CHAPTER 7

Trustworthy Cloud Computing

Olasunkanmi Matthew Alofe and Kaniz Fatema

Abstract Trustworthy cloud computing has been a central tenet of the European Union cloud strategy for nearly a decade. This chapter discusses the origins of trustworthy computing and specifically how the goals of trustworthy computing—security and privacy, reliability, and business integrity—are represented in computer science research. We call for further inter- and multi-disciplinary research on trustworthy cloud computing that reflect a more holistic view of trust.

Keywords Trustworthy computing • Cloud computing • Trust • Reliability • Security • Business integrity

O. M. Alofe (✉)
University of Derby, Derby, UK
e-mail: o.alofe1@unimail.derby.ac.uk

K. Fatema
Department of Computer Science, Aston University, Birmingham, UK

T. Lynn et al. (eds.), *Data Privacy and Trust in Cloud Computing*,
Palgrave Studies in Digital Business & Enabling Technologies,
https://doi.org/10.1007/978-3-030-54660-1_7

7.1 Introduction

In 2002, Bill Gates, in an email to Microsoft employees, presaged a future where computing would be "an integral and indispensable part of almost everything we do" (Gates 2002). Subsequently, Microsoft published a white paper defining what would ultimately become a seminal white paper for trustworthy computing. Recognising that trust is a complex concept, Mundie et al. (2002) explored trustworthy computing from three perspectives—the user's perspective (goals), the mechanisms employed by industry to meet the goals (means), and the way in which an organisation conducts its operations to deliver the components (execution). The key definitions of goals, means and execution are summarised in Table 7.1 below. While in 2002, cloud computing was not the dominant computing paradigm it is today, these perspectives reflect the dominant themes in computer science research on trust in cloud computing. Indeed they are reflective of the wider scholarly debate discussed throughout this book.

Improving the confidence and perception of trustworthiness is critical for the adoption of cloud computing, and has been a central tenet of the European Union cloud strategy for nearly a decade (European Commission 2020). The remainder of this chapter provides a brief overview of computer science research based on the goals of trustworthy computing identified above, namely security and privacy, reliability, and business integrity.

7.2 Security and Privacy

According to the National Information Systems Security Glossary, information security is the protection of information systems against unauthorised access to and modification of information and data in various forms such as data at rest, and in transit (Hayden 2000). Information security applies to the safeguarding of data in its various states and storage locations, as well as offering protection against attacks such as denial-of-service (DoS), which might adversely impact the confidentiality, integrity, and availability of information to authorised users. As discussed in Chap. 1, integrity is a key element in trust, and in the context of cloud computing, the maintenance of confidentiality and continuity of service availability are key signals of competence. As such, from a computer science perspective, designing attack-resilient systems is critical to building and maintaining trust. Different frameworks and models have been proposed and designed for the establishment of trust within cloud computing that offer system

Table 7.1 Definition of goals, means, and execution in trustworthy computing

Goals	**The basis for a customer's decision to trust a system**
Security & privacy	The expectation of attack-resilient systems and that the confidentiality, integrity, and availability of the system and its data are protected. The customer can control data about themselves, and those using such data adhere to fair information principles
Reliability	The customer can depend on the product to fulfil its functions when required to do so.
Business integrity	The vendor of a product behaves in a responsive and responsible manner.
Means	**The business and engineering considerations that enable a system supplier to deliver on the Goals**
Secure by design, secure by default, secure in deployment	A process is in place to protect the confidentiality, integrity, and availability of data and systems at every phase of the software development process.
Fair information principles	The collection and sharing of end-user data requires the consent of the end user, and privacy is respected, and data is only used in line with Fair Information Practices.
Availability	The system is available for use as required.
Manageability	The system is easy to install and manage, relative to its size and complexity.
Accuracy	The system performs its functions correctly. Data is protected from corruption and loss.
Usability	The software is easy to use and suitable to the user's needs.
Responsiveness	The vendor accepts responsibility for problems, and takes action to correct them. Support is available to customers as needed throughout their engagement with vendor.
Transparency	The vendor is open in its dealings with customers. Its motives are clear, it keeps its word, and customers know where they stand in a transaction or interaction with the vendor.
Execution	**The way an organisation conducts its operations to deliver the components required for trustworthy computing**
Intents	• Company policies, directives, benchmarks, and guidelines • Contracts and undertakings with customers, including Service Level Agreements (SLAs) • Corporate, industry and regulatory standards Government legislation, policies, and regulations

(*continued*)

Table 7.1 (continued)

Implementation	• Risk analysis • Development practices, including architecture, coding, documentation, and testing • Training and education • Terms of business • Marketing and sales practices • Operations practices, including deployment, maintenance, sales & support, and risk management • Enforcement of intents and dispute resolution
Evidence	• Self-assessment • Accreditation by third parties • External audit

Adapted from Mundie et al. (2002)

security and data privacy for cloud service providers and their customers. Five common approaches for protecting cloud systems and data in extant literature include multi-cloud storage, homomorphic encryption schemes, secure sharing systems, deployment of intermediary components, as well as more traditional security and privacy methods.

Multi-cloud storage strategies seek to reduce security and availability risks by diversifying this risk through the use of multiple cloud storage service providers (Bucur et al. 2018). For example, Alqahtani and Kouadri-Mostefaou (2014) propose a framework that ensures the security of mobile cloud computing by deploying distributed multi-cloud storage, data encryption, and data compression techniques. The framework operates by dividing the data into different segments at the user end based on the preference selected by the user before the encryption and compression of the segments. The compressed segments are stored on distributed multi-cloud storage service providers. Similarly, Abdalla and Pathan (2014) presented a framework using a data protection manager (DPM) deployed for the transmission of data to the cloud service provider. The DPM both fragments and merges the data in the proposed framework. First, it breaks the data into fragments and transmits them to the multi-cloud for storage. When a user requests the data, the DPM merges the data. The service provider maps the information of fragmented and merged data to the individual users and the multi-cloud technique applied protects data on other segments if one segment is compromised. While multi-cloud storage in theory has many advantageous attributes, in practice, it has significant

limitations, not least the lack of standards-based interoperable clouds and APIs, the possible amplification of the attack surface to multiple clouds, and the management and measurement of multiple service level agreements across multiple clouds (Bucur et al. 2018).

There is a long history of encryption as a means of securing systems. For example, many messaging systems use encryption to protect the content of messages through the use of shared public or private keys. These legacy systems have a number of limitations including data control and the management of keys (Acar et al. 2018). Homomorphic encryption schemes overcome these limitations by allowing a cloud service provider to perform certain computable functions on the encrypted data while preserving the features of the function and format of the encrypted data (Acar et al. 2018). Louk and Lim (2015) proposed a homomorphic data security encryption scheme that converted data into ciphertext and manipulated the ciphertext just like the original text without compromising the encryption. There are a variety of different homographic encryption types, for example multiplicative, additive and fully homomorphic, all of which have been applied to secure communication and storage in the cloud (Tebaa and Hajji 2014). There are significant performance limitations with fully homomorphic encryption schemes thus requiring optimisation at the architectural, algorithmic, and hardware resource levels (Moore et al. 2014).

The ubiquity of smartphones, and their dependence on cloud computing, present significant challenges for securing data at the edge, in the cloud, and in between. Smartphones, and indeed other Internet of Things end points, are typically resource constrained due to their form and bandwidth. As such, security methods need to be relatively lightweight. Wang et al. (2014) propose a secure sharing scheme that envisages users uploading multiple data pieces to different clouds, and using a watermarking algorithm for authentication of mobile users and cloud services. A key feature of this solution is the both the security and the reduced load on the network. Khan et al. (2014) propose a BSS (block-based sharing scheme) cryptographic method that divides data logically into multiple blocks, encrypting and decrypting the blocks, and reconstructing the data into their original form. Secure Data Sharing in Clouds (SeDaSC) is another approach to secure sharing comprising three entities—the user, a cryptographic server (CS) and the cloud (Ali et al. 2015). The CS is responsible for encryption, decryption, key management, and access control. Yu et al. (2015) proposed a public auditing protocol that ensures the integrity of

data stored in the cloud and shared data among users by using the asymmetric group key agreement scheme and proxy re-signature. The asymmetric group key agreement scheme allows the group to share both public and private keys and create a tag attached to files. The proxy re-signature updates the tags when there are changes in the group members. User identity information is preserved by anonymising the auditor and group members. In this way, data control is improved, in instances such as when employees leave an organisation.

Similar to the auditing scheme proposed by Yu et al. (2015), a number of works have proposed auditing schemes where, in effect, an independent third party serves as the verifier of data integrity. For example, Sookhak et al (2014) proposed a remote data auditing method for verifying the integrity of data stored in cloud; algebraic signatures are used to allow the auditor to check the possession of user data in cloud. Similarly, Yu et al. (2016) propose key-updating and authenticator-evolving mechanism with zero-knowledge privacy of the stored files for secure cloud data auditing, which incorporates zero-knowledge proof systems, proxy re-signatures and homomorphic linear authenticators. Yang et al. (2015) proposed an extended proxy-assisted approach that utilises an attribute-based encryption method to ensure scalable data sharing within the cloud. Tian et al. (2015) proposed a dynamic hash table (DHT) public auditing scheme. The DHT is a two-dimensional data structure used by the auditor to record data property information for rapid and dynamic auditing. Public key-based homomorphic authentication and random masking created by the auditor are used for the preservation of privacy.

While each of the approaches above represent novel means to securing data, the practical reality is that most cloud service providers rely on traditional security and privacy methodologies. A wide range of approaches have been proposed for securing cloud services including securing infrastructure using extant multi-component methods. For example, Liu et al. (2015) propose a secure infrastructure based on Advanced Encryption Standard (AES), Searchable Symmetric Encryption (SSE), Ciphertext-Policy Attribute-Based Encryption (CPABE) and Digital Signature (DS). Mollah et al. (2017) propose a scheme that utilises a combination of secret key encryption, public key encryption, searchable secret key encryption and digital signatures for a data searching and sharing scheme. The STOVE model proposed by Tan et al. (2014) secures data in the cloud by restricting the operational ability of applications. The model restricts untrusted applications and isolates the application using formal

verification methods to verify the isolated code; application execution is performed in isolation and under strict observation. The novelty of these methods, and many others, is in the combination of multiple approaches. However, the challenge for industry and researchers alike is identifying the most feasible candidates for a given use case.

7.3 Reliability

It is essential that services and data in the cloud are available to users at all times. As discussed in Chap. 2, availability is defined in the service level agreements between cloud service providers and their customers. The most commonly used definition of reliability in engineering applications according to Dummer et al. (1997, p. 79) is "the characteristic of an item expressed by the probability that it will perform a required function under stated conditions for a stated period of time." In general terms, service reliability can be represented as:

$$\text{Service Reliability} = \frac{\left(\text{Successful Responses}\right)}{\text{Total Requests}} \times 100\%.$$

While such a calculation may indicate service reliability, in hyperscale multi-tenant clouds the overall cloud may be reliable but specific services may be unreliable. Due to the scale of the clouds, one particular service failure or underperforming component may not impact an overall reliability score, while at the same time result in catastrophic failure. Huang et al. (2017) suggest that major cloud failures often result from subtle underlying faults in systems, so-called 'gray failures', that may be difficult to observe or even detect. They are characterised by this differential observability (Huang et al. 2017).

When ascertaining that a system will perform a specific function within a given cloud service environment, Adams et al. (2014) suggest the following key considerations:

- Service availability must be maximised to ensure users can access the service and perform their required task to completion without interference;

- The impact of system failure should be minimised for individual users, the overall number of users affected, and the downtime associated for the failure;
- Service performance and capacity should be maximised to reduce the impact of reduced performance even if no failure is detected; and,
- Business continuity should be maximised by responding to failures when they occur, protecting the integrity of data, and recovering as soon as possible.

Reliability and high availability are closely related and regarded as significant challenges in cloud computing. Obviously, cloud service providers and scholars invest a significant amount of effort in to the design of fault-tolerant, attack-resilient and reliable systems. A detailed discussion of this is beyond the scope of this chapter. These innovations are often opaque to the user. As such, we provide a high-level overview of approaches to reliability including ensuring reliability by design through monitoring, redundancy and disaster recovery, and the evaluation of performance and quality of service (QoS).

A major focus of computer science research is reliability by design so that no one point of failure can result in the failure of the entire system. There are a wide variety of causes of unplanned cloud outages including infrastructure or software failures, planning mistakes, human error, or external attacks (Endo et al. 2017). Three main strategies are employed to counter such failures namely, monitoring, redundancy, and disaster recovery. In the terminology of trust, two could be classified as trust-building mechanisms (monitoring and redundancy) while the third, disaster recovery, could be classified as a trust repair mechanism. A wide variety of general purpose and vendor-specific monitoring tools are used in cloud computing. From the user perspective, these are primarily used for accounting and billing, security and privacy assurance, and SLA management, while for the cloud service provider they may be used for other reliability functions, for example fault management (Fatema et al. 2014). As mentioned earlier, gray failures may not be detectable by extant monitoring systems that focus on singular failure detection. To mitigate the risk of such failures, Huang et al. (2017) suggest that cloud service providers must move to multi-dimensional cloud health monitoring. While accepting monitoring all applications and workloads in hyperscale multi-tenant systems is not feasible, they propose a number of techniques to close the observation gap including approximating application views, aggregating

observations from multiple components to infer the likelihood of a gray failure in an isolated component, as well as temporal analysis (Huang et al. 2017). As noted briefly in Chap. 1, monitoring data can be used more widely in the context of building knowledge-based trust. Emeakaroha et al. (2016) have proposed a system and show through experimental studies with business decision-makers that such monitoring systems can be used to build trust through communication strategies such as trust labels (Emeakaroha et al. 2016; van der Werff et al. 2018).

Cloud failures can be caused by issues that occur at different levels in the cloud stack e.g. at the data, application, and/or system level (Huang et al. 2017). Given organisational and consumer concerns about data and availability of data in the event of a failure, it is unsurprising that in addition to general system redundancy, data redundancy is a primary concern of cloud service providers. Data replication and erasure coding are commonly used data redundancy techniques in cloud computing (Nachiappan et al. 2017). With simple data replication, data is replicated in at least two locations on distributed cloud storage systems so that in the event of storage failure, it is just served from the replicated copy (Plank 2013). As such, data loss only occurs if data corrupted on all storage targets the replicated copies (Rajaasekharan 2014). As simple data replication carries a significant resource overhead in terms of storage, network and associated energy consumption, hyperscale cloud service providers, such as Facebook and Microsoft, use more advanced erasure coding, such as K out of N codes, to detect and correct errors in cloud storage, and provide a less resource intensive means to reconstruct data from parity data (Nachiappan et al. 2017; Rajaasekharan 2014).

Disasters differ in terms of scale and impact (although this is subjective), and are typically unpredicted events that occur relatively rarely over the lifetime of a given system. A full cloud service outage occurs more frequently than one might imagine but due to the disaster recovery systems in place, the recovery time is extremely fast. Disasters can result from natural, human, or technological causes, or a combination of two or more of these (Singh et al. 2016). To mitigate the impact of natural disasters or large-scale malicious physical attacks, cloud service providers, like many IT organisations, use distributed backups, online and offline, in geographic locations that are located sufficiently distant to avoid a homogenous natural event (Pokharel et al. 2010). Maintaining two infrastructures is extremely costly. However, cloud outages can also result from relatively small-scale localised natural causes, for example lightning strikes are a

significant threat to both primary and uninterruptible power supply (Li et al. 2013). Human causes include human error or malicious attacks from insiders or external third parties. The latter is largely a security issue while the former is a training and behavioural one. Li et al. (2013) document a wide range of public cloud outages resulting from human error including vehicle accidents, power shutdowns, and inputting commands in error. As discussed earlier in this section, application and system level failures can be technological causes of full service outage. In these instances, for application failures, the key requirement is business continuity through redundancy and rollback. It should be noted that a number of middleware approaches have been applied to address application-level reliability via application-independent failure detection, checkpoint and rollback and recovery (e.g. Hormati, et al. 2014), optimal replica placement (e.g. An et al. 2014), stop and copy VM migration (Sampaio and Barbosa 2018), and entity reputation management (Abawajy 2011). For system level failures, the primary focus is minimising recovery time (Singh et al. 2016). It is important to note that while these causes are isolated, they may be cascading, natural causes can result in unanticipated technological failures, which in turn may be exacerbated by human errors, and so forth.

As discussed in Chap. 2, the SLA details the level of service to be provided, often in the form of specific QoS metrics (Ghazizadeh and Cusack 2018). Obviously, in the context of trust, there is a close relationship between SLA metrics and monitoring, and unsurprisingly this is a major focus of both cloud monitoring systems (see Fatema et al. 2014) and trustworthy cloud computing research. This research primarily focuses on the decomposition of SLA parameters in to low-level system performance metrics, mapping these in to KPIs, and then ultimately aggregating these KPIs in to some form of aggregated quality indicator that can be used to mitigate transactional risk (Sun et al. 2012). A wide range of techniques are used to measure and predict cloud service performance (and indeed SLA violation). Typical metrics include availability, bandwidth, cost (including energy), CPU cycle, service duration, memory, request arrival rate, space/storage. Upgrade request frequency as well as other more specific performance metrics (throughput, response time, execution time etc.) are also present, although the importance of these will vary by cloud service (Faniyi and Bahsoon 2015). Cloud service providers may also include metrics that specifically acknowledge the risk of failure e.g. the maximum fraction of SLA violations allowed or penalty rates (Faniyi and Bahsoon 2015). Notably, security is an attribute metric that is extremely

difficult to measure and is typically based on a qualitative evaluation of cloud service provider policies and system features (Shaikh and Sasikumar 2015). Once such metrics have been extracted from the system, they can be shared with consumers to build trust or select cloud service providers. An example of the former is the cloud trust label mentioned earlier (Emeakaroha et al. 2016; van der Werff et al. 2018). Regarding the latter, Garg, et al. (2013) propose a Service Measurement Index Cloud (SMICloud) framework for assisting consumers to identify the most suitable cloud service provider to contract with. The SMICloud reviews Quality of Service (QoS) requirements and ranks services based on previous user experiences and performance of services based on KPIs such as those previously mentioned. As a final note on cloud performance metrics, the determination of the intervals for this data is an essential and somewhat open challenge. This includes the monitoring intervals between the collection of low-level metrics and the intervals between the aggregate KPIs or high-level quality indicators (Sun et al. 2012). A balance between intrusiveness and utility is required to avoid adverse impacts on system performance while ensuring the availability of sufficiently time-sensitive data to assure accurate SLA measurement (Sun et al. 2012).

7.4 Business Integrity

As discussed in Chap. 1, the trust literature views integrity generally as one party's perception that another party will adhere to a set of acceptable principles, act honestly, and fulfil their promises (Mayer et al. 1995; McKnight et al. 2011). This is consistent with the principles laid out by Microsoft in Mundie et al. (2002), namely that a vendor, in this case a cloud service provider, will behave in a responsive and responsible manner. While Mundie et al. (2002) exemplify this behaviour in terms of responsiveness to problems that may arise, others expand this, in a technological context, to mean that both the service and vendor behave predictably to the extent which it is possible to anticipate the system and the service provider's behaviour accurately (van der Werff et al. 2018). In one sense, it is no surprise that computer scientists have found it difficult to distinguish reliability, as an attribute, from integrity.

In computer science literature, integrity is more commonly found as an attribute of data and underlying systems rather than the service as a whole or the vendor. This is not to say that computer science researchers have not explored technological innovations in this regard. In addition to

attempts to communicate performance metrics and service measurement mechanisms similar to those outlined in Sect. 7.3 above, some researchers have focussed on more holistic evaluations of cloud services and service providers. As referenced briefly in Chap. 1, feedback systems and reputation management systems are two approaches explored in research to build trust. For example, Baranwal and Vidyarthi (2014) propose a Service Measurement Index (SMI) comprising two sets of metrics—application-dependent metrics and user-dependent metrics. Notably, in the context of Mundie et al. (2002), they include customer support as an application-dependent metric. Unlike the SLA-focused measurements discussed earlier, SMI includes reputation metrics based on feedback from users, user experience and certification of compliance with industry best practice and regulations. In a similar vein, Machhi and Jethava (2016) present a trust management framework that measures service provider trustworthiness based on feedback, aging factor, and other parameters, while eliminating or otherwise discounting unreliable feedback. Indeed a number of works have sought to combine SLA metrics with feedback systems as a means of communicating trust in the service and vendor (see, for example, Nguyen et al. 2010; Habib et al. 2011; Yau and Yin 2011; Garg et al. 2013; Noor, et al. 2015; Tang et al. 2017).

While these researchers have sought to explore integrity as a quantifiable attribute of a service, business integrity is typically either conflated as competence (see for example Chakraborty and Roy 2012), or as a function of information assurance practices and qualitative audits such as certification (Chakraborty et al. 2010).

7.5 Conclusion

This chapter presented a discussion on trustworthy computing from three perspectives—security and privacy, reliability, and business integrity. Computer science research has typically sought to focus on trust as an objective attribute of systems, and on occasion cloud service providers, that can be ultimately measured, compared and benchmarked. One might argue that it is a narrow view of trust that misses the more nuanced aspects of the psychological underpinnings of trust. This may go some way to explaining why trust remains a significant barrier to cloud computing adoption. As a starting point, researchers might consider using the taxonomy of trustworthy computing laid out by Microsoft in Mundie et al. (2002), i.e. goals, means and execution, to identify gaps in the literature and state of the art, and guide future avenues for research. As we move

towards the Internet of Things, and greater use of advanced autonomous technologies, such as self-learning, self-management, and artificial intelligence, a more inter- and multi-disciplinary approach is needed to ensure that all stakeholders benefit fully and fairly from these transformative technologies.

References

Abawajy, J. (2011). *Establishing Trust in Hybrid Cloud Computing Environments.* 2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (pp. 118–125). IEEE.

Abdalla, A. K. A., & Pathan, A. S. K. (2014). On Protecting Data Storage in Mobile Cloud Computing Paradigm. *IETE Technical Review, 31*(1), 82–91.

Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A Survey on Homomorphic Encryption Schemes: Theory and Implementation. *ACM Computing Surveys (CSUR), 51*(4), 1–35.

Adams, M., Bearly, S., Bills, D., Foy, S., Li, M., Rains, T., Ray, M., Rogers, D., Simorjay, F., Suthers, S., & Wescott, J. (2014). *An Introduction to Designing Reliable Cloud Services.* Redmond, WA: Microsoft Corporation.

Ali, M., Dhamotharan, R., Khan, E., Khan, S. U., Vasilakos, A. V., Li, K., & Zomaya, A. Y. (2015). SeDaSC: Secure Data Sharing in Clouds. *IEEE Systems Journal, 11*(2), 395–404.

Alqahtani, H. S., & Kouadri-Mostefaou, G. (2014). *Multi-clouds Mobile Computing for the Secure Storage of Data.* 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing (pp. 495–496). IEEE.

An, K., Shekhar, S., Caglar, F., Gokhale, A., & Sastry, S. (2014). A Cloud Middleware for Assuring Performance and High Availability of Soft Real-Time Applications. *Journal of Systems Architecture, 60*(9), 757–769.

Baranwal, G., & Vidyarthi, D. P. (2014). *A Framework for Selection of Best Cloud Service Provider Using Ranked Voting Method.* 2014 IEEE International Advance Computing Conference (IACC) (pp. 831–837). IEEE.

Bucur, V., Dehelean, C., & Miclea, L. (2018). *Object Storage in the Cloud and Multi-cloud: State of the Art and the Research Challenges.* 2018 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR) (pp. 1–6). IEEE.

Chakraborty, R., Ramireddy, S., Raghu, T. S., & Rao, H. R. (2010). The Information Assurance Practices of Cloud Computing Vendors. *IT Professional, 12*(4), 29–37.

Chakraborty, S., & Roy, K. (2012). *An SLA-based Framework for Estimating Trustworthiness of a Cloud.* 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (pp. 937–942). IEEE.

Dummer, G. W. A., Winton, R., & Tooley, M. (1997). *An Elementary Guide to Reliability*. Elsevier.

Emeakaroha, V. C., Fatema, K., van der Werff, L., Healy, P., Lynn, T., & Morrison, J. P. (2016). A Trust Label System for Communicating Trust in Cloud Services. *IEEE Transactions on Services Computing, 10*(5), 689–700.

Endo, P. T., Santos, G. L., Rosendo, D., Gomes, D. M., Moreira, A., Kelner, J., Sadok, D., Gonclaves, G. E., & Mahloo, M. (2017). Minimizing and Managing Cloud Failures. *Computer, 50*(11), 86–90.

European Commission. (2020). Shaping Europe's Digital Future. Retrieved June 6, 2020, from https://ec.europa.eu/info/sites/info/files/communication-shaping-europes-digital-future-feb2020_en_4.pdf

Faniyi, F., & Bahsoon, R. (2015). A Systematic Review of Service Level Management in the Cloud. *ACM Computing Surveys (CSUR), 48*(3), 1–27.

Fatema, K., Emeakaroha, V. C., Healy, P. D., Morrison, J. P., & Lynn, T. (2014). A Survey of Cloud Monitoring Tools: Taxonomy, Capabilities and Objectives. *Journal of Parallel and Distributed Computing, 74*(10), 2918–2933.

Garg, S. K., Versteeg, S., & Buyya, R. (2013). A Framework for Ranking of Cloud Computing Services. *Future Generation Computer Systems, 29*(4), 1012–1023.

Gates, B. (2002). Trustworthy Computing. Retrieved June 6, 2020, from https://www.wired.com/2002/01/bill-gates-trustworthy-computing/

Ghazizadeh, E., & Cusack, B. (2018). Evaluation Theory for Characteristics of Cloud Identity Trust Framework. In *Cloud Computing-Technology and Practices*. IntechOpen.

Habib, S. M., Ries, S., & Muhlhauser, M. (2011). *Towards a Trust Management System for Cloud Computing*. 2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (pp. 933–939). IEEE.

Hayden, M. V. (2000). *National Information Systems Security Glossary*. National Security Telecommunications and Information Systems Security Committee.

Hormati, M., Khendek, F., & Toeroe, M. (2014). *Towards an Evaluation Framework for Availability Solutions in the Cloud*. 2014 IEEE International Symposium on Software Reliability Engineering Workshops (pp. 43–46). IEEE.

Huang, P., Guo, C., Zhou, L., Lorch, J. R., Dang, Y., Chintalapati, M., & Yao, R. (2017). *Gray Failure: The Achilles' Heel of Cloud-Scale Systems*. Proceedings of the 16th Workshop on Hot Topics in Operating Systems (pp. 150–155).

Khan, A. N., Kiah, M. M., Ali, M., Madani, S. A., & Shamshirband, S. (2014). BSS: Block-based Sharing Scheme for Secure Data Storage Services in Mobile Cloud Environment. *The Journal of Supercomputing, 70*(2), 946–976.

Li, Z., Liang, M., O'Brien, L., & Zhang, H. (2013). The Cloud's Cloudy Moment: A Systematic Survey of Public Cloud Service Outage. *International Journal of Cloud Computing and Services Science, 2*(5), 1–15.

Liu, J. K., Au, M. H., Susilo, W., Liang, K., Lu, R., & Srinivasan, B. (2015). Secure Sharing and Searching for Real-Time Video Data in Mobile Cloud. *IEEE Network, 29*(2), 46–50.

Louk, M., & Lim, H. (2015). *Homomorphic Encryption in Mobile Multi Cloud Computing*. 2015 International Conference on Information Networking (ICOIN) (pp. 493–497). IEEE.

Machhi, S., & Jethava, G. B. (2016). *Feedback Based Trust Management for Cloud Environment*. Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (pp. 1–5).

Mayer, R. C., Davis, J. H., & Schoorman, F. D. (1995). An Integrative Model of Organizational Trust. *Academy of Management Review, 20*(3), 709–734.

McKnight, D. H., Carter, M., Thatcher, J. B., & Clay, P. F. (2011). Trust in a Specific Technology: An Investigation of Its Components and Measures. *ACM Transactions on Management Information Systems (TMIS), 2*(2), 12.

Mollah, M. B., Azad, M. A. K., & Vasilakos, A. (2017). Secure Data Sharing and Searching at the Edge of Cloud-Assisted Internet of Things. *IEEE Cloud Computing, 4*(1), 34–42.

Moore, C., O'Neill, M., O'Sullivan, E., Doröz, Y., & Sunar, B. (2014). *Practical Homomorphic Encryption: A Survey*. Circuits and Systems (ISCAS), 2014 IEEE International Symposium on (pp. 2792–2795). IEEE Computer Society. https://doi.org/10.1109/ISCAS.2014.6865753

Mundie, C., de Vries, P., Haynes, P., & Corwine, M. (2002). *Trustworthy Computing*. Technical Report, 10.

Nachiappan, R., Javadi, B., Calheiros, R. N., & Matawie, K. M. (2017). Cloud Storage Reliability for Big Data Applications: A State of the Art Survey. *Journal of Network and Computer Applications, 97*, 35–47.

Nguyen, H. T., Zhao, W., & Yang, J. (2010). *A Trust and Reputation Model Based on Bayesian Network for Web Services*. 2010 IEEE International Conference on Web Services (pp. 251–258). IEEE.

Noor, T. H., Sheng, Q. Z., Yao, L., Dustdar, S., & Ngu, A. H. (2015). CloudArmor: Supporting Reputation-Based Trust Management for Cloud Services. *IEEE Transactions on Parallel and Distributed Systems, 27*(2), 367–380.

Plank, J. S. (2013). Erasure Codes for Storage Systems: a Brief Primer. *Usenix Magazine, 38* (6), 44–50.

Pokharel, M., Lee, S., & Park, J. S. (2010). *Disaster Recovery for System Architecture Using Cloud Computing*. 2010 10th IEEE/IPSJ International Symposium on Applications and the Internet (pp. 304–307). IEEE.

Rajaasekharan, A. (2014). Data Reliability in Highly Fault-Tolerant Cloud Systems. Seagate. Retrieved June 6, 2020, from https://www.seagate.com/files/www-content/_shared/_masters/category-info/data-reliability-fault-tolerant-cloud-pv0031-1-1410-us.pdf

Sampaio, A. M., & Barbosa, J. G. (2018). A Comparative Cost Analysis of Fault-Tolerance Mechanisms for Availability on the Cloud. *Sustainable Computing: Informatics and Systems, 19*, 315–323.

Shaikh, R., & Sasikumar, M. (2015). Trust Model for Measuring Security Strength of Cloud Computing Service. *Procedia Computer Science, 45*, 380–389.

Singh, S., Jeong, Y. S., & Park, J. H. (2016). A Survey on Cloud Computing Security: Issues, Threats, and Solutions. *Journal of Network and Computer Applications, 75*, 200–222.

Sookhak, M., Akhunzada, A., Gani, A., Khurram Khan, M., & Anuar, N. B. (2014). Towards Dynamic Remote Data Auditing in Computational Clouds. *The Scientific World Journal*, 2014.

Sun, L., Singh, J., & Hussain, O. K. (2012). *Service Level Agreement (SLA) Assurance for Cloud Services: A Survey from a Transactional Risk Perspective.* Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia (pp. 263–266).

Tan, J., Gandhi, R., & Narasimhan, P. (2014). *STOVE: Strict, Observable, Verifiable Data and Execution Models for Untrusted Applications.* 2014 IEEE 6th International Conference on Cloud Computing Technology and Science (pp. 644–649). IEEE.

Tang, M., Dai, X., Liu, J., & Chen, J. (2017). Towards a Trust Evaluation Middleware for Cloud Service Selection. *Future Generation Computer Systems, 74*, 302–312.

Tebaa, M., & Hajji, S. Â. E. (2014). Secure Cloud Computing Through Homomorphic Encryption. Preprint arXiv:1409.0829. [Online]. Available: https://arxiv.org/abs/1409.0829.

Tian, H., Chen, Y., Chang, C. C., Jiang, H., Huang, Y., Chen, Y., & Liu, J. (2015). Dynamic-Hash-Table Based Public Auditing for Secure Cloud Storage. *IEEE Transactions on Services Computing, 10*(5), 701–714.

van der Werff, L., Real, C., & Lynn, T. (2018). Individual Trust and the Internet. In R. H. Searle, A. M. I. Nienaber, & S. B. Sitkin (Eds.), *The Routledge Companion to Trust* (pp. 391–407). Abingdon: Routledge.

Wang, H., Wu, S., Chen, M., & Wang, W. (2014). Security Protection between Users and the Mobile Media Cloud. *IEEE Communications Magazine, 52*(3), 73–79.

Yang, Y., Liu, J. K., Liang, K., Choo, K. K. R., & Zhou, J. (2015). *Extended Proxy-Assisted Approach: Achieving Revocable Fine-Grained Encryption of Cloud Data.* European Symposium on Research in Computer Security (pp. 146–166). Cham: Springer.

Yau, S. S., & Yin, Y. (2011). *QoS-based Service Ranking and Selection for Service-Based Systems.* 2011 IEEE International Conference on Services Computing (pp. 56–63). IEEE.

Yu, Y., Li, Y., Au, M. H., Susilo, W., Choo, K. K. R., & Zhang, X. (2016). *Public Cloud Data Auditing with Practical Key Update and Zero Knowledge Privacy.* Australasian Conference on Information Security and Privacy (pp. 389–405). Cham: Springer.

Yu, Y., Mu, Y., Ni, J., Deng, J., & Huang, K. (2015). *Identity Privacy-Preserving Public Auditing with Dynamic Group for Secure Mobile Cloud Storage.* International Conference on Network and System Security (pp. 28–40). Cham: Springer.