International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 2 (2018) pp. 861-866
© Research India Publications. http://www.ripublication.com

# Two-factor Authenticated Key Agreement Protocol for Healthcare System

**Jungseok Lee**
*Department of Computer Science and Engineering, Kyunpook National University, Korea.*
*ORCID: 0000-0003-2490-3549*

**Kee Young Yoo**
*Department of Computer Science and Engineering, Kyunpook National University, Korea.*
*ORCID: 0000-0002-8484-0992*

**Abstract**

Cloud computing provides a services for sharing and storing the data. Both cloud server and trusted authority are semi-trusted party. The generation of trust value for cloud server and trust authority is a tedious process. Many researchers proposed different methods for generating trust value and it is still in open research. So, we propose a trust evaluation scheme for cloud data security using fuzzy based approach. By using fuzzy based approach, the trust value is generated. After obtaining the trust value, data can be re-encrypted by a trusted authority for security. After re-encryption the data are stored in cloud server which then can be shared and stored securely in cloud.

**Keywords:** Cloud Security, Trust Evaluation, Re-Encryption, Trust Authority, Fuzzy Control

## INTRODUCTION

Cloud computing is an emerging computing paradigm and a novel business model. It reduces cost and avoids the expenses by pay as you go model [1-2]. From data center [3] the cloud services are deployed and operated by the users. For cloud providers and cloud consumers [4, 8], cloud computing provides the speed for data sharing. By information and communications technology, pertinent data is accessed in a cloud environment. Instead of stored data, only on a local server/computer, remote hardware is used in cloud environment. At low cost, high increase in storage data is available in cloud environment. There are some risks concerning privacy and resilience in current cloud computing model. Lack of knowledge about cloud service selection for cloud consumer and provider is the main reason for risks in privacy. The trust based assumption in service selection satisfies the customer believes and it fulfills consumer's requirement. This is known as trustworthiness. Trust means an act of faith; confidence and reliance in something that's expected to behave or deliver as promised. Consumers get more confident and reliance for the care for their data when better trustworthiness is provided by the cloud service. It is the main consideration by the consumers. As discussed in [1], reputation degree is the main dependent factor of trustworthiness. So along with reputation we have to consider more trust factors while making decision in cloud service selection. For cloud service selection, there are enormous challenges. Thus there is no standardized metric system, it is difficult to compare trustworthiness of cloud services and also it is too expensive and time consuming to evaluate trustworthiness of cloud services.

The main objective of the paper is to create a trust model framework for data sharing and storage by trust evaluation service authority. The trust value is generated for the cloud servers and trusted authority. The parameter value of cloud server and trusted authority is collected from the cloud benchmark service. The rule generator computes the trust value based on the history and sends the it to the cloud user. The cloud user will select the cloud server based on the requirement.

The rest of this paper is organized as follows. We review the trust model framework in section 2. In section 3, we propose a trust evaluation scheme using fuzzy based approach. In section 4 we discuss the experimental analysis. Finally we conclude this paper in section 5.

## RELATED WORKS

Fan et al. in [16] proposed a novel trust management framework for multi-cloud environments based on trust service providers (TSPs). In the paper, the problems of trust management in multi-cloud environment are addressed based on a set of distributed $TSP_s$. The trust related services are provided to cloud participants by cloud providers, cloud service providers, cloud users. In cloud the $TSP_s$ are distributed and bring out the evidence of trust from different sources. By using this information the objective of trust is evaluated. Through a trust propagation network the TSP communicates and obtains trust information about cloud service provider. This experiment is effective and stable in differentiating trustworthy and untrustworthy. Habib et al. in [11] proposed a trust as a facilitator in cloud computing. They analyzed the existing trust evaluation schemes and characterizing the individual strength and weakness. Noor et al. in [7] proposed a trust management of services in cloud environments. They presented the synopsis of the cloud service models. The survey is on trust management service and issues in cloud.

Fan et al. in [17] proposed a novel two-stage model for cloud service trustworthiness evaluation. They addressed the trust evaluation problem in cloud service and proposed a novel evaluation model based on the evidential reasoning approach and fuzzy gap measurement to provide trust values. From the evaluation values three gaps are generated that evaluate the final values in formative and decision making. Pawar et al. in

[9] proposed a trust model for optimized cloud services. They proposed an uncertainty model. The reputation of cloud service provider is calculated using logical operators and computed opinion values. The proposed model is compared with the existing reputation models. Huang et al. in [5] proposed a trust mechanism for cloud computing. Trust establishment for existing mechanism and limitations are surveyed. They integrated the framework for combining various trust mechanisms.

Wang et al. in [6] proposed a trustworthiness evaluation framework in cloud computing for service selection. They presented a framework to rank and measure the trustworthiness of cloud services. The existent trustworthiness record is calculated by using the true datasets. The trustworthiness measurement framework is done in many experiments which are inaccurate and not flexible. Shaikh et al. in [10] proposed a trust framework for calculating security strength of a cloud service. It focuses on trust based solutions to achieve security in cloud. The framework calculates the trust value. Based on the trust value the cloud service can be selected. Wang et al. in [12] proposed a dynamic trust evaluation and scheduling framework for cloud computing. They proposed a trust mechanism based on task scheduling model by Bayesian cognitive method in which the trustworthiness of nodes are evaluated. The trustworthiness of node is integrated into dynamic level scheduling algorithm and then the trust dynamic level scheduling algorithm is proposed.

## TRUST EVALUATION SCHEME USING FUZZY BASED APPROACH

### *A. Architecture*

The general architecture of our proposed system is shown in Fig. 1, which involves three layers. The *first layer* is the cloud service consumer layer which includes *n* number of users in the clusters; clustering is not in part of our research and it is done based on the any of existing clustering algorithms [19]. The *second layer* is the trust management service which includes the web interface and trust evaluation service authority. Finally the *third layer* is the cloud service provider layer which includes cloud server and trusted authority. The steps involved in our system is given below;

1. Cloud user (*CU*) requests to request manager
2. Request manager (*RM*) verifies the request
3. Forwards the request to trust evaluation service authority (*TESA*)
4. Retrieves the benchmark results
5. Collects cloud server (*CS*) and trusted authority (*TA*) results
6. Submit the final result to *CU*

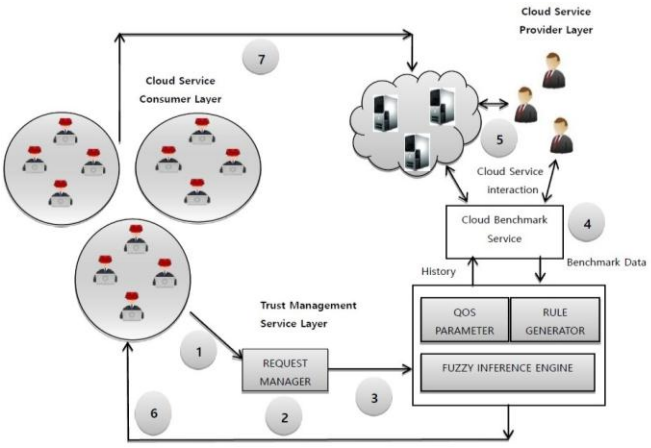


**Figure 1.** System architecture

### *B. Cloud service consumer layer*

In this layer the *CU* selects *CS* and *TA* based on the trust value generated from the cloud bench mark service. *CU* submits the request to web interface for trust value evaluation of different server in cloud. After evaluating the trust value, *TESA* will submit the results to *CU*. *CU* will deploy the service from *CS*.

**[Pseudo code for *CU*]**

```
Input: Request
Output: Trust value
  Broadcast SREQ to RM
  While (true)
  { Verifies certificates of CU
    If (CU non_valid)
      Discard req
    Else
      CU_i ← TV
    End if
  }
```

### *C. Trust Management service layer*

In this layer *CU* sends a request to *RM*. *RM* verifies the request of *CU*. *CU* must be from the appropriate group. *RM* verifies whether *CU* is from the authorized group or not. After verifying *CU*, *RM* forwards the request to *TESA*.

**[Pseudo code for *RM*]**

```
Input: CU_iReq
Output: Forward CU_iReq to TESA
  RM receives the SREQ
  While (true)
  {
    Verifies certificates of CU
    If (CU non_valid)
      Discard  req
    Else
      CU_iReq → TESA
    End if
  }
```

### *D. Trust Evaluation service authority*

In this layer trust evaluation service contains QoS parameter[1], rule generator and the fuzzy inference engine. It takes the input from *RM* and output as the trust value which is collected from the cloud benchmark service. *TESA* will send the output to *CU* based upon the requirement.

**[Pseudo code for *TESA*]**

```
Input: History data
Output: Trust value
  TESA receives the SREQ
  RM_req → TESA
  While (true)
  {
    Performs
    QoS parameter
    Rule generator
    Fuzzy inference engine
  }
```

### *E. Fuzzy based approach*

The inference process usually involves five major steps as shown in Fig. 2.

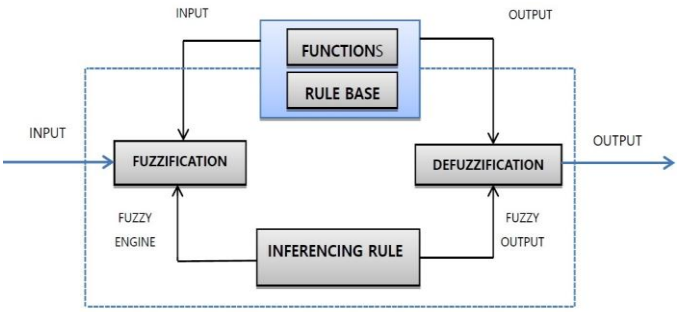


**Figure 2.** Fuzzy inference process

**Inference Engine:** Fuzzy logic operators and defuzzifier used in inference process is defined by inference engine.

**Membership Functions:** In what degree the fuzzy element belongs to the corresponding fuzzy set is defined by membership function. The values are mapped to membership degrees between the limit 0 and 1 by membership function. Own set of membership functions is given to each and every input and output variable by fuzzy inference system.

**Rulebase:** Inference model is defined by rulebase, which is a set of "If-Then" rules.

**Fuzzification:** To obtain corresponding membership degrees of each input variable, the input values are turned into membership functions regarding specific fuzzy set.

**Defuzzification:** Using defined defuzzification algorithm, aggregated fuzzy set is transformed into a value.

### *F. Cloud service provider layer*

In this layer trust value is generated by TESA and is sent to the intended user. After receiving the trust value the user sends the data to TA which has high trust value for re-encrypting the data. After re-encryption the user uploads data to the cloud server based on the trust value. The cloud user can also download the data from the cloud based on trust value.

**[Pseudo code for service deployment]**

```
Input: request
Output: message
  CU request to the cloud
  While (true)
  {
    Checks the trust value TV
    If (TV is high)
      Stores the message in cloud
    Else
      Drop data
    End if
  }
```

### *G. Requirements for calculating trust value*

**Numerical Requirements***:* For numerical requirements, the system calculates the QoS inputs of the $m_{th}$ service regarding the $n_{th}$.

**Range Requirements***:* For range requirements, the system calculates statistical indicators of the benchmark service traces[2] as the QoS inputs. We test our system with different indicators, i.e., very low, low, medium, high, and very high.

**Leaf Level Attributes Inference Module**

1. **Numerical:**
   **Rule 1:** If $P_i$ is good then $P_i$_trust is good
   **Rule 2:** If $P_i$ is bad then $P_i$_trust is bad.
2. **Range:**

$$T_{Good} = \begin{matrix} f(x) & x<threshold \\ 1 & x>threshold \end{matrix} \quad (1)$$

$$T_{Bad} = \begin{matrix} 1 & \text{if } x<threshold \\ f(x) & \text{if } x<threshold \end{matrix} \quad (2)$$

**Higher Level Attributes Inference Module**

**Rule 1:** If $P_o$_trust is Good and $P_i$ is Good then p= Good
**Rule 2:** If $P_o$_trust is Bad and $P_i$ is Bad then p= Bad

[1] *QoS parameters of cloud server and trusted authority are not specified in this document by considering the length*

[2] *Types and requirements are not included*

## RESULTS AND DISCUSSION

### *A. Experiemntal setup*

Our experiment is implemented on an Intel core (*TM*) i7 processor running at 1.5 GHZ, 8.00 GB of RAM, and SSD Serial ATA 3.0 Gbit/s drive with a 16MB buffer. The trust values are collected from cloud benchmark service. Standard dataset report (CloudHarmony [18]) for performance and compliance evaluation was investigated[3]. This report consists of the results of benchmarks on cloud servers and trusted authority. The proposed framework has been demonstrated by extracting a sample datasets from cloud harmony report. A simulation run of 6 CSPs for cloud servers and trusted authority has been performed. Trust is evaluated by applying the trust evaluation scheme.

### *B. Case study – Trust Evaluation for cloud servers based on compliance*

In this case study, *TESA* employs the trust evaluation scheme to generate trust on 18 $CSP_s$ based on 6 parameters from the cloud harmony a sample dataset. The cloud server includes Amazon Web Services (AWS), Google Cloud Platform, Microsoft Azure, Digital Ocean, Rackspace Cloud, and Soft Layer. The dataset consists of three types of cloud servers (small, medium, large) depends upon type of CPU cores, memory size and storage size. The sample dataset consists of performance of 18*CS*s on 8 parameters which are CPU speed, memory R/W, disk R/W, availability, time, startup time, shut down time, storage cost, data transfer cost. The trust value is generated on 18 $CS_s$ as shown in Table 1.

**Table 1.** Trust value generated for 18CS

| Cloud Servers | Trust |
|---|---|
| Amazon EC2(S) | 0.6501 |
| Digital Ocean(S) | 0.5355 |
| Google(S) | 0.7065 |
| Microsoft Azure(S) | 0.4333 |
| Rackspace(S) | 0.8577 |
| SoftLayer(S) | 0.403 |
| Amazon EC2(M) | 0.43 |
| Digital Ocean(M) | 0.5929 |
| Google(M) | 0.8779 |
| Microsoft Azure(M) | 0.2026 |
| Rackspace(M) | 0.8833 |
| SoftLayer(M) | 0.6154 |
| Amazon EC2(L) | 0.356 |
| Digital Ocean(L) | 0.5178 |
| Google(L) | 0.2562 |
| Microsoft Azure(L) | 0.126 |
| Rackspace(L) | 0.7737 |
| SoftLayer(L) | 0.5545 |

[3] *Compliance of 18 CSs and 5 TAs are not listed*

The trust value for CPU speed(0.0979) is high for Amazon EC2(L).The trust value for mem R/W(0.1416) is high for digital ocean(s).The trust value for disk R/W(0.3210) is high for Amazon EC2 (s).The trust value for availability(0.3014) is high for Rackspace(L).The trust value for startup time(0.1781) is high for Google(s).The trust value for shutdown time(0.2520) is high for Rackspace(M).The trust value for storage cost(0.1727) is high for Google(M). The trust value for data transfer cost (0.1902) is high for Digital ocean (L) as shown in Fig. 3.

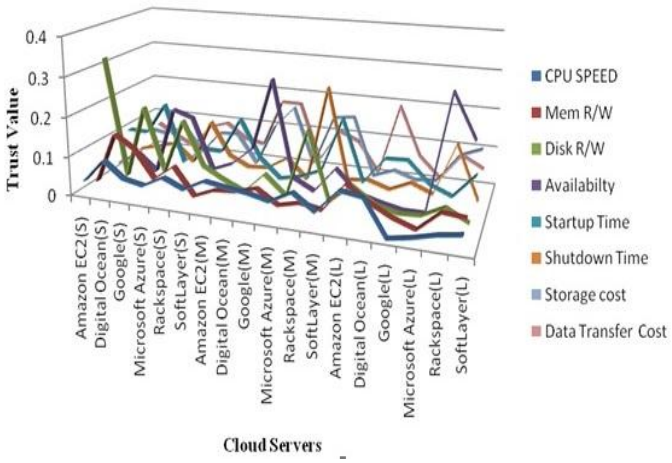


**Figure 3.** Compliance of the various parameters for cloud servers

### *C. Case study – Trust evaluation for trust authority based on compliance*

Our experiment in this case study, *TESA* generate trust on sample dataset extracted from the Cloud Harmony performance report on trusted authority. *TA*s covered in the report includes *TA*1, *TA*2, *TA*3, *TA*4, and *TA*5. The sample dataset consists of performance of 5*TA* on 3 benchmark parameters namely encryption management, performance and cost for encryption. The trust generated on $5TA_s$ is shown in Table 2.

**Table 2.** Trust value generated on 5TAs

| Trusted Authority | Trust |
|---|---|
| *TA*1 | 0.634 |
| *TA*2 | 0.921 |
| *TA*3 | 0.514 |
| *TA*4 | 0.416 |
| *TA*5 | 0.692 |

Trust generated on various cloud servers by our approach are shown in Fig.4. The trust value for Amazon EC2(S) is approximately (95.7%). The trust value for Digital Ocean (S) is approximately (92.6%).The trust value for Google (S) is approximately (96.4%). The trust for Microsoft Azure (S) is approximately (90.1%).The trust value generated for Rackspace (S) is approximately (98.3%).The trust value for

Soft layer(s) is approximately (90.1%).The trust value for Amazon EC2 (M) is approximately (91.7%).The trust value for Digital Ocean (M) is approximately (98.6%).The trust value for Google (M) is approximately (83.4%). The trust for Microsoft Azure (M) is approximately (90.1%).The trust value generated for Rackspace (M) is approximately (98.9%).The trust value of value of Soft layer (M) is approximately (96.1%).The trust value for Amazon EC2 (L) is approximately (91.7%).The trust value for Digital Ocean (L) is approximately (92.6%).The trust value for Google (L) is approximately (82.4%). The trust for Microsoft Azure (L) is approximately (75.1%).The trust value generated for Rackspace (L) is approximately (99.1%).The trust value of value of Soft layer (L) is approximately (96.1%).

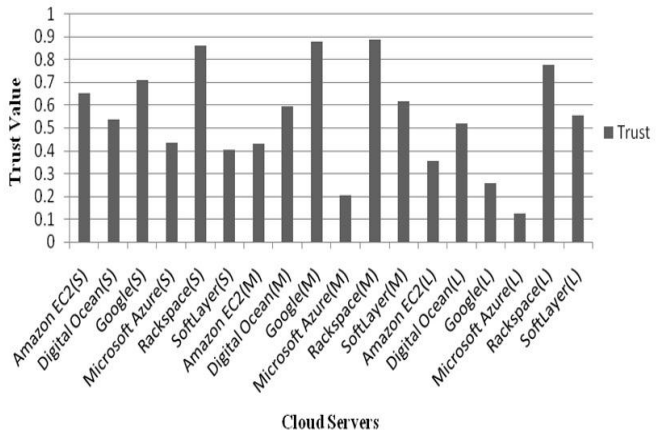


**Figure 4.** Trust evaluated for *CS* using fuzzy approach

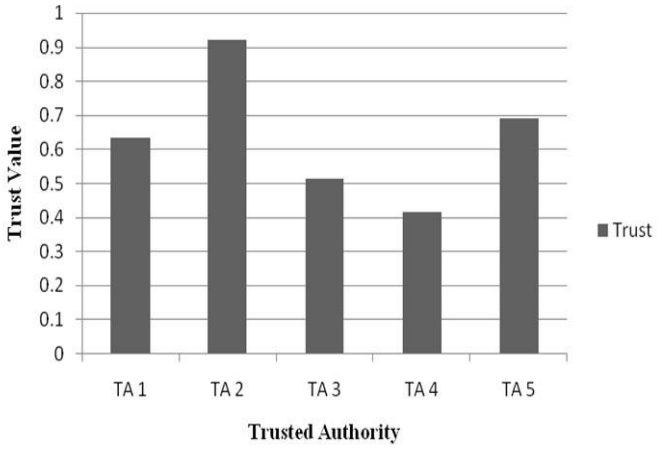


**Figure 5.** Trust evaluated for *TA* using fuzzy approach

Similarly trust generated by our approach on various trusted authority are also shown in Fig. 5. The trust value for *TA*1 is approximately (95.7%). The trust value for *TA*2 is approximately (98.6%).The trust value for *TA*3 approximately (92.4%). The trust value generated for *TA*4 is approximately (90.1%).The trust value generated for *TA*5 is approximately (97.3%).

## CONCLUSION

Trust value scheme evaluates the trust value for cloud server and trusted authority based on fuzzy based approach. The framework evaluates trustworthiness of *CS*s and *TA* quantitatively as a fraction between 0 and 1.The frameworks generates trust on *CS*s by evaluating the compliance of QoS parameters and then by utilizing the fuzzy based approach. As future work, we have planned to start research for the solution of trust revision problem, trust timeliness, storage and propagation of trust in the cloud environment.

## ACKNOWLEDGEMENTS

Corresponding author is Hyunsung Kim. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2017R1D1A1B04032598).

## REFERENCES

[1] Aldini, A., Gorrieri, R.: Foundations of Security Analysis and Design VI. Lecture Notes in Computer Science, 6858 (2011). doi: 10.1007/978-3-642-23082-0

[2] Buyya, R., Yeo, C., Venugopal, S., Boreberg, J., Brandic, I.: Cloud computing and emerging IT platforms: Vision, type, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 25, 599–616 (2009).

[3] Fan, W., Perros, H.: A novel trust management framework for multi-cloud environments based on trust service providers. Knowledge based systems, 70, 392-406 (2014). doi:10.1016/j.knosys.2014.07.018

[4] Foster, I., Zhao, Y., Raicu, I., Lu, S.: Cloud computing and grid computing 360-degree compared. In Grid Computing Environments Workshop, 1–10 (20080. doi: 10.1109/GCE.2008.4738445

[5] Huang, J., Nicol, D.M.: Trust mechanisms for cloud computing. Journal of Cloud Computing Advances, Systems and Applications 2(9) (2013). doi: 10.1186/2192-113X-2-9

[6] Wang, L., Wu, Z.: A Trustworthiness evaluation framework in cloud computing for service selection. In: IEEE 6th International Conf. on Cloud computing technology and science, Washington, DC, USA, 101-106 (2014). doi:10.1109/CloudCom.2014.107

[7] Noor, T.H., Sheng, Q.Z., Zeadally, S., Yu, J.: Trust management of services in cloud environments: Obstacles and solutions. ACM Computing Surveys, 46(1), 1-35 (2013). doi:10.1145/2522968.2522980

[8] Noor, T.H., Sheng Q.Z.: Trust as a Service: A Framework for Trust Management in Cloud Environments. In: Bouguettaya, A., Hauswirth, M., Liu, L.(eds,) WISE 2011. Lecture Notes in Computer Science, vol. 6997. pp. 314-321. Springer, Berlin, Heidelberg (2011), doi: 10.1007/978-3-642-24434-6_2

[9] Pawar, P.S., Rajarajan, M., Nair, S.K., Zisman, A.: Trust Model for Optimized Cloud Services. In:Dimitrakos, T., Moona, R., Patel, D., McKnight, D.H. 6th International Conference on Trust Management, AICT-374, pp. 97-112. Springer (2012), doi: 10.1007/978-3-642-29852-3_7

[10] Shaikh, R., Sasikumar, M.: Trust framework for calculating security strength of a cloud service. In: Proceedings of International conference on Communication, Information & Computing Technology, 703-709 (2012), doi: 10.1109/ICCICT.2012.6398163

[11] Habib, S.M., Hauke, S., Ries, S.: Trust as a facilitator in cloud computing – A survey. Journal of Cloud Computing: Advances, Systems and Applications, 1(19) (2012). doi:10.1186/2192-113X-1-19

[12] Wang, W., Zeng, G., Zhang, J., Tang, D.: Dynamic trust evaluation and scheduling framework for cloud computing. Security and communication networks. 5, 311–318 (2012). doi:10.1002/sec.350

[13] Li, X., Xie, H., Chen, L., Wang, J., Deng, X.: News impact on stock price return via sentiment analysis. Knowledge-Based Systems. 69, 14-23 (2014). doi:10.1016/j.knosys.2014.04.022

[14] Xie, H., Li, X., Wang, T., Chen, L., Li, K., Wang, F.L., Cai, Y., Li, Q., Min, H.: Personalized search for social media via dominating verbal context. Neurocomputing. 172, 27-37 (2016). doi:10.1016/j.neucom.2014.12.109

[15] Rao, Y., Li, Q., Wu, Q., Xie, H., Wang, F.L., Wang, T.: A multi-relational term scheme for first story detection. Neurocomputing (2017), doi:10.1016/j.neucom.2016.06.089

[16] Fan, W., Perros, H.: A novel trust management framework for multi-cloud environments based on trust service providers. Knowledge-Based Systems. 70, 392-406 (2014), doi:10.1016/j.knosys.2014.07.018

[17] Fan, W., Yang, S., Pei, J.: A novel two-stage model for cloud service trustworthiness evaluation. Expert Systems. 31, 136-153 (2014). doi:10.1111/exsy.12017

[18] Cloud Harmony Cloud Reports: Available from: https://cloudharmony.com/reports/state- of- the cloud-compute-report, last accessed 2015/2/5.

[19] Clustering algorithms: Available from: https://web.stanford.edu/ class/ cs345a/ slides/12 clustering.pdf, last accessed 2015/2/5.