IET Cyber-Physical Systems: Theory & Applications

Research Article

IET Journals
The Institution of Engineering and Technology

Fuzzy logic approach to modelling trust in cloud computing

ISSN 2398-3396
Received on 31st January 2017
Revised 5th April 2017
Accepted on 22nd June 2017
doi: 10.1049/iet-cps.2017.0016
www.ietdl.org

Mary-Jane Sule[1,2] ✉, *Maozhen Li*[1], *Gareth Taylor*[3], *Clement Onime*[4]

[1] *Department of Electronic and Computer Engineering, Brunel University London, Uxbridge, UK*
[2] *Current affiliation: Department of Computer Science, University of Jos, Nigeria*
[3] *Brunel Institute of Power Systems, Brunel University London, Uxbridge, UK*
[4] *Information and Communication Technology Section, International Centre for Theoretical Physics, Trieste, Italy*
✉ *E-mail: mjaysule@gmail.com*

Abstract: Despite the growing deployment of mission critical applications on computing systems, trust and security continues to hinder its full adoption and deployment on cloud computing platforms. In addition to accountability and non-repudiation on the cloud deployment, end-users want to be confident of availability and reliability of services. For any cloud platform to be secure and trusted, the individual layers of the platform must be secure as there is no 'one fits all solution' for securing all the layers. This work presents a multi-layer trust security model (MLTSM) based on unified cloud platform trust that employs a fuzzy logic combination of on-demand states of several different security mechanisms, such as identification, direct and in-direct trust, across all cloud layers. In addition, results from a MATLAB-based simulation of the model are also presented. A MLTSM can improve the secure deployment of cloud infrastructure in mission critical sectors such as electrical power system operation, as it provides empirical evidence that allows direct (on-demand) determination and verification of the trust state of any given cloud computing platform or service. Such a modelling approach is useful for comparison, classification and improving end-user confidence in selecting or consuming cloud computing resources.

1 Introduction

Cloud computing service providers continue to leverage on the ability of the cloud to provide scalable, on demand, pay as you go, virtualised computing resources to users [1]. As its popularity continues to grow, many mission critical applications are now gradually being deployed over a variety of cloud services [2]. Security concerns though continue to be a deterrent to a full dynamic interaction and deployment of the various cloud services [3]. End-users are unable to fully adopt the cloud platform unlike other computing technologies based on concerns about transparency, accountability and governance [4].

For many end-users, providers offer a service level agreement (SLA) that is most often useful to guarantee agreed performance or service levels. An SLA may sometimes include tools for monitoring and measuring performance targets, though the agreements could also even lock-in an end-user to some proprietary service, application or provider. While, an SLA could provide some assurance about security policies such as data-centre-policy, it does not guarantee that all established security mechanisms are continuously in place nor can it guard against negligent activities that could result in security related breaches or threats.

A user wants to be sure among other things that the consumed (or accessed) service is free from unauthorised disclosure or modification of data or information. The user wants to be confident that the service is available and reliable, and that there is accountability and non-repudiation: that is, tamper-proof evidence that proves the originality and integrity of data. This is only possible when cloud end-users can verify the security state of the services they are accessing, whether across the different cloud layers or physical locations.

Like in other computing systems, detection and prevention of unauthorised access and actions is fundamental for cloud computing security. Indeed, several security protocols and tools have been enhanced and adopted to fit cloud computing, few of these take into account specific issues as it relates to the cloud end-users [4, 5]. While it is critical for providers to continue to secure the cloud platform and deployments effectively as users take advantage of the operational and financial benefits of cloud for their services, it is also important that users can evaluate and attest to the configured security mechanisms across all layers of the cloud platform. It is this evaluation that provides the end-user with information whether to trust the platform or not.

Trust can be defined as an act of faith, believe or confidence and reliance that the system or component which the user is intending to interact with will behave as expected or anticipated [6]. Based on first-hand empirical information, it then becomes possible for an end-user to directly and quickly trust a provider who not only has a proven background but also satisfies some security scenarios required by the end-user.

Cloud platforms and services are implemented in layers, therefore, for any platform to be considered secure and trusted, the security mechanisms across every layer must be enforced and configured as it is the individual security mechanism on each layer that make up the overall security status of the cloud platform and therefore each layer must be properly secure as there is no 'one fits all' security solution that would be applied on the platform but all the layers contribute to the security of the whole platform. Three popular layers of the cloud are – infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS). The IaaS is the most important layer as it is the underlying layer of all cloud platforms; therefore, the presence or absence of an adequately secured IaaS will affect the overall trust level of the platform or services running over it irrespective of the context in which the platform is being accessed. A similar consideration applies to both PaaS and SaaS, where it may be argued that security is a combination of specific layer security mechanisms and the security of that of the underlying layer(s). A subjective aggregation or evaluation of various security mechanisms configured across the layers of a cloud platform would provide useful information on the trustworthy state of the platform [7, 8].

This paper presents a fuzzy logic approach to modelling cloud trusts based on a multi-layer evaluation and aggregation of some well-known security mechanisms. The next section presents some related works, a developed multi-layer security trust model (MLSTM) is presented in Section 3. Results of a brief evaluation of the model based on a MATLAB simulation is presented in Section 4 and Section 5 concludes the paper.

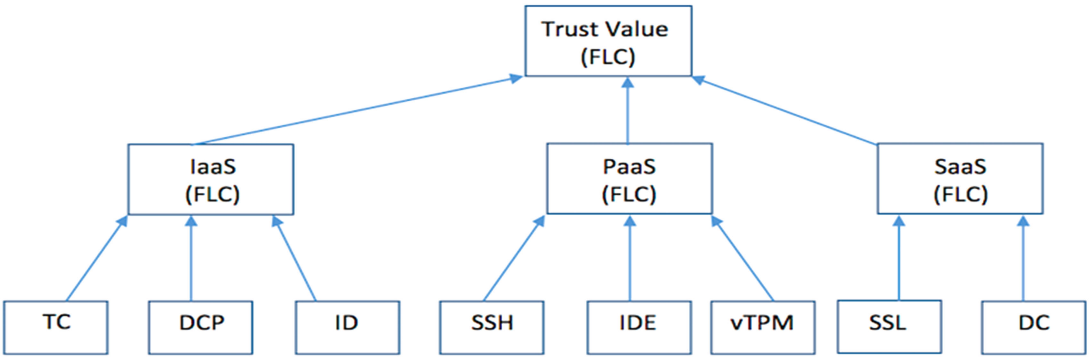

Fig. 1 *MLSTM concept*

2 Related work

Security and trust related research though not new is still an emerging field in cloud computing. Considering that cloud computing itself is an evolving and unique technology, serving a variety of users with various needs and demands, a single security architecture may be impossible to achieve [1, 9]. Most cloud security related research tend to focus more on the IaaS with limited or no considerations for other layers that make up the cloud platform.

Huang and Nicol [6] used a subjective logic approach to evaluate the trust status of a cloud platform. While Shaikh and Sasikumar [10] list some parameters necessary for measuring the overall security of a cloud platform and the deployed service, they only provided limited information about implementing their framework. Xu in [11] ascertained that a platform can only be secured when all players or stakeholders work together and [11] also considered a fuzzy reputation for trust management in cloud based on detection of malicious attacks with some set of metrics. Fan and Perros in [12] considered objective and subjective trustworthiness, with subjective trust based on SLA(s) and some quality of service attributes. This approach requires that end-users depend on some third party trust brokers or providers. Our approach instead focuses on direct evaluation by end-users themselves as this eliminates doubts that trust ratings are biased and may not be completely relevant to the end-user's desired scenarios.

Banirostam *et al*. [13] presented a trust-based approach using trusted computing (TC) which is applied only at the IaaS layer, and as mentioned earlier a cloud end-user also wants security mechanisms configured across the layers of a cloud platform to enable the end-user make a more informed decision. Yang *et al*. [14] proposed data access control mechanism but the security considerations were not sufficient. While Gu *et al*. [15] extended the TC chain of trust (CoT) from the physical infrastructure domain (or IaaS layer) to the PaaS layer. It is clear here that, trust can only be evaluated on cloud platforms that have implanted this extension.

It is also possible to protect cloud infrastructure using TC (without extensions) as discussed and implemented in [2]. In TC based on the trusted platform module (TPM), the CoT from the TPM hardware device through the firmware (BIOS), boot loader (kernel) and operating system of the cloud middle-ware may be extended to cover virtual storage devices. When TC is used to created trusted clouds, the attestation of the cloud-infrastructure (based on the TPM hardware chain-of-trust) from a running VM could be carried out as a single TC operation targeted at the local controller over a secure-shell connection. The inclusion of a software-based virtual TPM (vTPM) device within the instances allows a new and unique CoT to be built for the kernel, operating system and software components of the instance [2].

Mohsenzadeh and Motameni [16] calculated trust based on historic, direct and recommended (in-direct) values, they do appear to consider attributed trust that may be derived from the identified behaviour of the platform, although, their proposed model does not appear to cater for diverse end-users needs and requirements.

The related works discussed in this section show the current layer specific approach to security and trust related research in cloud computing and also highlights the limited consideration of distinct end-users needs and requirements for security. The next section presents a MLSTM that derives trust based on fuzzy logic combinations of different security mechanisms from various cloud layers in a flexible manner that allows easy customisation to fit the needs and requirements of individual end-users.

3 Multi-layer security trust model

The security assessment of a cloud platform should be of paramount importance to any user regardless of the service required. In this section, we propose a cloud security trust model that is based on fuzzy logic control (FLC) system with the following characteristics [17]:

- FLC aggregation of states from several layer specific security mechanisms across different cloud layers.
- State changes are based on Gaussian fuzzy numbers.
- Various operators are used to represent the rules.
- The overall control action of the system is computed to reflect the accumulated security strength.

As show in Fig. 1, MLSTM cuts across the three well-known layers of a cloud platform – IaaS, PaaS and SaaS. The trust evaluation at any layer is derived from the identified behaviour of individual security mechanisms at distinct layers of the platform. Various mechanisms can be applied on a cloud platform to enforce security as applying only a particular traditional security mechanism is unsuitable for a cloud computing platform [18].

For the IaaS layer, the trust value is obtained from an FLC combination of three security mechanisms, namely data-centre-policy (DCP), TC and group-based intrusion detection/prevention (ID). In general, the physical environment of the host (server) may be protected by a DCP, which ensures that the server host is protected against power-outages, loss of connectivity and even limits access to authorised personnel. Typically, SLAs may be used to monitor and guarantee the enforcement of an adequate DCP. Furthermore, the server (host) computer as part of the IaaS, may be secured using a suitable mechanism such as TC, which refers to the TPM dependent CoT that is built from the cryptographic storage of measurements for the various component parts of a computing platform including BIOS, boot-loader, O.S. kernel, system libraries and/or virtualisation middleware.

Finally, IaaS layers are expected to include some form of network ID and prevention in the form of security zones or security groups that serve primarily to insulate or prevent unauthorised communications or interactions between instances that belong to different security groups or zones. Although there are other security mechanisms for the IaaS layer, these three (DCP, TC and ID) are either commonly found on all cloud implementations or easy to add [19].

Similarly, PaaS layer trust value is obtained from the FLC combination of verifications of three security mechanisms: secure shell (SSH), ID engine (IDE) and vTPM. The latter is a software-based TPM device within an instance that allows a new and unique CoT to be built for the kernel, operating system and software components of the instance. PaaS layer access to an instance is typically over some secure channel such as SSH, which includes a direct (peer-to-peer) key-based verification process before securing the communication channel using cryptographic encryption [20]. SSH connections depend on identity trust relationship based on possession of matched credentials for a direct-trust between two parties if the remote host is already known and fully trusted. In SSH, the first connection between two parties may follow an assumptive trust relationship/model as the remote host may be initially unknown.

The security state of an access service such as SSH may be derived from a consideration of its patch level. In a high-security context, an instance may also include an IDE to ensure that contents (applications, files and data) of the instance have not been tampered with or have only undergone authorised modification or changes. The results of a verification process conducted by an IDE and a verification based on an (internal) vTPM can provide information about the integrity of various internal parts of a PaaS instance. The vTPM security mechanism provides TC like protection within the virtual machine or instance. That is, a CoT is built from the cryptographic storage of measurements for the various component parts of the instance including BIOS, boot-loader, O.S. kernel, system libraries and/or applications.

The SaaS Layer trust value is obtained from the FLC combination of two security mechanisms namely secure socket layer (SSL) and data colouring (DC). Connections to SaaS services are commonly protected using SSL, which is based on the use of digital certificates for both identification and cryptographically securing the communications channel. The SSL certificates may be self-signed for arbitrary (take-it-or-leave-it) trust or based on external third party certification authorities for in-direct trust. At the application level, data at the SaaS layer may be protected against theft or loss using a suitable mechanism such as hashing, data-colouring, encryption or obfuscation. Data-colouring is chosen as an optimal technique for cloud computing platforms or applications because coloured data may still be processed without overhead (the colouring is transparent) while also providing the ability to detect tampering, identifying and reporting data-loss. Considering security related metrics of confidentiality, integrity and availability in a cloud context, the coloured-data (output of a data-colouring process) is able to maintain integrity and availability of data during processing. While, hashing as a data-protection technique can only provide integrity, obfuscation provides only limited confidentiality; data-colouring provides integrity and availability; while encryption provides confidentiality and integrity but not availability [21].

In a high security context, it is possible that DC is used to secure data for cloud-based processing and storage. In an implementation of DC, the original data is coloured (via steganographic techniques) using digital bits that can uniquely identify the data-owner, cloud-service and data-recipient. In the case of data loss or theft, DC can help to highlight the path through which the loss occurred. The choice of SSL is governed by its popularity as it is widely used for secure access to many on-line services; While DC is relatively new, it was chosen for its transparency during cloud processing.

In the multi-layer trust security model (MLTSM), the end-user is presented with a single trust value derived from the FLC combination of trust values from IaaS, PaaS and SaaS layers which may be used to decide if platform is trustworthy or not.

The use of a security mechanism-based approach in the MLTSM is based on the following consideration from set theory:

Assume a given cloud (x) is considered secure if it is a member of the set of secure clouds (X) as presented by (1), while the converse is true that is a given cloud platform is NOT secure when it is NOT a member of the set of secure clouds (X) as shown in (2). Equation (1) is a crisp representation that imposes a sharp boundary on a set where each member of a set is assigned 1 while a member outside the set is assigned a value of 0, that is in crisp representation an element either belongs to a set or does not. This simply means if a given cloud is true for 1 that means the cloud belongs to ($\in$) a set of secure clouds and if the given cloud is true for 0 that means the cloud does not belong to ($\notin$) a set of secure cloud, given that 1 means 'secure' or 'true' and 0 means 'not secure' or 'false'

$$x \in X\text{is true for 1} \tag{1}$$

or

$$x \ \notin X \text{ is true for 0} \tag{2}$$

Considering the security of cloud (x) as a super-set of distinct security mechanisms arranged by layers (m_{ly}), and membership of X (secure clouds) is based on a given set of security mechanisms (m_i), it becomes possible for a matching sub-set of security mechanisms of a cloud platform x to satisfy the membership requirements of a given secure cloud (X) while some other part of the same cloud service do not. That is, cloud (x) is secure when $m_i \cup m_{ly} = 1$ or true for all items of m_i. In other words, an individual end-user considers a cloud service secure when it can satisfactorily match all items of his defined membership set.

The choice of fuzzy logic is informed by its ability to represent information with some varying (non-crisp) degree of membership. This is important in evaluating the trustworthiness of the cloud platform and that of the deployed service in an ever-changing or dynamic world where even security is not static but ever evolving. In fuzzy logic, membership of a given element in a set is determined as a fractional value between 0 and 1 known as the degree of membership, which conveys an idea of how much of that element is contained within a given set.

In a security context, the value 0 could represent 'low security' and 1 for high security; any membership range ~0.5 can then be considered to be of medium value or security. A crisp value can only be a 0 or 1 which represents the absence or presence of a security mechanism in a cloud layer. Therefore, for each cloud layer under consideration, an FLC value may be derived by the combination of several chosen security mechanisms. For example, a user might choose to evaluate a cloud platform based on TC, ID, DCP; SSH, IDE and vTPM and not bothered about DC (data colouring). Table 1 presents an analysis of the MLSTM FLC system, where as shown, the corresponding linguistic values of the inputs (IaaS, PaaS and SaaS), are combined using fuzzy (fired) rules into a trust value and their corresponding fuzzy levels computed with t-norm product to obtain the corresponding crisp values.

For an arbitrary trust level say 0.5, the degree of membership function is given by

$$f = \begin{cases} 2\ [\ \mu_A(s)]^2, & 0\ \le\ \mu_A \le 0.5 \\ 1 - 2\ [1 - \mu_A(s)]^2, & 0.5\ \le\ \mu_A \le 1 \end{cases} \tag{3}$$

where f represents function, μ represents degree of membership , s represents security configuration,
A represents crisp set that can only take two values – 0 and 1

With the Gaussian membership function:

$$\mu_{A_i}(x) = \exp\left(-\frac{(c_i - x)^2}{2\sigma_i^2}\right) \tag{4}$$

In (4), σ is the standard deviation c is the centre of the ith fuzzy set of A_i (the membership function always returns values in the range of 0 and 1).

Table 1, the FLC system analysis table shows the association between cloud layers, membership degree values and crisp

Table 1 FLC System analysis table

Cloud layer	Membership (μ) range	Fuzzy level	Crisp value
IaaS	0.5–1.0	medium	0.5927
PaaS	0.5–1.0	medium	0.5935
SaaS	0.0–0.5	medium	0.6324
trust		high	0.7323

Table 2 MLSTM categories

Category	Combination of security mechanisms							
	IaaS layer			PaaS layer			SaaS layer	
	TC	ID	DCP	SSH	IDE	vTPM	SSL	DC
high security	1	1	1	1	1	1	1	1
normal security	1	0	1	1	0	1	1	0
some-how secure	0	0	1	1	1	0	1	0
low security	0	0	1	1	0	0	1	0

Table 3 MLSTM transaction combinations and sequencing by category

Trans/seq	Combination of transactions			
	High security	Normal security	Some-how secure	Low security
1	TC	DCP	DCP	DCP
2	TC + ID	DCP + SSH	DCP + SSH	DCP + SSH
3	TC + ID + DCP	DCP + SSH + SSL	DCP + SSL	DCP + SSH + SSL
4	TC + ID + DCP + SSH	DCP + TC	DCP + SSH + SSL	DCP + SSH + SSL
5	TC + ID + DCP + SSH + IDE	DCP + SSH + TC	DCP + IDE	DCP + SSH + SSL
6	TC + ID + DCP + SSH + IDE + vTPM	DCP + SSH + vTPM + SSL	DCP + IDE + SSL	DCP + SSH + SSL
7	TC + ID + DCP + SSH + IDE + vTPM + SSL	DCP + TC + SSH + vTPM	DCP + IDE + SSH	DCP + SSH + SSL
8	TC + ID + DCP + SSH + IDE + vTPM + SSL + DC	DCP + TC + SSH + vTPM + SSL	DCP + IDE + SSH + SSL	DCP + SSH + SSL

interpretive values used for the MLSTM. The computed crisp trust value from the MLSTM is 0.7323.

The degree of membership from fuzzy logic can be used to support vague concepts and model real world situations including the dynamic evolution and changing nature of security of a cloud platform with much higher accuracy compared with a crisp representation. That is, thanks to the fuzzy representation, it becomes possible to say a given cloud platform is x% secure or y% unsecure.

Alternatively, based on the use of fuzzy representations, a cloud platform cannot be said to be completely secure, it may be secure to a certain degree or level even when the components that make up the system are assumed to be fully secured or completely unsecure.

Based on fuzzy logic, the degree is usually a real number between the range of 0 and 1. Unlike in crisp representation of cloud security that can only provide a binary ('true' or 'false'; 1 or 0) answer to the question of 'Can I trust the cloud platform?', the fuzzy representation goes further and can provide an answer to the question of 'how trustworthy is the platform?' even when presented with diverse or varying requirements.

With fuzzy logic, any given cloud platform would have a varying degree of membership in two distinct universal sets of secure-clouds and unsecure-clouds. For an element with varying degree of membership in two different sets, the membership value in the resulting intersection (fuzzy AND) of both sets would be the lower of both membership values, while the membership value in a union (fuzzy OR) of both sets would be the higher value.

That is, the fuzzy OR operation is given by

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \ \mu_B(x)\} \tag{5}$$

While, the fuzzy AND operation is given by

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \ \mu_B(x)\} \tag{6}$$

The MLSTM is a tool for assessing and evaluating the diverse security concerns related to cloud services and can provide users with the ability to evaluate the security of a chosen cloud platform as part of the process of establishing trust. This section discussed the architectural concept of the MLSTM along with some implementation details that show its flexibility in satisfying varying end-user requirements along with the fuzzy logic approach to aggregating security states. The next section presents some results obtained from a MATLAB simulation of the MLSTM based on eight different security mechanisms from the IaaS, PaaS and SaaS layers.

4 Results and evaluation

A MATLAB-based simulation was developed for the MLSTM [22, 23]. Working from the end-user perspective, the previously described eight security mechanisms were combined into four separate categories namely high, normal, some-how and low security requirements as shown in Table 2. An end-user with low security needs would be content with testing and satisfying DCP, SSH and SSL security mechanisms, a high security user would test and satisfy all eight mechanisms.

The combination for each category was based on the weight each security configurations carries. Each security configuration was assigned a weight during the research. For a high-security category, it is paramount to have all eight identified security mechanisms present. For a normal security category, the cloud platform can be considered to be of normal security if across the different layers one or two of the security mechanisms are not present. In the course of this research, the absence of the following mechanisms across the different layers *data colouring* (SaaS), IDE (PaaS) *and ID* (IaaS) would make the platform to be considered as fit for normal security and not high security. In event only *DCP*(IaaS), *SSH, IDE* (PaaS) and *SSL* (SaaS) are configured on a platform then that platform would be considered to be 'some-how secure' and a platform would be considered to be of low security if only the following security mechanisms have been configured or are present – a *DCP* (IaaS), *SSH* (PaaS) and *SSL* (SaaS).

In addition, a set of eight MLSTM transactions defined as the controlled combinations and sequencing of specific probes were developed for collecting verification results for the security mechanisms under test where the '+' sign represents a fuzzy logic combination of the security mechanism for each sequence 1–8. Table 3 shows the eight MLSTM transactions for each category from Table 2.

Fig. 2 shows the trust values obtained from the eight MLSTM transactions for each security category. As shown, after only four MLSTM transactions, it is possible to identify with reasonable accuracy cloud computing platforms or services that satisfy the high or normal security categories, and after seven transactions all categories may be accurately identified.

Fig. 3 shows the success interaction rate comparison between the MLSTM, trust model fuzzy mathematics (TMFM) and the dynamic model trust C (DMTC) [16]. The MLSTM high success rate derived from a MATLAB simulation may be attributed to its use of special sequencing of eight specific transactions devoted to

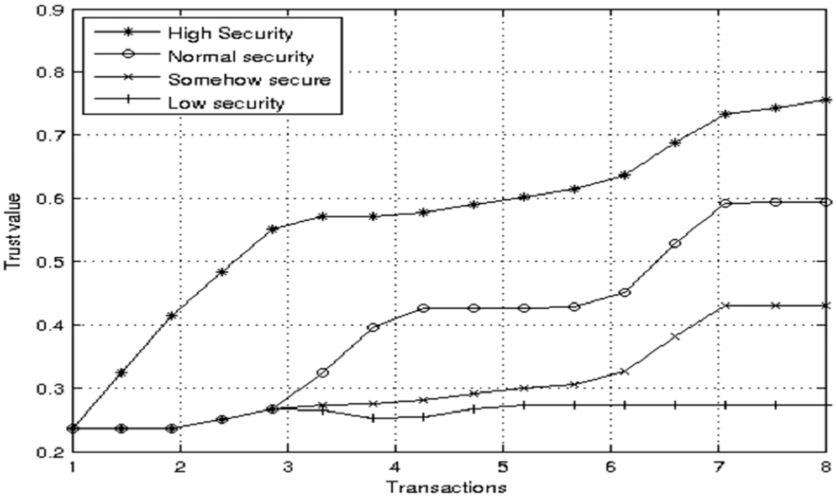

Fig. 2 *Multi-layer security trust for four identified categories*

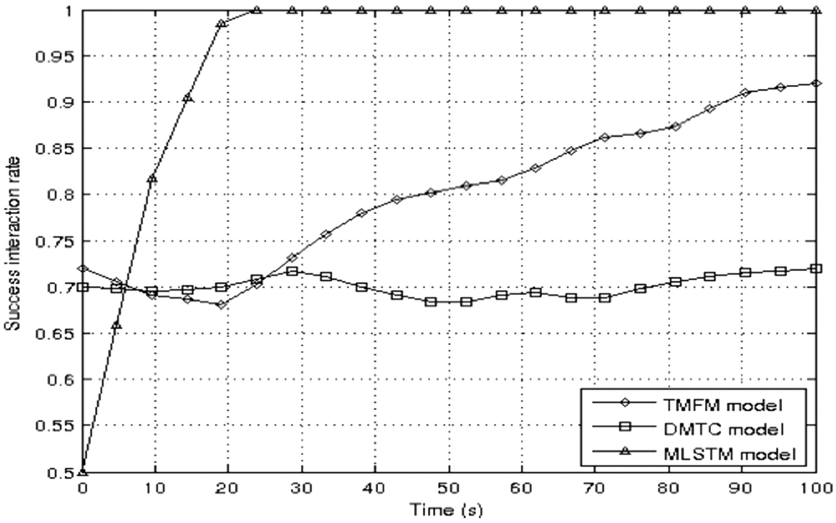

Fig. 3 *MLSTM success interaction rate comparison with MDMTC and TMFM models*

security probes and an initial bias that the given entity under investigation may be malicious.

4.1 MLSTM and cloud deployment for critical sector applications

Currently, security in most cloud computing platforms is limited to the ability to partition/group instances by owners or hierarchical levels of administration. They do not include or enable support for TC integrity measurements/verifications based on TPM, an industry standard for computing hardware integrity measurement and testing, that depends on special (already available) hardware storage of cryptographic keys. Cloud platforms also do not include adequate ID mechanisms for certifying that the instances/virtual machines have not been tampered with.

Mission critical applications such as in electrical power systems can benefit from a trusted cloud computing deployment [2, 19], as the energy sector relies heavily on efficient and secure data. The energy sector requires a trusted cloud computing infrastructure that can guarantee secure ownership and integrity of the uploaded data even when it is decrypted for processing.

Wider spread adoption of a MLSTM can improve cloud-based deployment of applications for the mission critical sectors such as electrical power systems. For example, within a national electrical power system, several different entities responsible for the generation, transmission and distributions of electrical power may be required to share, exchange and process data from one another or with a central coordinating organisation such as a system operator [2, 19]. On a MLSTM enabled cloud, each entity colours their data before sharing on the cloud computing platform as this would guarantee the ability to detect unauthorised modifications and/or illegal use. More importantly, each entity can independently evaluate in an on-demand manner the instant security state of the entire cloud stack (IaaS, PaaS and SaaS) relative to their individual needs and applications.

5 Conclusion and future research

End-user concerns about trust, security and governance remain critical issues affecting the full adoption of cloud based services and platforms. Securing cloud computing platforms and services requires a holistic approach across all layers. This paper presented a multi-layer security trust model (MLSTM) that derives a unified cloud computing platform trust value from the fuzzy logic combination of states (measured on-demand using specific MLSTM transactions) of several different security mechanisms spread out across all cloud layers. The simulation based results show how the empirical output of the MLSTM may be used to classify a cloud platform/service and suggests that the model is capable of faster classification with a high degree of accuracy when compared to similar models such as the TMFM. The MLSTM is capable of improving end-user confidence and trust in cloud platforms and services in selecting or consuming cloud resources.

Future research will include evaluating the model's overhead and characteristics; additional comparisons with other existing cloud trust models and comparing/selecting amongst similar public cloud platforms/services based on the MLTSM output classification.

6 References

[1] Mell, P., Grance, T.: 'The NIST definition of cloud computing (draft) recommendations of the National Institute of Standards and Technology', *Nist Spec. Publ.*, 2011, **145**, (6), p. 7

[2] Sule, M.-J., Li, M., Taylor, G.A., *et al.*: 'Deploying trusted cloud computing for data intensive power system applications'. 2015 50th Int. Universities Power Engineering Conf. (UPEC), 2015, pp. 1–5

[3] Mell, P.: 'What's special about cloud security?', *IT Prof.*, 2012, **14**, (4), pp. 6–8

[4] Pearson, S., Benameur, A.: 'Privacy, security and trust issues arising from cloud computing'. 2010 IEEE Second Int. Conf. on Cloud Computing Technology and Science, 2010, pp. 693–702

[5] Ko, R.K.L., Jagadpramana, P., Mowbray, M., *et al.*: 'TrustCloud: a framework for accountability and trust in cloud computing'. 2011 IEEE World Congress on Services, 2011, pp. 584–588

[6] Huang, J., Nicol, D.M.: 'Trust mechanisms for cloud computing', *J. Cloud Comput. Adv. Syst. Appl.*, 2013, **2**, (1), p. 9

[7] Sun, D., Chang, G., Sun, L., *et al.*: 'A dynamic multi-dimensional trust evaluation model to enhance security of cloud computing environments', *Int. J. Innov. Comput. Appl.*, 2012, **4**, (3), pp. 200-212

[8] Canedo, E.D., Rafael Timóteo de Sousa, R.R.de.C., Junior, R.de.O.A.: 'Trust measurements yield distributed decision support in cloud computing', *Int. J. Cyber-Secur. Digit. Forensics*, 2012, **1**, (2), pp. 140–151

[9] Jansen, W., Grance, T.: 'Guidelines on security and privacy in public cloud computing', *Technical Report, NIST, Gaithersburg, MD, United States*, 2011, SP 800-144

[10] Shaikh, R., Sasikumar, M.: 'Trust model for measuring security strength of cloud computing service', *Procedia Comput. Sci.*, 2015, **45**, pp. 380–389

[11] Xu, W.: 'A fuzzy reputation-based trust management scheme for cloud computing', *Int. J. Digit. Content Technol. its Appl.*, 2012, **6**, (17), pp. 437–445

[12] Fan, W., Perros, H.: 'A novel trust management framework for multi-cloud environments based on trust service providers', *Knowledge-Based Syst.*, 2014, **70**, pp. 392–406

[13] Banirostam, H., Hedayati, A., Zadeh, A.K., *et al.*: 'A trust based approach for increasing security in cloud computing infrastructure'. 15th Int. Conf. on Computer Modelling Simulation, 2013

[14] Yang, K., Jia, X., Ren, K., *et al.*: 'DACS-MACS: effective data access control for multiauthority cloud storage systems', *IEEE Trans. Inf. Forensics Secur.*, 2013, **8**, (11), pp. 1790-1801

[15] Gu, L., Wang, C., Zhang, Y., *et al.*: 'Trust model in cloud computing environment based on fuzzy theory', *Int. J. Comput. Commun. Control*, 2014, **9**, (5), p. 570

[16] Mohsenzadeh, A., Motameni, H.: 'A trust model between cloud entities using fuzzy mathematics', *J. Intell. Fuzzy Syst.*, 2015, **29**, (5), pp. 1795–1803

[17] Prasath, V., Bharathan, N., Lakshmi, N.N.P.N., *et al.*: 'Fuzzy logic in cloud computing', *Int. J. Eng. Res. Technol.*, 2013, **2**, (3), pp. 1-5

[18] Chen, X., Chen, C., Tao, Y., *et al.*: 'A cloud security assessment system based on classifying and grading', *IEEE Cloud Comput.*, 2015, **2**, (2), pp. 58–67

[19] Wallom, D., Turilli, M., Martin, A., *et al.*: 'myTrustedCloud: trusted cloud infrastructure for security-critical computation and data managment'. Cloudcom, 2011 IEEE Third Int. Conf. on Cloud Computing Technology and Science, 2011, pp. 247–254

[20] SSH Communication Security: 'Why SSH Communication Security?'. 2016. Available at http://www.ssh.com/

[21] Hwang, K., Li, D.: 'Trusted cloud computing with secure resources and data coloring', *IEEE Internet Comput.*, 2010, **14**, (5), pp. 14–22

[22] Sivanandam, S.N., Sumathi, S., Deepa, S.N.: '*Introduction to fuzzy logic using MATLAB*' (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007)

[23] Zadeh, L.A.: 'The concept of a linguistic variable and its application to approximate reasoning', *J. Inf. Sci.*, 1975, **8**, pp. 199–249