecting decision and behavior. According to [2], trust can be decomposed into device trust, entity trust, and data trust. Device trust in the IoT is a challenge, as a priori trust in devices cannot always be established, e.g., due to high dynamics and cross domain relations. Entity trust in the IoT refers to the expected behavior of participants such as persons or services. In the paper [3] the authors state that the trust in data in IoT may appear in two ways. First, reliable data may be obtained from unreliable sources by aggregation. Second, IoT services themselves may create data that need to be evaluated for trustworthiness.

Among the many different definitions and contexts of trust [4], the authors focus on establishment of trust between entities no matter of the type of entities introducing the factor of time within the trust.

B. Trust Models in IoT

Trust management has a very important role in IoT in order to reliable gather and process data, use qualified services, improved privacy of users and safety of information. The reference [5] defines model with the following characteristics: trust relationship and decision, data perception trust, privacy preservation, data fusion and mining trust, data transmission and communication trust, quality of IoT services, system security and robustness,

generality, human–computer trust interaction and Identity trust - IT.

The reference [6] proposes autonomous model of trust with the agent and agent platform in each of the nods by application of TAEC (Trustworthy Agent Execution Chip) architecture that uses highly safe software and hardware platform for secure agent functioning.

The reference [7] presents TRM-IoT trust and reputation model in order to strengthen cooperation between single elements in IoT/CPS networks and the cooperation rely on behaviour on these single elements.

Trust model should be modelled by daily trust model between people in everyday life; therefore, the proposed model is the one for the service provision for the needs for IoT, as in [8].

The reference [9] elaborates four trust dimensions: device trust, processing trust (service provider), connection trust and system trust (overall perspective), while the reference [10] considers that the key to development of appropriate algorithms and models based on logic measurements and trust calculations are trust and reputations mechanisms.

The reference [11] develops general framework for trust management for IoT using formal language based on semantics in a way that IoT is viewed through three layers: sensors, network and application.

All above mention papers look at trust through the concept of trust management in IoT. Time dimension is not considered, Application of trustworthiness by public key and digital signature is considered, as in [12]. The trust is transmitted through transitivity via nods, that is by creation of trust chain. IoT entities from the same or from different environments may have certificates issued by different certification authorities. It is not considered how to establish trust between entities with certificates issued by different certification authorities. Also, time factor of trustworthiness is not considered.

The reference [13] proposes IoT architecture that includes a general systematic network and application security through basic requests of the data safety. Papers do not elaborate on how IoT integrate technology of digital signature and time stamp.

III. THE PUBLIC KEY INFRASTRUCTURE

Public Key Infrastructure (PKI) is a complex system that consists of hardware, software, people, policies and procedures necessary for the creation, management, distribution, use, PKI storage and revocation of electronic [14].

PKI enables the establishment of connections between public keys and entities (in the form of certificates), check the connection by other entities and service necessary for key management in distributed systems.

A. The Components of PKI architecture

The model of PKI architecture is composed of five components specified in [14]: the certification authority, the registration authority, PKI Repositories, archive, end entities and their mutual relationships. The model of PKI architecture, their functional components and interconnection are shown in Fig 1.

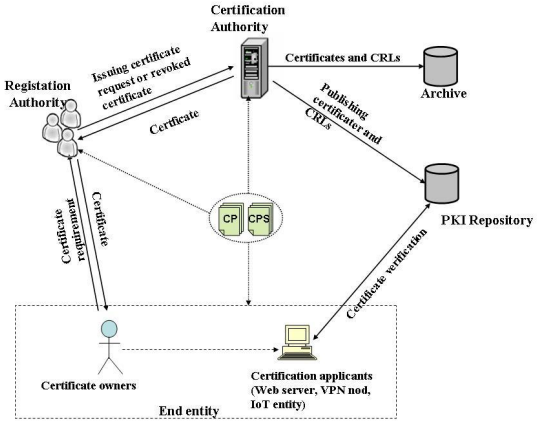

Figure 1. The relationship between PKI components

The Certification Authority (CA) is a collection of computer hardware, software and human resources. It is responsible for issuing certificates (created and signed) management information on the status of certificates and Certificate Revocation List (CRL), publish certificates and CRLs, and archive management of expired certificates. The CA can delegate responsibilities to other infrastructure components, such as registration authority.

Registration Authority (RA) [15] is a confidential representative of CA responsible for verifying the identity of an applicant for a certificate. In addition, RA can perform other functions which CA conveyed it, as well as providing reports of revoked certificates, generating key pairs or archiving keys. RA cannot issue certificates and generate a CRL.

PKI repository provides storage of certificates and information about their status. PKI database must fulfill the following requirements: a simple and standardized approach, modern way of data storage, built-in protection, data management and the possibility of storing similar data. The database is implemented as a directory according to standard X.500. The directory storages and distributes certificates and manages their changes. PKI applications access the directory across LDAP (Lightweight Directory Access Protocol) [16] protocol which is a customized version of DAP (Directory Access Protocol) protocol.

Archive. The archives are stored CA certificates for a longer period of time. Archive must guarantee that certificates have not been and will not be changed while they are in the archive. Before the certificate is issued by CA archives is necessary to determine whether the certificate comes from CA and is valid. The certificates are stored in the archives that some signature older documents could be verified.

End Entity (End-Entity, EE) is defined as a user of PKI certificates and / or end user of system that is the subject of the certificate [14]. In other words, the PKI system, the end entity is a general term for subject that uses any services or functionality of the PKI system and can be the owner of the certificates (individuals, organizations or other entities) or the applicant (may be an application, service, CA, ...) a certificate or CRL.

Certificate. The purpose of the certificate is to establish a link between an identified (the notified) the entity and the public key, indirectly with corresponding private key

entity. This is accomplished when CA uses its private key for signing the certificate, so that certificate can be latter verifiers by any entity which has the public key CA. Today we use version v3 of the X.509 standard for the structure the certificate which is specified by the IETF [17].

Certification Policy (CP). Certification policy defines overall conditions PKI participants have to fulfill in order to be allowed to work within PKI. Usually CP describes benefits of the certificates issued and categories of individuals and organizations that can participate in the PKI.

Certification Practice Statements (CPS). CPS describes working rules and procedures applied by one or more CAs. CPS encompasses the same chapters as CP but elaborates them in more detail in order to make users acquainted with described procedures and applied security mechanisms thus building confidence with CA's services.

B. Digital signing and hash functions

Hash function is a function performing compression of input content into significantly smaller size output content. The values returned by a hash function are called hash values, hash codes, digests, or simply hashes.

Cryptographic hash function has special characteristics suitable for cryptographic use. One of these functions used for digital signing belongs to the family of one-way functions. This family of functions within has functions has the characteristics of simple calculations of any input, while calculated output, which is random, cannot be converted into input. It is characteristics that a minimal change in the input leads to the significant change of the output.

Definition [18]: A family of one-way hash functions is an infinite set of functions $\{h_l\}$ such that the functions $h_l : \{0,1\}^* \rightarrow \{0,1\}^l$ have the following properties:

- There exists a polynomial P such that for each integer l, $h_l(x)$ is computable in time $P(l, |x|)$ for all $x \in \{0,1\}^*$.
- There is no polynomial P such that there exists a probabilistic polynomial time algorithm which, for all sufficiently large l, will when given l and some $x \in \{0,1\}^*$, find an $x' \in \{0,1\}^*$ such that $h_l(x) = h_l(x')$ with probability greater than $1/P(l)$ when x is chosen uniformly among all elements of $\{0,1\}^{|x|}$.

By applying a one-way hash function to a document, it becomes computationally infeasible to find another document which hashes to the same value. For this reason, authors utilize one-way hash functions in their model, such as SHA-1 (Secure Hash Algorithm), SHA-224, SHA-256, SHA-512 and RIPEMD-160.

One of the fundamental tools that will be used for our digital signing and time-stamping systems is that of one-way hash functions.

Digital signature is a mathematics scheme for realization of authentication of digital messages or documents. Relevant digital signature indicates authentication to a receiver, non-repudiation and integrity of the message.

A digital signature scheme typically consists of 3 algorithms:

- Generator of a pair of keys (private and public cryptographic key);
- Algorithm for signing. This algorithm generates digital signature by given message and private key;
- Algorithm for authentication of digital signature. This algorithm verifies or denies the statement that the message is authentic based on the message, public key and signature.

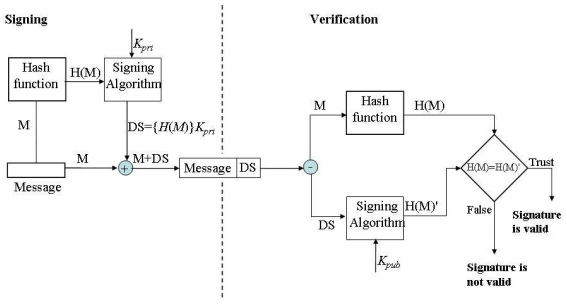

Figure 2. Digital signing and verification of digital signature

More formally, sender generates the pair of the keys (K_{pri}, K_{pub}) and calculates H(M). With the private key K_{pri} he digitally signs $H(M)$. The signed message consists of the pair (M, $\{H(M)\}K_{pri}$). The recipient can use the matching public key K_{pub} to decrypt $\{H(M)\}K_{pri}$ and compare the result to its own computation of $H(M)$. Given that it is infeasible to find a message M' that has the same digestand assuming that only the sender is in possession of K_{pri}, the receiver can be sure that M has indeed been signed by the correct sender and that the received message is the same as the one sent. The Fig. 2 shows the process of signing and verification of the digital signature

Nowadays, standard asymmetric cryptographic algorithms are used for digital signing, like [19]:

- RSA (Rivest Shamir Adleman)) by application of standard PKCS#1 with minimal length of RSA module with n of 1024 bit;
- DSA (Digital Signature Algorithm) with minimal length of parameters p and q of 1024 and 160 bits, respectively;
- ECDSA (Elliptic Curve Digital Signature Algorithm) with minimal length of parameters p and q of 192 and 160 bit, respectively.

IV. TRUSTED TIMESTAMPING

Trusted timestamp is a service of secure time monitoring creation and modification of documents.

A stamping service consists of a set of principals with the Time-Stamping Authority (TSA) and the publication authority (PA) together with a quadruple (S, C, V, P) of protocols. The stamping protocol S is used by a participant to hand over a message to the TSA for time-stamping. During the stamp completion protocol C a participant obtains a time certificate from the TSA. The verification algorithm V is used by a principal having two time certificates to verify the temporal order of the corresponding stamping events. The publication protocol P is used by a TSA to handle the round stamp (short fingerprint of the round) to the PA who will publish it on some authenticated and easily accessible medium [18].

TSA has special device synchronizing the current time with world time and mechanisms for signing time stamps.

Timestamping procedures is conducted in two steps:

- User sends a timestamping request to TSA containing a content to be attributed a timestamp, usually a hash value content.
- TSA delivers generated timestamp to the user.

A. *Timestamping*

The service of timestamping is designed in a way that certified TSA generates time-stamp tokens (TST) – token – contains received cryptographic data stamp and exact time – and is signed digitally, protecting thus the integrity of the token.

The procedure of timestamping generation is standardized by RFC 3161 and is conducted through the set of the following steps, Fig 3.:

- User submitting timestamping request creates hash contents to be attributed timestamps (1);
- User submits request containing hash value to the TSA (2);
- TSA provides that its watch is synchronized with the authoritative time source (3);
- TSA checks that hash value is of correct length, and does not research hash value in any order way in order to secure privacy (4);
- TSA generates TST containing hash value from the request and time sign TST (5). TST is digitally signed by private key TSA used for timestamping purposes (6) and thus trusted timestamp is created.
- TSA adds plain text timestamp to the trusted timestamp and delivers it to the user that submitted the request for timestamping (7) who then keeps it together with the original content (8) for future verification.

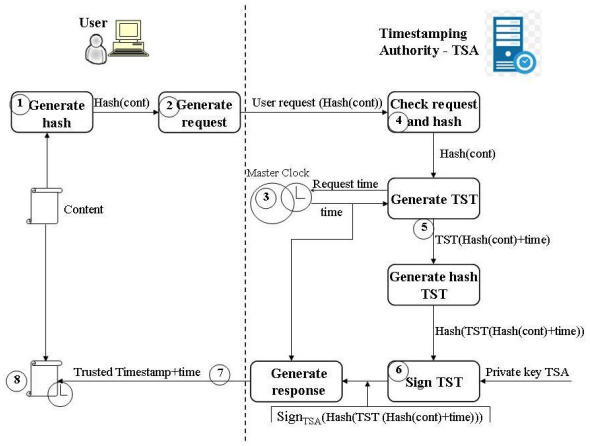

Figure 3. Timestamp Generation

B. *Timestamping verification*

In order to verify timestamp, verification entity does not need TSA. Verification entity needs signed content and TSA certificate. Verification entity performs verification in the following way:

- Verification entity firs separates signed content (1) to: content, trusted timestamp and timestamp.
- Then verification entity decrypts digital signature of trusted timestamp by public key from TSA certificate (2).
- Simultaneously, verification entity generates hash value content (30 and aggregates it with time sign (4). Then, verification entity generates hash value of the newly generated content (5).
- Verification entity compares previously generated hash values with decrypted digital signature (6). If digital signature is good, then is it reliable information that TSA generated TST, that is:
  - That content has not been modified upon timestamping
  - That verification entity knows that hash content has been used for TST generation

That TSA has observed content is the specific moment.

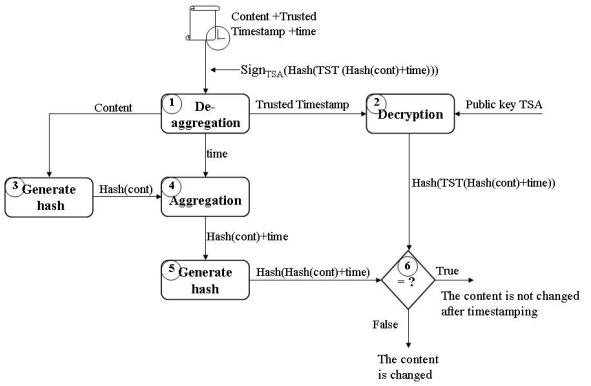

Figure 4. Timestamping Verification

V. PROPOSED TRUST MODEL

Proposed model is based on the trust of IoT entity in the PKI architecture and technology of the timestamping. By the application of PKI digital certificate X.509, IoT entity obtains digital identity to present itself to other IoT entities. Digital certificate of the IoT is signed by the certification authority with its private key. Fig 5. shows IoT trust model based on PKI and time stamp.

A. *Preconditions for Model Application*

In order to establish trust between entities it is necessary to establish trust in the PKI architecture. Trust is established by entity generating a pair of keys (public and private key). Entity submits request for public key generation to the certification authority. Certification authority generates the key, signs it with its private key and delivers it to the entity. Private keys of the entity and certifications authority are not available to other entities involved as the trust is based on the private key.

Trust is verified by checking the trust chain from the certificate of the entity to the certificate of the PKI architecture trust point (for example, trust point of the hierarchy PKI architecture is root certification authority).

Trust in TSA is based on verification of its digital certificate or trust chain if another certification authority has issued certificate.

B. *Elaboration of the Proposed Trust Model*

IoT entity model establishes trust token consisting of two parts: the first part of the trust token (authentication part) includes digital signature of the content and digital certificate, while the second part of the token (time part) includes trusted timestamp token, time stamp and TSA

certificate. With the first part of the trust token, the IoT entity proves to the second part that it has generated the content and trust or lack of trust, while the second part of the trust token proves that the content was not created or has not been modified after generated time.

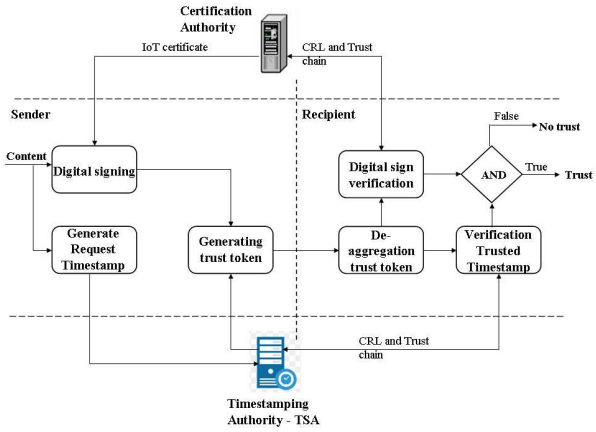

Figure 5. IoT trust model based on PKI and time stamp

Only in case of successful verification of authentication and time parts of the token, the trust in the IoT entity is full. Full trust means trust in the identity of the IoT entity, revocation of delivered content, permanence of the content and time of the content creation.

Within the same IoT environment, IoT entity trust may be managed by management of digital certificate. When in the same IoT environment digital certificate of IoT entity is suspended then other IoT entities have information that there is a temporarily revocation of trust. If digital certificate of IoT entity is revoked, then there is no trust in the entity.

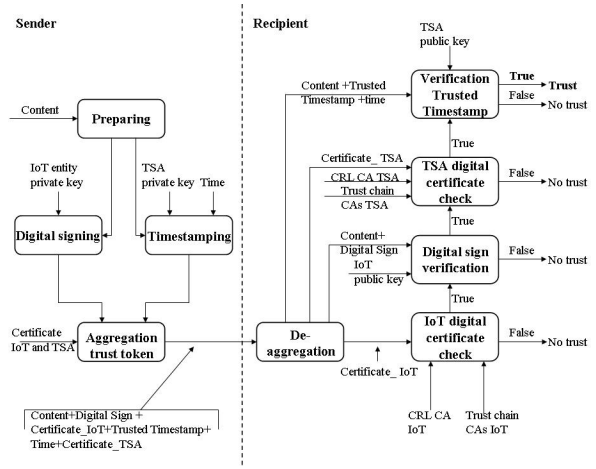

Figure 6. Cycle Diagram of Establishment and Verification of Trust

Fig 6. shows diagram of establishment and verification of trust based on the proposed model. IoT entity to be trusted is attributed digital certificate by the selected certification authority. Using private key corresponding to the public key from the certificate IoT entity digitally sign the content to be sent. Simultaneously, the request for trusted timestamping is sent to the TSA. Trust token is generated by aggregation and is added to the content and delivered to the IoT entity to perform trust verification.

On the side of the receiver, IoT entity performs trust verification so as to firstly perform disaggregation of the trust token. Then, the first part of the trust token related to digital signature is verified. By verification of the certificate expiration time information on digital identity validity of IoT is obtained. Verification of digital certificate of IoT entity and all digital certificates in the chain of trust IoT entity determines if there is a trust in PKI, and by this if the content was indeed signed by the IoT entity digitally presented by the certificate.

Checking the list of revoked certificates IoT entity confirms that: the entity may be trusted, that the trust to the entity has been revoked (certificate revoked) or that the trust has been temporarily revoked (certificate suspended).

With successful verification of digital signature, IoT entity receiver may be certain that the content has not been modified during transmission and that the content has not been signed by anybody else but IoT entity stated in the digital certificate.

At this stage of verification of the trust IoT verification entity has trust:

- That received content has originated from IoT entity sender.
- That content has not been modified during transmission.
- That there is still trust in IoT entity.
- That IoT entity cannot deny generating and sending the content.

In the second phase, verification of time is performed. It is determined if there is trust in TSA by verification of the status of digital certificate and chain of trust of TSA, meaning if TSA is trustworthy at the moment of timestamping. Verification of trusted time stamp determines that the content was not created or was not modified after generated time stamp.

C. Analysis of the Proposed Model

The proposed model is based on the cryptography of the public key with all advantages and disadvantages of the model originating from this characteristic. Advantages are multiple as they do not relate only to the current trust in the communications, but in future trust. Future trust means that we can trust for certain that delivered content has not been altered and that we can trust its origin. Besides, the exact information on the time of the content generation, as well as that the content has not been altered after that time contribute significantly to the quality of future decisions.

In general, disadvantages of the model are related to the trust establishment via third parties which may lead to partial or complete loss of trustworthiness. Having in mind that the trust is based on the private key of the certification authority or TSA, its compromises trustworthiness. However, ASC X9.95 and RFC 3161 [20, 21, 22, 23] provide additional instructions on certificate and security management in order to secure TSA and certification authority not being easily compromised.

The second disadvantage is related to the need to increase hardware resources in order to implement cryptographic functions used for trust establishment. Today technology development and implementation of "lightweight cryptography", should not have an effect on the implementation of the proposed model.

Advantages of the proposed model are:

- Fulfilment of the basic safety requirements, like trust, authentication, integrity, revocation
- Flexibility, as in the possibility of application in devices, software, services, humans (smart cards)
- Scalability – not limited to the specific number of entities
- Mobility of entities – mobility not affecting trust establishment between entities no matter of their locations
- Time dimension of trust – model enables time management of trust and clearly emphasises trust in the moment after which the content has not been altered
- Historic overview of the original content in time – based on the time stamp, it can be precisely shown from which period in time the content has not been altered which is of importance for timely and reliable decision-making
- Transparency of trust establishment between entities from different PKI environments – it is not necessary to create changes in the model, but transparency is achieved on the level of PKI architecture by establishment of some of the interoperability models [24].

VI. CONCLUSION

The paper proposes the model of establishment of trust in IoT based on application of PKI and timestamp. The model enables trust establishment between IoT entities from one environment linked to one or more trust nods.

The model introduces time stamp in order to guarantee that data has not been gathered, processed or altered prior to the time stated in the time stamp, that is after verified by the time stamp. Management of digital certificates enables management of IoT entities' trust.

The proposed model enables implementation of the multi-dimensional trust in IoT environment by device trust, processing trust, connection trust, system trust, data trust and time trust. The authors believe that the proposed model is appropriate for implementation in the IoT systems requiring huge trust necessary for timely and quality decision-making.

REFERENCES

[1] A. Whitmore, A. Agarwal, and L. D. Xu, "The Internet of Things—A survey of topics and trends", Inf Syst Front, vol. 17, pp. 261–274, 2015.

[2] J. Daubert, A. Wiesmaier, and P. Kikiras, "A review on privacy and trust in IoT", in In IOT/CPS-Security Workshop, IEEE International Conference on Communications, ICC 2015, London, 2015.

[3] A.Arabsorkhi, M. S. Haghighi, and R. Ghorbanloo, "A Conceptual Trust Model for the Internet of Things Interactions", 8th International Symposium on Telecommunications (IST), 2016.

[4] Z. Hornák, I. Nyilas, D. Petró, J. Schrammel, P. Wolkerstorfer, et. al., Technology and Standard Report (uTRUSTit project), 2010.

[5] Z. Yan, P. Zhang, and A. V.Vasilakos, "A survey on trust management for Internet of things", Journal of Network and Computer Applications, vol. 42, pp. 120-134, 2014.

[6] X.Xu, N.Bessis, and J.Cao, "An Autonomic Agent Trust Model for IoT systems", Procedia Computer Science, vol. 21, pp. 107-113, 2013.

[7] D. Chen, G. Chang, D.i Sun, and J. Li, "TRM-IoT: A Trust Management Model Based on Fuzzy Reputation for Internet of Things", Computer Science and Information Systems, vol. 8(4), pp.1207-1228, 2011.

[8] A. Arabsorkhi, R. Ghorbanloo, and M. S. Haghighi, "A Conceptual Trust Model for the Internet of Things Interactions", Conference: International Telecommunication Symposium, 2016.

[9] J. Daubert, A.Wiesmaiera, K. Panayotis, "A View on Privacy & Trust in IoT", 2015 IEEE International Conference on Communication Workshop (ICCW), 2015.

[10] T.Abera, N.Asokan, L.Davi, and et al., "Things, Trouble, Trust: On Building Trust in IoT Systems", Proceeding DAC '16 Proceedings of the 53rd Annual Design Automation Conference, 2016.

[11] J. Wang, S. Bin, Y. Yu, and X. Niu, "Distributed trust management mechanism for the internet of things", Appl. Mech. Mater., vol. 347-350 (4), pp: 2463–2467, 2013.

[12] R.Lacuesta, G.Palacios-Navarro, C. Cetina, and et al., "Internet of things: Where to be is to trust", EURASIP Journal on Wireless Communications and Networking 2012, 2012.

[13] H. Ning, H. Liu, and T.L. Yang, "Cyberentity security in the Internet of Things", Computer, vol. 46 (4), pp: 46–53, 2013.

[14] A. Arsenault, and S. Turner, *Internet Draft PKIX: Internet X.509 Public Key Infrastructure: Roadmap*, 2003.

[15] D. E. Sheehy, M.Greene, M. Lundin, and J. Ward, *Trust Service Principles and Criteria for Certification Authorities*, Version 2.0, Canadian Institute of Chartered Accountants, 2011.

[16] H.Johner, S.Fujiwara, S. A.Yeung, and et al, Deploying a Public Key Infrastructure, IBM Redbooks, SG24-5512-00, IBM, 2000.

[17] D. Cooper, S. Santesson, S. Farrell, at all, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, RFC 5280, 2008,

[18] A. Bulda, H. Lipmaa, and B. Schoenmakers, "Optimally Efficient Accountable Time- Stamping", Conference: Proceedings of the Third International Workshop on Practice and Theory in Public Key Cryptography, 1999.

[19] ETSI, *ETSI ESI SR 002 176: Algorithms and Parameters for Secure Electronic Signatures*, 2003.

[20] P. Turner, W. T. Polk, and E. B. Barker, *Preparing for and Responding to CA Compromise and Fraudulent Certificate Issuance*, NIST Publication, 2012.

[21] ETSI, Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements, Draft ETSI EN 319 411-1 V1.2.0 (2017-08), 2017.

[22] C. Adams, P. Cain, D. Pinkas, and R. Zuccherato, Internet x.509 public key infrastructure timestamp protocol (TSP), RFC 3161,The Internet Engineering Task Force (IETF), 2001.

[23] ANSI, Trusted time stamp management and security, Standard X9 x9.95-2005, American National Standards Institute, 2005.

[24] R. Prodanović, I. Vulić, "Model for PKI Interoperability in Serbia" Vojnotehnički Glasnik/ Military Technical Courier, Vol. 65 (2), 2017.