Kuwait J.Sci., Vol.48 (3), July.2021, pp(1-18)

Audit logs management and security - A survey

Ahmad Ali[1;*], Mansoor Ahmed[2], Abid Khan[3]
[1;2]*Dept. of Computer Science, COMSATS University, Islamabad, Islamabad, Pakistan*
[3]*Dept. of Computer Science, Aberystwyth University, Aberystwyth, UK*
**Corresponding author: engr.ahmadali@yahoo.com*

Abstract

Audit logs are key resources that show the current state of the systems and user activities and are used for cyber forensics and maintenance. These logs are the only source that can help in finding traces of some malicious activities or troubleshooting a system failure. Insight view for trouble-free availability of computing resources and performance monitoring and meaningful forensic audit depends on the management and archival system of audit logs. These logs are prone to multidimensional threats and superusers or system administrators have unprecedented access to these logs and can alter these logs as and when required. Similarly, repudiation is another serious issue in computer forensics and non-repudiation can be provided by a secure recording of event logs. Periodic backups, encrypted data transfer, off-site storage and certificate based storage of these logs are commonly being used. In this survey, we searched for the requirements of securing audit logs and available approaches to secure these logs. Based on the available literature, a taxonomy of audit log management is developed. We have drawn a comparison between these approaches and also highlighted the current challenges to these logs security and their available options.

Keywords: Audit Logs, Audit Logs Security, Log Management, Logs Immutability, Logs Storage .

1. Introduction

Information systems are composed of a variety of interconnected devices. Internet of Things (IoT) is an example of such connectivity. Servers, client terminals, switches, routers, firewalls, storage area networks (SANs), network attached storage (NAS) and other miscellaneous smart devices are examples of an information system's components. Software applications like operating systems, database management systems and other custom applications also generate logs to record their progress or profiles etc. Structure of these logs vary from machine to machine and each type of logs contains different information. Syslog, RSyslog and syslog-ng are well known standards for remote log management. Operating systems like windows and Linux etc generate multiple audit log files i.e. Security, and Authentication etc. Log files are depicted in Figure.1 and their usage is also listed in Table-1.New concepts of information systems as smart cities, smart homes and smart everything are the examples where the Internet of Things (IoT) has appeared as an area of higher impact, potential, and growth. Cisco Inc. predicted to have 50 billion connected devices by 2020 (Khan *et al*., 2017). The Internet of Things (IoT) is playing an amazing role in every dimension of our daily lives. It has covered many fields including health-care, automobiles, entertainment, industrial appliances, sports, homes, etc. The pervasiveness of IoT eases someone everyday activities, enriches the way people interact with the environment and surroundings, and augments our interactions with people and objects. This holistic vision, however, raises

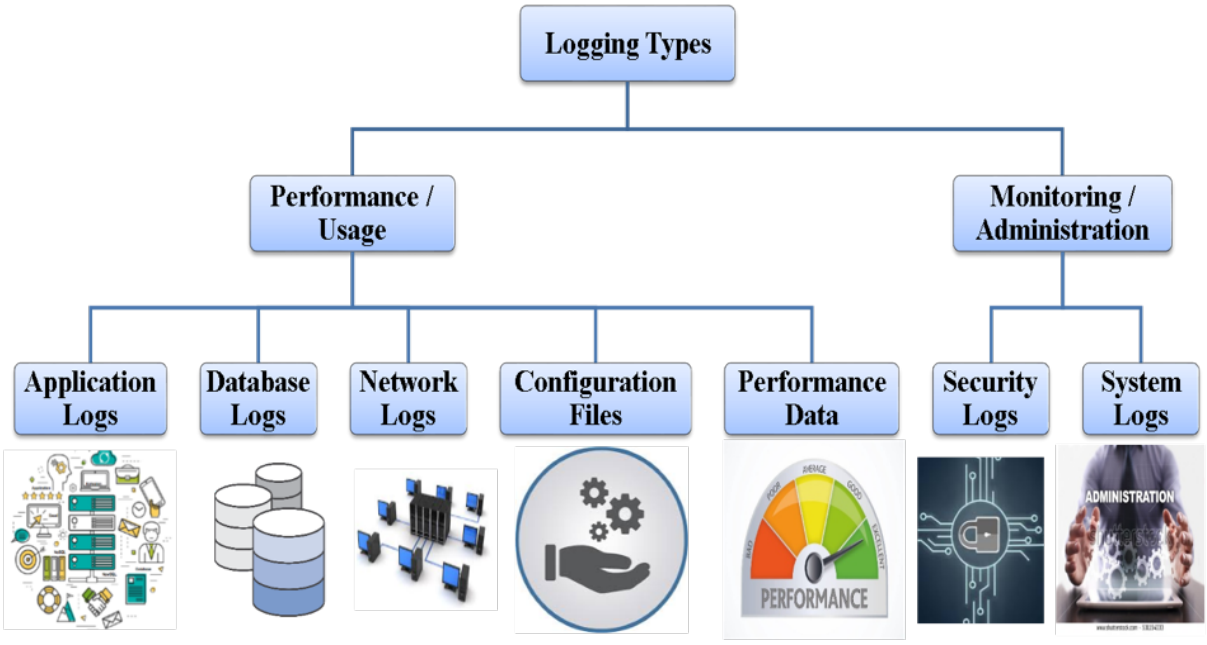


Fig. 1. Types of Log Files.

Table 1. Common Log Files, Respective Sources and Usage

Sr.	Log File	Source	Usage
1	System Log	Operating systems like Windows or Linux etc.	Operating System activities logs
2	Network Log	Network devices	Network communication logs
3	Security Log	Authentication schemes	Security related logs
4	Web-server Log	Web Servers	Related to Web Server activities
5	Application Log	Custom Application Logs	Developers monitor applications' behavior
6	Setup Log	Application installers	Installer logs
7	VM Log	Virtual Resource Managers like VM-Ware etc	Logs used by Virtual Machines
8	Custom Audit Log	Custom application logs	Miscellaneous authentication requests

some concerns also, like which level of security these systems could provide and how it offers and protects the privacy of its users (Khan *et al*., 2017, Stojmenovic & Wen, 2014) and (Wang *et al*., 2013).

The Internet is the backbone of this digital world and after the development of cloud-based applications like Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS) (Khan *et al*., 2016) similar technologies came into being. These systems are composed of multiple interconnected devices which belong to various categories and perform different functions. Distributed systems merged into cloud computing where central management and performance issues are being handled by an emerging research area known as Fog Computing (Stojmenovic & Wen , 2014; Bonomi *et al*., 2012).It is a decentralized computing infrastructure in which data, computing, storage and applications are distributed in the most logical and efficient way between the cloud and the data source. In fog computing (Boyle, 2015), smart devices like IoTs are known as Edge Nodes and connected to a cloud using a gateway. This gateway also acts as a micro data center (MDC) (Boyle, 2015). Along with its advantages and drawbacks fog computing has increased the availability of cloud services because of its localized behavior and has also improved the performance of the units (Bonomi, 2011; Stanciu, 2017).

Trust is an important factor for the utility of computing services, therefore role of trust and trust management is very significant in the usage, infrastructure, reliability of services, and the whole information system. Verifiable and trustable trust(Voas & Laplante, 2007) highly depends on the monitoring features being provided by service providers. Specification and recognition of trust behaviors and respective services also demand storage of patterns in a safe way which are required for better investigations (Yahyaoui *et al*., 2020). Information security and privacy are the key requirements for a user's trust in these technologies. Though physical security, safety, and availability of hardware resources are the key responsibilities of the cloud service providers, but trust is the core point for effective utilization of these systems. With the availability of trust, users can get most out of these systems.

Accountable audit logs can provide better level of trust for service providers as well as service tenants. In trusted computing systems evaluation criteria (Qiu *et al*., 1985), security requirements of audit logs are described as audit data must be protected from modification and un-authorized destruction to permit detection and after-the-fact investigation of security violations. These cloud-based information systems, as well as fog units, are prone to different security issues like authentication, authorization and insider threats. In contrast to the computing architectures discussed above, users' trust can be established by providing un-challenge-able availability, security and privacy management methods in the cloud environment. Virtual environments provided in the cloud are controlled / administered remotely. Availability can be provided by redundant resources. Data Security can be provided by a secure and legitimate authentication and authorization mechanism for remote users which is, really a challenging task. Functionality and performance of information systems including their unit devices can be monitored and explored by analyzing the logs generated by these interconnected devices. All above discussed operations are recorded in audit logs. Activities of insiders and local administrators along with other users can also be determined by these logs.

In system security perspective, any malicious activity can also be detected by analyzing these logs, therefore, security, safety, and integrity of these logs are very crucial. These logs also help administrators to counter information security issues, forensics is highly dependent on these logs and such resources play key roles in information systems security. Unfortunately, insiders have

unprecedented access to these logs and are prone to different multidimensional threats like privacy, insider threat (Ali *et al*., 2017), and alteration and deletion possibility. In this scenario, secure storage of such logs is important.

A new concept of trustworthiness-based trust management system has been proposed in (Ali *et al*., 2017) using a Log Analyzer. Storing activity/audit logs is a well-established mechanism to monitor performance and troubleshoot problems in these systems. Log management systems are provided to record activities of computing units, users, and administrators. These systems also help in information system security practices and forensics for digital evidence (Khan *et al*., 2016). Different sources generate logs in various structures and there is no single standard and consistent method for storage of these logs (Please see Table-1). Periodic backups, encrypted data transfer, off-site storage (Scarfone & Souppaya, 2006) and certificate-based storage of these logs are commonly being used. Repudiation is another serious issue in computer forensics. Non-repudiation can be achieved if the recording of event logs is secure and their integrity remains un-compromised.

1.1. Security Parameters

Basic building blocks of information security are confidentiality, integrity and availability which makes a CIA triad as shown in Figure 2. Confidentiality, integrity, authentication, non-repudiation, availability and privacy are the building blocks of a secure computing system (Khan *et al*., 2016).The presence of these information security measures for an information system somehow contributes to build users trust. So, it is an established fact that event logs are of much importance to ensure security parameters. In secure log management practices are detrimental to the computing environment. Log generation, collection, transfer, storage, security and analysis are common functions of a generic log management system. These logs are prone to different threats at each function or level. Log file rotation is also another aspect of logs management in which file may be rotated or truncated periodically or according to a predefined procedure. In log file rotation, successful segregation and separate storage of important logs is also very crucial.

1.2. Logging Modes

Recording an event at the time of system execution along with its metadata is known as logging process (Zeng *et al*., 2015). Logging needs additional processing and storage for collecting and storing various events when a system is performing smoothly. It is also very important to induce methods to detect system failures and susceptibilities affecting the system. These logs are well-known sources to identify sequential steps of the susceptibilities. Two main logging modes (Circular Logging and Linear Logging) (Qiu *et al*., 1985)

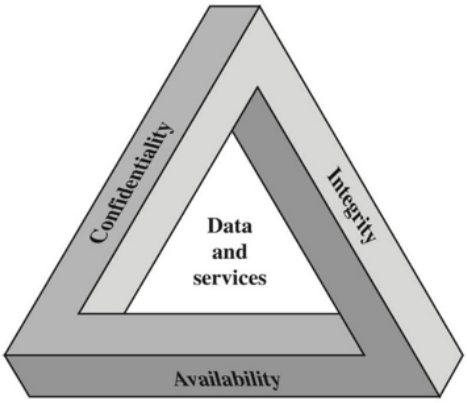


Fig. 2. A CIA Triad.

are in practice that explains how the logs should be stored and which information should be recovered from logs to investigate different vulnerabilities. In circular logging, storage remains available for reuse and no special maintenance is required but could not retain logs for longer periods. In linear logging, logs are stored for longer duration and human intervention is required to process the queue and free occupied space.

1.3. Logs Storage Scenarios

Logs are required to be stored with efficient handling mechanisms. Logs storing techniques are based on the utility of logs. De-centralized logging, centralized logging, distributed logging are known methods for storage of logs. Depending on the availability of resources, processing, and requirements, initially, these logs were stored on the computers/edge nodes. In case of any failure or technical issue, administrators had to visit the site physically, this technique is known as decentralized storage of logs. Because of in-efficiencies and management issues in decentralized logs storage, central log storage scheme was introduced. In centralized log storage, all the events are captured at a central location where monitoring and evaluation of performance and issues are carried out centrally. This technique was not considered suitable for a large network of networks where networks reside at a distant location. In case of failure of a central logging or dis-connectivity, all the units remain out of sight and monitoring. To handle such issues, distributed log storage is in practice where distributed logging servers are configured to store logs of concerned devices/computers and further synched with other servers for data consistency. These Logs can be stored in CSV. files as well as in different databases e.g. plain text files and RDBMS. Graph databases are also being adopted because of their dynamic nature

1.4. Why - Securing Audit Logs?

Non-repudiation is the guarantee of evidence that the actor cannot deny his activity. Specifically in computer forensics, non-repudiation refers to the evidence recording capability to ensure that the miscreant cannot deny his activity. Audit logs are the only source to record the activities in a computing environment. If this recording is not secure enough then proof of any malicious activity can be destroyed.

In Computer Forensics, the most important factor of information security is "Non-Repudiation" which can be managed by recording each sensitive activity of user. Along with some handling issues like (real-time storage, accessibility and alert generation based on some specific event) following are critical issues in log storage and analysis schemes. A logging scheme is considered to be a comprehensive resource for recording critical events. These requirements can be categorized into basic security requirements and extended security requirements. A taxonomy of log management is shown in Figure. 4. Confidentiality, Integrity, Availability, Non-Repudiation, and Privacy are considered as basic security requirements of a secure and trusted logging scheme. Confidentiality means the prevention of unauthorized access. Integrity is required to safeguard data from being altered or even deleted. Availability means the assurance and guarantee of data being available when required in the form it was saved. Non-Repudiation is the property required to provide proofs having sufficient data of occurrence of an activity. Privacy treats personal matters and that should not be leaked or shared with others (Khan *et al*., 2016). In Extended Security Requirements, along with the

Table 2. Security parameters for audit log management system

Ser	Parameter	Highlighted by
SP-1	Confidentiality	SP-1 to SP-5 are commonly highlighted by following:- NIST Guidelines (Scarfone & Souppaya, 2006),BBOX (Accorsi, 2010), SecLAAS (Zawoad *et al.* 2013), Immutable authentication and integrity schemes ((Yavuz 2018)),Data-centric accountability ((Jin *et al.* 2018))
SP-2	Integrity	
SP-3	Availability	
SP-4	Authenticity	
SP-5	Immutability	
SP-6	Heterogeneity Support	Heterogeneity and dynamicity of clouds (Reiss *et al.* 2012)
SP-7	Semantics	Evidence-based context-aware log data management (Sato *et al.* 2016)
SP-8	Statistics Sharing	Trust Management (Ali et al., 2017)

basic security requirements, logging schemes are required to provide Correctness, Integrity with forwarding Security, Immutability, Insider Threat Mitigation, and Statistics Sharing. Correctness is to deal with the authenticity of information whereas Integrity with forward Security makes logs data protected in case of any miscreants' activity. If a system gets compromised, its previously stored logs data should not be altered by the miscreants or even by the system administrators by any means. Immutability covers all the aspects relating to the data consistency i.e. it makes logs data un-changeable after its storage. Insider Threat mitigation is to deal with insiders with malicious intentions because they have unprecedented access to the system resources. Verifiability is another aspect of logs storage in which these logs can be verified by some other resources (Waters *et al*., 2004). Since, logs are an excellent source of information for administrators and help them in maintenance and up-keeping the computing systems. Because of heterogeneity in computing units and logs structure (Reiss *et al*., 2012, Sato *et al*., 2016), it is very difficult to make them machine understandable. Semantic web is a promising technology for making data machine understandable. Researchers are working to make social machines (Hendler & Berners-Lee, 2010) which can semantically interact and share information. Furthermore, extraction of relevant information from logs is very much required (Henze *et al*., 2017).

Trustworthiness of a service provider is closely related to the reality of activities and sharing statistics of such activities which contributes towards stakeholders' trust. A logging scheme should have such methods to share statistics of activities to establish its trustworthiness (Ali *et al*., 2017).

2. Known Types of Attacks on Logging Schemes

A comprehensive log management system provides different functions. Log generation, collection, transfer/transport, storage and analysis are some very common functions of a generic log management system. Every function is prone to different threats and vulnerabilities. Users with malicious intentions can perturb the whole log management system by distracting or disabling any function. Summarizing the threats and vulnerability points, following attacks are common to logging schemes that can disrupt logging functions (Henze *et al*., 2017) and lead to severe consequences. Withhold Attack: Any malicious activity which temporarily hold the logs and not pass on the certain message to the network with in the required time span, e.g., to block the authorization/de-authorization to devices for a specific time and date (Henze *et al*., 2017).

- **Modification Attack:** Malicious users / administrators may change log entriesbefore forwarding to a storage system, e.g., may alter configuration values (Henze *et al*., 2017).
- **Insertion Attack:** Malicious users or even man in the middle may create fake messages and forward to the controller, e.g., to take access of a particular device. This category of attacks also extends duplication of real messages which may cause an in consistent state of the system (Henze *et al*., 2017).
- **Reorder Attack:** In log recordings, sequence of actions along with their happening time means a lot and any change in the order of messages prior their distribution in the network, surely change the entire semantics contained in the messages (Henze *et al*., 2017).
- **Privacy Attack:** Log data contains lot of information regarding users and their activities, which may cause privacy issues for a specific environment. These logs may also be leaked out by the attackers (Henze *et al*., 2017).

3. Literature review process

To conduct this survey in information systems, we sought guidelines from the methodology for systematic literature review (SLR) proposed by Chitu Okoli (Okoli, 2015). They devised an eight-step methodology as shown Figure 3. These steps are concluded as under:

3.1. Steps Followed - Review Methodology

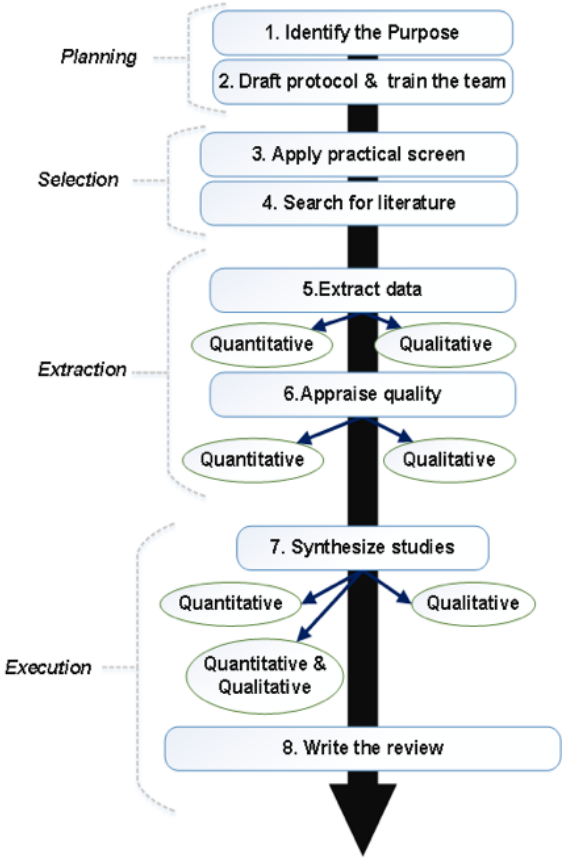
Following steps are highlighted in the subject methodology for a useful systematic literature survey.

3.1.1. Identify the purpose

Purpose of this SLR is to review available methods for logs security to comply with the requirements / parameters of secure and trusted information systems.

3.1.2. Draft protocol and train the team

A very simple but comprehensive protocol for this review is to find literature describing methods or schemes available for the subject purpose.



3.1.3. Apply practical screen

Selected literature (as summarized in Table 3) was screened to find available practical solutions and their application in this domain and shortlisted to 19 number of papers. Comparison metrics of these schemes are given in Table 4.

3.1.4. Search for literature

Screened literature is further explored with reference to the security parameters as given in Table 2.

3.1.5. Extract data

A comparative Table 4 is populated by exploring the selected literature with reference to their features/functions.

3.1.6. Appraise quality

Further to the comparison of logs security schemes, their applicability and usage is critically analyzed.

3.1.7. Synthesize studies

Available schemes are explored and analyzed with the requirements of logs security and additional features for the enhancements of logs security schemes are highlighted.

3.1.8. Writing the review

To culminate this SLR and after synthesis of the results, weaknesses of existing schemes and guidelines for securing logs are proposed.

3.2. Research Questions

Based on the forensic requirements of information systems we drawn the following questions for our research and conducted the literature review.

- What are the existing techniques for logs security?
- What methodologies are being followed in these log securing techniques?
- What implications will these findings have, when creating new systems?
- What are the limitations of these techniques?

3.3. Literature Search

To find the answers of research questions described in the section 'Research Questions', relevant papers for logs security were collected using famous online digital libraries like IEEE, Elsevier, SpringerLink, ACM, and other miscellaneous resources.

Table 3. Literature Reviewed

Year	IEEE	Elsevier	ACM	Springer	Misc	Total
2020					1	1
2019					1	1
2018	1					1
2017	3	1		2	5	11
2016	2			2		4
2015	2	1		1	2	6
2014	3			3	1	7
2013	4	2	1	1	3	11
2012	2		1			3
2011	2			1	1	4
2010	2			1		3
2009	1		1		1	3
Earlier	5	1	4	4	11	25
Total	**27**	**5**	**7**	**15**	**26**	**80**

We searched the terms “Logs Security”, “Secure Audit Log management”, and Logs Immutability. We conducted a title/ abstract/ keyword search related to our survey. This resulted in an initial set of 80 articles in total which were further shortlisted as depicted in Table 3.

3.4. Evaluation Metrics

Secure logging schemes can be evaluated in quantitative and qualitative approaches. In this research we focused on the functions or features being provided in the available schemes. Available schemes are explored in terms of their features compared with the security parameters as given in Table 2.

3.5. Related Work

In computing environments, log management is quite a mature subject. Various solutions are available for log management. GFI Events Manager(GFI Event Log Viewer and Analyzer, Network Monitoring and Management Software-GFI EventsManager n.d.), Syslog-ng (syslog-ng-Log Management Solutions n.d.), Manage-Engine Log Storage and Analyzer (ManageEngine EventLog Analyzer - SIEM Log management software. n.d.), LOGalyze (LOGalyze - Open Source Log Management Tool, SIEM, Log Analyzer n.d.), Splunk (Enterprise Security (SIEM), Premium Solutions, Splunk) and Open source SIEM Solutions, OSSEC and OSSIM (OSSEC Audit Log Storage (Open source Security; OSSIM: The Open Source SIEM — Alien Vault) etc. are few examples of such solutions. In these schemes, a central logging server is required to manage various logs. All these log management applications are capable enough to store and analyze logs of various IoT units but these solutions have the limitation of rendering full control to the system administrators that can compromise the integrity of these logs. Different solutions have been proposed by researchers to maintain the integrity of the logs even after a system compromise but insider threat or escalated access permissions of system administrators has not been addressed comprehensively. To maintain the data security, symmetric as well as asymmetric encryption schemes have been proposed. Waters *et al*. (Waters *et al*., 2004) proposed an encrypted and search-able audit log scheme that provides verifiability and protects confidentiality and especially searchable encryption would allow decrypting only relevant messages in the message log. Attila Yavuz has proposed a cryptographic scheme as append only secure audit logging (LogFAS) in (Yavuz *et al*., 2012a). Schneier & Kelsey, (1998) have presented a scheme for secure logging that detects attempts of deletion or modification on untrusted machines. LogCrypt (Holt, 2006) proposes

an asymmetric cryptography scheme for log security that secures the logs using publicly verifiable encryption. Considering a distributed setting, Accorsi, (2010) proposed "BBox" approach to apply trusted computing to ensure authenticity and confidentiality of log entries. Another solution SLOPPI (Von *et al*., 2013) was presented that used encryption based scheme for data integrity as well as policy compliance. These schemes have not addressed additional overhead of encryption / decryption processes and maintenance of keys for these encryption types moreover these schemes have not catered for immutability, statistics distribution and semantics of logs altogether.

Aggregation Schemes using encryption like Blind Aggregate Forward (BAF), Improved Blind Aggregate Forward (BAFi), and Fast Immutable BAF (Fi-BAF) etc. have been proposed in (Yavuz & Ning, 2009; Yavuz *et al*., 2012a; Yavuz & Kampanakis, 2015). Similarly, (Ma & Tsudik, 2007) and (Hartung *et al*., 2017) proposed aggregation based schemes. FssAgg scheme (Ma & Tsudik, 2007) has been proposed for aggregation based secure authentication to protect previously logged in long sessions only.

CLASS (Ahsan *et al*., 2018) is another logs management scheme where provided log security by Proof of past logs (PPL) using Rabin's fingerprint and bloom filter based approach. In this scheme, only a cryptographic scheme as append only integrity of logs can be maintained but remaining requirements are missing.

eCLASS (Park & Huh, 2019) is an edge nodes log confidentiality protection scheme which implements RSA based encryption for cloud environment. This scheme provided data confidentiality and integrity using encryption whereas availability, immutability, statistics sharing with stakeholders is missing.

Using Authentication Data Structures (Balloon) (Pulls & Peters, 2015) proposes another scheme to store log entries in a famous balloons addition approach which introduced only a data structure for secure authentication. Prime objective of these schemes is integrity of logs however other suggested security parameters given in Table 2 were not addressed.

Logging as a Service has also been proposed for cloud users after the deployment of cloud architectures as IaaS, PaaS, and SaaS. This scheme also depends on trustworthiness and integrity of cloud service provider where insiders have the possibility to manipulate these logs. Söderström, *et al*.,(2013), proposed a scheme for receiving, storing and analyzing these logs using a central log server and SecLaaS was proposed in (Zawoad *et al*. 2013) which uses cloud functions to store logs. The major limitation of these proposed schemes can be compromised by miscreant cloud administrators. Khan *et al*., (2017) proposed "Secure Logging as a Service using Reversible Watermarking" where logs are stored in the cloud for longer duration and content authentication is carried out by reversible watermarking. This scheme has catered forensics of contents authentication and integrity of logs only and has ignored other parameters given in Table 2.

Hardware-based logging schemes are explained in (Jaquette *et al*.,), authors introduced storage of logs in a forward integrity concept using write once and read multiple (WORM) scenarios. Tamper Proof Storage of Logs using Trusted Platform Module 2.0 is explained in (Sinha *et al*.,2014). EmLog as a Tamper-Resistant System Logging introduced for Constrained Devices with Trusted Execution Environments (TEEs) (Shepherd *et al*., 2017). Secure Audit Logging with iButton based tamper resistant hardware solution was proposed in (Chong *et al*., 2003) which further use encryption and ROM storage. File System based Logging Schemes were discussed in (Ko *et al*., 2011) whereas contradictory proof for secure audit log storing techniques was highlighted in (Rosenblum & Ousterhout, 1992).

These hardware-based schemes are constrained in terms of storage capacity and processing. Their capacity, cost, and

availability are not suitable for larger networks.

Immutability of logs using **Distributed Ledger Technology** is explained in (Cucurull & Puiggal´ı, 2016) and Privacy logs storage using Blockchain is proposed in (Sutton & Samavi, 2017) where only private data is protected from public auditors. Similarly, another approach "Proactive Forensics in IoT" (Janjua *et al*., 2020) used holochains to record audit logs in fog environment using bots. However, this scheme could not confirm authenticity, availability and other requirements as the bot can be stopped any time by the insiders.

Various **semantics extraction** based solutions have been proposed by different researchers also. Adam *et al* highlighted the needs of log analysis based on the semantics presented by the logs (Oliner & Stearley, 2007). Authors have explained data extraction using data mining approaches and methods to deal with heterogeneity in (Forcher *et al*., 2011) and (Shafiq, 2015) whereas security, immutability and other basic as well as extended security requirements were not addressed.

Anonymizing the logs data during its handling will surely help in data security and privacy as proposed in (Rajalakshmi *et al*., 2014). FP-Growth- a scheme of central logs collection and analysis was proposed in (Amar *et al*., 2016). This scheme is useful for a smaller network environment, in case of larger networks logs control and verifiability are real challenges. In these schemes basic requirements of data integrity, immutability and semantics are missing in the proposed scheme.

Another recent approach **SLiC** (Blass & Noubir, 2017) has been proposed for Secure Logging having a built-in Crash tolerance mechanism. In this approach, log time-stamp and its data are stored separately. After that, these entries are randomized to protect their identity. Though, this presented a unique approach for logs storage, but it failed to provide immutability of logs, logs heterogeneity, semantics and statistics sharing. Furthermore, this technique is unable to provide verifiability.

Henze *et al*., (2017) proposed a framework for secure communication in IoT. In order to protect an IoT network from a cloud services provider with malicious intentions, the proposed framework allows for configuration of IoT network from a central location. It stores logs of control messages at multiple locations which can be verified through different gateways. The size of log messages is minimized by removing old messages continuously. The verification of log messages is then used to indicate malicious behavior which in turn protects cloud-based IoT from modification, withholding, insertion and reordering of log messages. This framework provides immutability for control messages or configuration files on distributed locations, similar approach is suggested in our proposed scheme for immutability of logs using distributed ledgers technology (DLT).

3.6. The Taxonomy

A lot of work is going on for the development of a useful audit log management. In this survey, a taxonomy is developed to highlight major components of this research domain. This taxonomy provides and oversight of log management research area. This taxonomy (Fig.4) provides security requirements, securing techniques, storage methods and semantics are the major research components. Security requirements are further divided into basic as well as extended security requirements as shown in the taxonomy. Based on the available research in this domain, securing techniques are categorized as per their method to secure log entries. Log entries can be stored in different ways as text files, relational databases, and markup files like XML and JSON documents.

3.7. Comparison of Selected Schemes

From a broader research literature base, shortlisted logging security schemes are explored and compared with the security

parameters given in Table 2. Each highlighted scheme is strong enough to provide claimed feature(s). As a wholesome, all the features required and highlighted in different researches are not available in a unique scheme. Tabular comparison of this study is presented in Table 4.

4. Conclusion & Future Directions

The audit log management is an uphill task which has been under research for a long time. A substantial research has been carried out for log management and multiple methods for log management as well as logs security have been proposed. In this research, we have explored audit log management approaches to find out the basic and extended requirements. Threats and vulnerabilities posed to this domain are also discussed. Parameters required for the security of audit logs and to provide counter measures against threats are also extracted from research articles published from time to time.

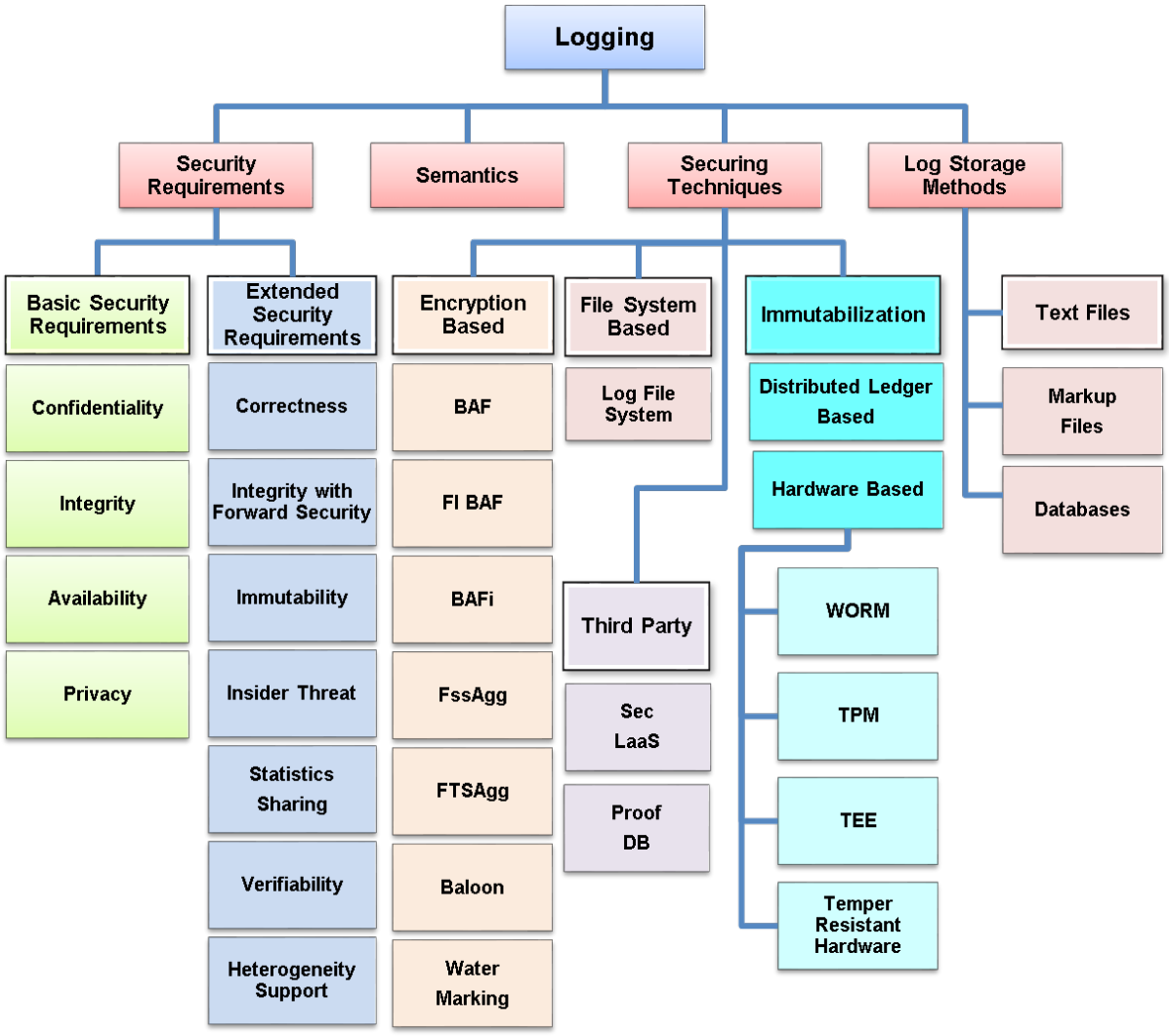


Fig. 4. Audit Log Management - Taxonomy

Table 4. Comparison of Logging Schemes

Ser	Paper	Method	*SP-1	SP-2	SP-3	SP-4	SP-5	SP-6	SP-7	SP-8	Remarks
1	SecLaaS (Zawoad *et al.*, 2013)	Proof DB	✓	✓	✓	✓	X	X	X	X	Dependent on Trusted Admin
2	SPOC / FP Growth (Amar *et al.*, 2016)	Centralized Log Storage	✓	✓	✓	✓	X	X	X	X	Dependent on Trusted Admin
3	BAF (Yavuz & Ning, 2009)	Aggregate Signature Schemes	✓	✓	X	✓	X	X	X	X	Constrained Devices Only
4	BAFi (Kampanakis & Yavuz, 2015)	"	✓	✓	X	✓	X	X	X	X	"
5	FI-BAF (Yavuz *et al.*, 2012a)	"	✓	✓	X	✓	X	X	X	X	"
6	LogFAS (Yavuz *et al.*, 2012b)	"	✓	✓	X	X	X	X	X	X	Verifiability Issues
7	Immutable auth. & integrity schemes (Yavuz, 2018)	"	✓	✓	X	✓	X	X	X	X	Log Confidentiality \Integrity Solution only
8	FssAgg (Ma & Tsudik, 2007)	Seq. Signatures	X	✓	X	✓	X	X	X	X	Log Integrity Solution only
9	Balloon (Pulls & Peters, 2015)	Enc. Data structure	✓	✓	X	✓	X	X	X	X	Log Integrity Solution only
10	Semantic Logging (Forcher *et al.*, 2011)	RDF Generation	X	X	X	X	X	✓	✓	X	User Profiling for Document Aware System (DAS) only
11	Semantically Formalized Logging (Sokolowski & Banks,2015)	Semantic Formalism for Logs	X	X	X	X	X	✓	✓	X	Only semantics, Basic as well as extended Security parameters are missing
12	BBox (Accorsi, 2010)	TPM Based	✓	✓	✓	✓	X	X	X	X	Hardware Based Solution
13	EMLog (Shepherd *et al.*, 2017)	TEE	✓	✓	✓	✓	X	X	X	X	Tamper Resistant Hardware Based Solution

* Legend	* SP-1 Confidentiality	SP-2 Integrity	SP-3 Availability	SP-4 Authenticity	SP-5 Immutability	SP-6 Heterogeneity Support	SP-7 Semantics	SP-8 Statistics Sharing

Table 4. Comparison of Logging Schemes (Continued…)

Ser	Paper	Method	*SP-1	SP-2	SP-3	SP-4	SP-5	SP-6	SP-7	SP-8	Remarks
14	Data-centric accountability & auditability (Jin *et al.* 2018)	Auditing and arbitration	✓	✓	X	✓	X	X	X	X	Dependent on Trusted third party for arbitration
15	SLOPPI (Von Eye *et al.* 2013)	Central Log Server	✓	✓	✓	✓	X	X	X	X	No distributed storage and statistics sharing
16	SLiC (Blass & Noubir, 2017)	Segregates event message and time stamp, randomize and store separately	✓	X	X	X	X	X	X	X	Protect logs data by randomizing logs positions
17	Practical and Robust Secure Logging (Hartung *et al.* 2017)	Sequential Aggregate Signatures	✓	✓	X	X	X	X	X	X	Publicly-verifiable secure logging scheme
18	Distributed Configuration, Authorization and Management(Henze *et al.* 2017)	DLT based Configuration Logs storage, Logs Verification and distributed control on configuration logs	✓	✓	✓	✓	✓	X	X	✓	Covers only Configurations logs storage,
19	Secure Logging as a Service Using Reversible Watermarking (Khan *et al.* 2017)	Strong Content authentication mechanism	✓	✓	X	✓	X	X	X	X	Real-time log reception and storage is not considered
20	CLASS (Ahsan *et al.* 2018)	Content concealment using Cryptography	✓	✓	✓	✓	X	X	X	X	No immutability and statistics sharing with stakeholders
21	eCLASS (Park & Huh, 2019)	Encryption based logs security	✓	✓	X	X	X	X	X	X	No immutability and statistics sharing with stakeholders, Missing availability
22	Privacy-aware log-preservation architecture (Janjua *et al.* 2020)	Holochain based logs security	✓	✓	X	X	✓	X	X	X	Logs heterogeneity support and semantics are missing

* Legend	* SP-1 Confidentiality	SP-2 Integrity	SP-3 Availability	SP-4 Authenticity	SP-5 Immutability	SP-6 Heterogeneity Support	SP-7 Semantics	SP-8 Statistics Sharing

In addition to that, available schemes for the security of audit logs and their management are explored and a comparison is drawn to segregate their features. After this review, deficiencies in audit log management applications have been earmarked.

While classifying the threats to log management approaches, we also found that insiders' threat is totally out of sight in the whole research. Super users and administrators are fully trusted people and organizations have blind faith on them. Now, it is time to uncover the insiders' threats to information systems as well as log management systems.

After completing this survey, we have reached to a conclusion that a secure and mission critical computing environment must have such an audit log management which must confer the following:-

Confidentiality is required in a log management where logs data cannot be accessible by un-authorized people.

Integrity is another property of a log management system which assures logs data consistency, completeness and total accuracy.

Availability feature confirms that logs data will be available to the authorities even after a system compromise.

Authenticity provided by a log management system confirms that the evidence recorded is genuine, legitimate and real without any modification.

Immutability is the most important functionality where recorded evidence cannot be modified/altered in any circumstance what come may.

Heterogeneity Support covers the strength of a log management system where the logs generated by different devices are not only recorded but can also be meaningfully.

Semantics can be achieved by enrichment of log entries with respect to the structure of logs and making them understandable by tagging the different parts of a log entry.

Statistics Sharing is very much important for the trust of cloud services tenants. By sharing real-time statistics of a computing environment will surely increase its trustworthiness.

In our future work, we are planning to develop a comprehensive solution to provide missing features collectively in a transparent way and have strength against reported threats and vulnerabilities.

References

Accorsi, R. (2010), BBox: A distributed secure log architecture, in 'European Public Key Infrastructure Workshop', Springer, pp. 109–124.

Ahsan, M. M., Wahab, A. W. A., Idris, M. Y. I., Khan, S., Bachura, E. & Choo, K.K. R. (2018), 'Class: Cloud log assuring soundness and secrecy scheme for cloud forensics', IEEE Transactions on Sustainable Computing.

Ali, A., Ahmed, M., Ilyas, M. & Ku¨ ng, J. (2017), MITIS-An Insider Threats Mitigation Framework for Information Systems, in 'International Conference on Future Data and Security Engineering', Springer, pp. 407–415.

Ali, A., Ahmed, M., Khan, A., Ilyas, M. &Razzaq, M. S. (2017), A trust management system model for cloud, in 'International Symposium on Networks, Computers and Communications (ISNCC), 2017', IEEE, pp. 1–6.

Amar, M., Lemoudden, M. & El Ouahidi, B. (2016), Log file's centralization to improve cloud security, in '2nd International Conference on Cloud Computing Technologies and Applications (CloudTech), 2016', IEEE, pp. 178–183.

Blass, E.-O. &Noubir, G. (2017), 'Secure Logging with Crash Tolerance.',IACR Cryptology ePrint Archive 2017, 107.

Bonomi, F. (2011), Connected vehicles, the internet of things, and fog computing, in 'The eighth ACM international workshop on vehicular internet working (VANET), Las Vegas, USA', pp. 13–15.

Bonomi, F., Milito, R., Zhu, J. &Addepalli, S. (2012), Fog computing and its role in the internet of things, in 'Proceedings of the first edition of the MCC workshop on Mobile cloud computing', ACM, pp. 13–16.

Boyle, B. (2015) (accessed December 20, 2016), Edge market will boost demand for micro data centers. URL: http://www.datacenterdynamics.com/powercooling/edge-market-will-boost-demand-for-micro-data-centers/95070

Chong, C. N., Peng, Z. &Hartel, P. H. (2003), Secure audit logging with tamper-resistant hardware,in 'IFIP International Information Security Conference', Springer, pp. 73–84.

Cucurull, J. & Puiggal´ı, J.(2016), Distributed immutabilization of secure logs, in 'International Workshop on Security and Trust Management', Springer, pp. 122–137.Enterprise Security (SIEM), Premium Solutions, Splunk (n.d.).

Forcher,B., Agne, S., Dengel,A., Gillmann, M.& Roth-Berghofer, T. (2011), Semantic logging: Towards explanation-awaredas, in 'International Conference on Document Analysis and Recognition (ICDAR),2011', IEEE, pp. 1140–1144.

GFI EventLogViewerandAnalyzer,Network Monitoring and Management Software— GFI Events Manager (n.d.). Accessed on May 10,2018. **URL**: https://www.gfi.com/productsandsolutions/network-security-solutions/gfieventsmanager.

Hartung, G., Kaidel, B., Koch, A., Koch, J. &Hartmann, D. (2017), Practicaland Robust Secure Logging from Fault-Tolerant Sequential Aggregate Signatures,in 'International Conference on Provable Security', Springer, pp. 87–106.

Hendler, J. & Berners-Lee, T. (2010), 'From the Semantic Web to social machines: A research challenge for AI on the World Wide Web', Artificial Intelligence **174**(2), 156–161.

Henze, M., Wolters, B., Matzutt, R., Zimmermann, T. &Wehrle, K. (2017), Distributed configuration, authorization and management in the cloud-based internet of things, in 'Trustcom/BigDataSE/ICESS, 2017 IEEE', IEEE, pp. 185–192.

Holt, J. E. (2006), Logcrypt:forward security and public verification for secure audit logs, in 'Proceedings of the 2006 Australasian workshops on Grid computing and e-research-Volume 54', Australian Computer Society, Inc., pp. 203–211.

Janjua, K., Shah, M. A., Almogren, A., Khattak, H. A., Maple, C. & Din, I. U. (2020), 'Proactive forensics in iot:Privacy-awarelog-preservationarchitecture in fog-enabled-cloud using holochain and containerization technologies', Electronics **9**(7), 1172.

Jaquette, G. A., Jesionowski, L. G., Kulakowski, J. E. & McDowell, J. A. (n.d.), 'Low cost tamper-resistant method for write-once read many (WORM) storage'.

Jin, H., Zhou, K. &Luo, Y. (2018), 'Aframework with data-centric accountability and auditability for cloud storage', The Journal of Supercomputing pp. 1–24.

Kampanakis,P. &Yavuz, A. A. (2015), 'BAFi: a practical cryptographic secure audit logging scheme for digital forensics',Security and Communication Networks 8(17), 3180–3190.

Khan, A., Yaqoob, A., Sarwar, K., Tahir, M. & Ahmed, M.(2017), 'Secure Logging as a Service Using Reversible Watermarking', Procedia Computer Science **110**, 336–343.

Khan, M. A. & Salah, K. (2017), 'Iot security: Review, blockchain solutions, and open challenges',Future Generation Computer Systems.

Khan, S., Gani, A., Wahab, A. W. A., Bagiwa, M. A., Shiraz, M., Khan, S. U., Buyya, R. &Zomaya, A. Y. (2016), 'Cloud log forensics: Foundations, state of the art, and future directions', ACM Computing Surveys (CSUR **49**(1), 7.

Ko, R. K., Jagadpramana, P. & Lee, B. S. (2011), Flogger: A file-centric logger for monitoring file access and transfers within cloud computing environments, in 'IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2011', IEEE, pp. 765–771.

LOGalyze - Open Source Log Management Tool, SIEM, Log Analyzer (n.d.).Accessed on May 10,2018. **URL**:http://www.logalyze.com/

Ma, D. & Tsudik, G. (2007), Forward-secure sequential aggregate authentication,in 'IEEE Symposium on Security and Privacy,2007. SP'07', IEEE, pp. 86–91.

ManageEngineEventLog Analyzer - SIEM Log management software. (n.d.).Accessed on May 10,2018. **URL**: https://www.manageengine.com/

Okoli, C. (2015), 'A guide to conductinga standalone systematic literature review'.

Oliner, A. &Stearley,J. (2007), What supercomputers say: A study of five system logs, in 'Dependable Systems and Networks, 2007. DSN'07.37th Annual IEEE/IFIP International Conference on', IEEE, pp. 575–584.

OSSEC Audit Log Storage (Opensource Security (n.d.). Accessed on May 10,2018. **URL**: https://www.ossec.net/

OSSIM: The Open Source SIEM — AlienVault (n.d.). Accessed on May 10,2018. **URL**:https://www.alienvault.com/products/ossim

Park,J. &Huh, E.-N. (2019), 'eclass: Edge-cloud- log assuring-secrecy scheme for digital forensics', Symmetry **11**(10), 1192.

Pulls, T. & Peters,R. (2015), Balloon: A forward secure append-only persistent authenticated data structure, in 'European Symposium on Research in Computer Security', Springer, pp. 622–641.

Qiu, L., Zhang, Y., Wang, F., Kyung, M. & Mahajan, H. R. (1985), Trusted computer system evaluation criteria, in 'National Computer Security Center', Citeseer.

Rajalakshmi,J. R., Rathinraj, M. &Braveen, M. (2014), Anonymizing log management processfor secure logging in the cloud, in 'International Conference on Circuit, Power and Computing Technologies (ICCPCT),2014', IEEE, pp. 1559–1564.

Reiss, C., Tumanov, A., Ganger, G. R., Katz, R. H.&Kozuch, M. A. (2012), Heterogeneity and dynamicity of clouds at scale: Google trace analysis, in 'Proceedings of the Third ACM Symposium on Cloud Computing', ACM, p. 7.

Rosenblum, M. &Ousterhout,J. K. (1992), 'The design and implementation of a log-structuredfile system', ACM Transactions on Computer Systems (TOCS **10**(1), 26–52.

Sato, T., Himura, Y. & Yasuda, Y. (2016), Evidence- based context-aware log data management for integrated monitoring system, in 'Network Operations and Management Symposium (APNOMS), 2016 18th Asia-Pacific', IEEE, pp. 1–4.

Scarfone, K. K. & Souppaya, M. P. (2006),Guide to Computer Security Log Management, Technical report.

Schneier, B. & Kelsey,J. (1998), Cryptographic Support for Secure Logs on Untrusted Machines., in 'USENIX Security Symposium', Vol. **98**, pp. 53–62.

Shafiq, M. O. (2015), Semantically Formalized Logging and Advanced Analytics for Enhanced Monitoring and Management of Large-scale Applications, PhD thesis, University of Calgary.

Shepherd,C.,Akram,R.N.&Markantonakis, K. (2017), 'EmLog: Tamper-Resistant System Logging for Constrained Devices with TEEs', arXiv preprint arXiv:1712.03943.

Sinha, A., Jia, L., England,P. &Lorch, J. R. (2014), Continuous tamper-proof logging using tpm 2.0, in 'InternationalConference on Trust and Trustworthy Computing', Springer, pp. 19–36.

So¨derstro¨m, Olof and Moradian, Esmiralda (2013), 'Secure audit log management', Procedia Computer Science **22**, 1249–1258.

Sokolowski,J. A. & Banks, C. M. (2015), Agent implementation for modeling insider threat, in 'Proceedings of the 2015 Winter Simulation Conference', IEEEPress, pp. 266–275.

Stanciu, A. (n.d.) (2017), Blockchainbased distributedcontrol system for edge computing.

Stojmenovic, I. & Wen, S. (2014), The fog computing paradigm: Scenarios and security issues, in 'Federated Conference on Computer Science and Information Systems (FedCSIS), 2014', IEEE, pp. 1–8.

Sutton, A. &Samavi, R. (2017), Blockchain Enabled Privacy Audit Logs, in 'International Semantic Web Conference', Springer, pp. 645–660. syslog-ng - Log Management Solutions (n.d.). Accessed on May 10,2018. **URL**: https://syslog-ng.com/

Voas, J. & Laplante, P. (2007), 'The services paradigm: Who can you trust?', IT Professional **9**(3), 58–61.

Von Eye, F., Schmitz, D. & Hommel, W. (2013), SLOPPI-A Framework for Secure Logging with Privacy Protection and Integrity, in 'Proceedings of the Eighth International Conference on Internet Monitoring and Protection (ICIMP', Citeseer, pp. 14–19.

Wang, C., Chow, S. S., Wang, Q., Ren, K. & Lou, W. (2013), 'Privacy-preserving public auditing for secure cloud storage', IEEE transactions on computers **62**(2), 362–375.

Waters, B. R., Balfanz, D., Durfee, G. &Smetters, D. K. (2004), Building an Encrypted and Searchable Audit Log., in 'NDSS', Vol. 4, pp. 5–6.

Yahyaoui, H., El-Qurna, J.&Almulla, M.(2020), 'Specification and recognition of service trust behaviors', Kuwait Journal of Science **47**(1).

Yavuz, A. A. (2018), 'Immutable authentication and integrity schemes for outsourced databases', IEEE Transactions on Dependable and Secure Computing **15**(1), 69–82.

Yavuz, A. A. & Ning, P. (2009), BAF: An efficient publicly verifiable secure audit logging scheme for distributed systems, in 'Computer Security Applications Conference, 2009. ACSAC'09. Annual', IEEE, pp. 219–228.

Yavuz, A. A., Ning, P. & Reiter, M. K. (2012a), 'BAF and FI-BAF: Efficient and publicly verifiable cryptographic schemes for secure loggingin resource constrained systems', ACM Transactions on Informationand System Security (TISSEC **15**(2), 9.

Yavuz, A. A., Ning, P. & Reiter, M. K. (2012b), Efficient, compromise resilient and append-only cryptographic schemes for secure audit logging,in 'International Conference on Financial Cryptography and Data Security', Springer, pp. 148–163.

Zawoad, S., Dutta, A. K. & Hasan, R. (2013), SecLaaS: Secure logging-as-a-servicefor cloud forensics, in 'Proceedings of the 8th ACM SIGSAC symposium on Information,computer and communications security', ACM, pp. 219–230.

Zeng, L., Xiao, Y. & Chen, H. (2015), Linux auditing: overhead and adaptation, in 'Communications (ICC), 2015 IEEE International Conference on', IEEE, pp. 7168–7173.

Submitted: 23/9/2020
Revised: 12/12/2020
Accepted: 16/12/2020
DOI: 10.48129/kjs.v48i3.10624