2011 IEEE World Congress on Services

TrustCloud: A Framework for Accountability and Trust in Cloud Computing

Ryan K L Ko [1], Peter Jagadpramana [1], Miranda Mowbray [2], Siani Pearson [2],
Markus Kirchberg [1], Qianhui Liang [1], Bu Sung Lee [1]

[1] Cloud & Security Lab
Hewlett-Packard Laboratories, Singapore
{ryan.ko | peter.jagadpramana | markus.kirchberg |
qianhui.liang | francis.lee}@hp.com

[2] Cloud & Security Lab
Hewlett-Packard Laboratories, Bristol
{miranda.mowbray | siani.pearson}@hp.com

Abstract— **The key barrier to widespread uptake of cloud computing is the lack of trust in clouds by potential customers. While preventive controls for security and privacy are actively researched, there is still little focus on detective controls related to cloud accountability and auditability. The complexity resulting from large-scale virtualization and data distribution carried out in current clouds has revealed an urgent research agenda for cloud accountability, as has the shift in focus of customer concerns from servers to data. This paper discusses key issues and challenges in achieving a trusted cloud through the use of detective controls, and presents the TrustCloud framework, which addresses accountability in cloud computing via technical and policy-based approaches.**

Keywords- trust in cloud computing, logging, auditability, accountability, data provenance, continuous auditing and monitoring, governance.

I. Introduction

Cloud computing requires companies and individuals to transfer control of computing resources to cloud service providers (CSPs). Such transfers naturally pose concerns for end-users. A 2010 survey by Fujitsu Research Institute [1] found that 88% of potential cloud consumers are worried about *who* has access to their data, and demanded more awareness of *what* goes on in the backend physical server. Such surveys demonstrate the urgency for practitioners and researchers to quickly address obstacles to trust.

While risks can be mitigated via preventive measures for privacy and security (e.g. encryption, access control based on ID profiling, etc), they are not enough. There is a need to complement such measures with equally important measures that promote transparency, governance and accountability of the CSPs. This was also identified by the European Network and Information Security Agency (ENISA)'s cloud computing risk assessment report [2], which states that the 'loss of governance' is one of the top risks of cloud computing, especially Infrastructures as a Service (IaaS).

Despite auditability being a crucial component of improving trust, current prominent providers *(e.g. Amazon EC2/S3, Microsoft Azure)* are still not providing full transparency and capabilities for tracking and auditing of file access history and data provenance of the physical and virtual servers utilized [1]. Currently, users can at best monitor the virtual hardware performance metrics and system event logs of the cloud services engaged. The cloud computing research community, particularly the Cloud Security Alliance, has recognized this. In its *Top Threats to Cloud Computing Report (Ver.1.0)* [3], it listed seven top threats to cloud computing:

1. Abuse and nefarious use of cloud computing
2. Insecure application programming interfaces
3. Malicious insiders
4. Shared technology vulnerabilities
5. Data loss or leakages
6. Account, service and traffic hijacking
7. Unknown risk profile.

Methods increasing the accountability and auditability of CSPs (e.g. tracking of file access histories) will empower service providers and users to reduce five of the above seven threats: 1,2, 3, 5 and 7. In this paper we propose a framework that addresses trust in cloud providers from accountability and auditability perspectives, via data-centric and file-centric logging. Using the abstraction layers defined in the framework, we identify and list key research issues which we discuss throughout the paper.

II. Trust In Cloud Computing

While there is no universally accepted definition of trust in cloud computing, it is important to clarify its components and meaning. In dictionaries, *trust* is generally related to *"levels of confidence in something or someone"*. Hence we can view trust in the cloud as the customers' **level of confidence in using the cloud**, and try to increase this by mitigating technical and psychological barriers to using cloud services. Further analysis of definitions of trust in cloud computing is found in [4].

A. Components of Trust in Cloud Computing

To best mitigate barriers to confidence, we need to understand the main components affecting cloud trust:

1) ***Security -*** Mechanisms (e.g. encryption) which make it difficult or uneconomical for an unauthorised person to access some information.

2) ***Privacy -*** Protection against the exposure or leakage of personal or confidential data (e.g. personally identifiable information (PII)).

3) ***Accountability*** – Defined in [5] as the obligation and/ or willingness to demonstrate and take responsibility for performance in light of agreed-upon expectations. Accountability goes beyond responsibility by obligating an organization to be answerable for its actions.

4) ***Auditability –*** The relative ease of auditing a system or an environment. Poor auditability means that the system has poorly-maintained (or non-existent) records and systems that enable efficient auditing of processes within the cloud.

978-0-7695-4461-8/11 $26.00 © 2011 IEEE
DOI 10.1109/SERVICES.2011.91

IEEE computer society

Auditability is also an enabler of (retrospective) accountability: It allows an action to be reviewed against a pre-determined policy to decide if the action was compliant, and, if it was not, to hold accountable the person or organization responsible for the action.

B. Preventive versus Detective Controls

Trust components can be also classified as ***Preventive Controls*** or ***Detective Controls***. ***Preventive controls*** mitigate the occurrence of an action from continuing or taking place at all (e.g. firewalls). ***Detective controls*** are used to identify the occurrence of a privacy or security risk that goes against the privacy or security policies and procedures (e.g. intrusion detection systems, or security audit trails, logs and analysis tools). In addition, there are *corrective controls*, which are used to fix an undesired result that has already occurred. This paper focuses on detective controls for cloud computing.

Despite the lack of direct ability to stop irregularities from occurring, detective controls act as psychological obstacles to breaching policies in the cloud, and also serve as a record for post-mortem investigations should any non-compliance occur. They act as in a similar way as speed cameras do for traffic control: cameras deter law-abiding citizens from speeding, but their presence cannot prevent speeding from taking place. Detective controls hence complement preventive controls. A combination of both is usually required for reasonable protection.

III. Complexities Introduced in Cloud Computing

Compared to traditional server architectures, end-users' focus of monitoring and accountability is shifting from a *server-health* perspective to a *user's data* perspective as companies move to the public cloud. Companies which used to own in-house servers are no longer concerned about the health of CSP servers they do not own. Instead, they are more concerned about the integrity and safety of their data deposited into the hands of CSPs. On the other hand, with cloud computing's elasticity [6, 7] introduces new complexities in the area of accountability.

A. Challenges Introduced by Virtualisation

1) Tracking of virtual-to-physical mapping & vice versa

Large-scale virtualization by CSPs allows higher resource utilization, and adaptation to peaks and troughs in users' demand for computation and storage. However, the addition of virtualized layers also means that accountability requires the identification of events not only on the virtual server, but also the physical server. Currently, there are only tools (e.g. HyTrust [8]) which are able to log virtual-level logs and system health monitoring tools for Virtual Machines (VMs). There is still a lack of transparency of (1) linkages between virtual and physical servers, (2) relationships between virtual and physical server locations, and (3) how files are written into both virtual and physical memory addresses. Such information is currently not available as a single-point-of-view for the customers.

2) Multiple operating system environments to track

Many different operating systems are available for VMs, and this potentially introduces the need to manage the logging of machines in the cloud which uses a large number of different operating systems. Enforcing a single operating system for all VMs would solve this issue, but it would make the provider less competitive.

B. Logging from Operating System Perspective versus Logging from File-Centric Perspective

Current tools focus on logging from the systems perspective, but few emphasize the *file-centric perspective*. By the *file-centric perspective*, we mean that we need to trace data and files from the time they are created to the time they are destroyed. When we log from a file-centric perspective, we view data and information independent from the environmental constraints. This in fact is reflective of the very elastic nature of cloud computing. With the transfer of control of data to providers, these providers can ease the minds of consumers by providing them with capabilities for tracking their data.

C. Scale, Scope and Size of Logging

The elasticity of cloud computing also increases the need for efficient large-scale logging. By efficient, we mean that the exponential increase in log size has to be manageable, without wiping out the memory of servers hosting the cloud logging features. Detailed logs may reveal information that is private or sensitive, and there need to be adequate controls over *who* gets access to this information, and for *what* purposes. We need policies that can help to clearly define the areas which loggers are assigned to log in. For example, a service provider may label its own network as a *safe zone*, while its suppliers or mirror sites *trusted zones*, and any other network outside of these are labeled as *unsafe zones*. Zonal planning will greatly reduce the complexities of network data transfer tracing within a cloud.

D. Live and Dynamic Systems

While there are proposals for adoption of provenance-aware mechanisms (that allow tracing back the source or creator of data) in cloud computing, these proposals are unable to address all challenges, as cloud systems are live and dynamic in nature. Provenance techniques propose reports (e.g. audit trails) as the key to forensic investigations. However in practice, a snapshot of a running, or "live" system such as the VMs turned on within a cloud only represents the situation at a specific instant in time and cannot be reproduced in a later time-frame. As a result, with a live system, data taken from a probe at one instant of time will be different from data from another probe say 15 minutes later [9]. This means that cloud accountability demands complex ***real-time*** accountability, where key suspected events are captured almost instantaneously.

IV. The TrustCloud Framework

Our team is currently focusing on addressing cloud accountability from all aspects, via five abstraction layers in the TrustCloud framework. In this section, we also list a large number of issues identified by our team.

A. TrustCloud Accountability Abstraction Layers

Logs range from *system-level* logs to *workflow-level* audit trail transactional logs. There needs to be a clear definition of abstraction layers to reduce ambiguity and increase research focus and impact. We propose the TrustCloud framework, which consists of the following **layers of accountability**:

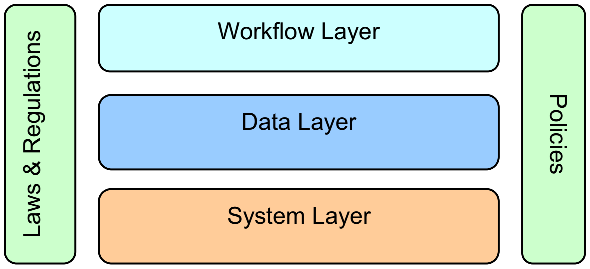

Figure 1. Abstraction Layers of Accountability in Cloud Computing

Figure 1 shows the abstraction layers for the type of logs needed for an accountable cloud. It extends the layers in our previous work [10], which stipulated three basic layers: workflow, data and system layers. It is important to note that the focus is on the abstraction layers of logs and not on architectural layers. Hence, the TrustCloud framework is independent of virtual or physical environments, and consequently, the current cloud layers of IaaS, PaaS and SaaS. Such explicit definition of layers allows us to efficiently identify the areas of their application and their focus areas. At a glance, the five layers look deceptively simple, but the problem is more complex than it looks.

Each layer has a slightly different focus, and different set of sub-components for each context. Our model simplifies the problem and makes accountability more achievable. The usefulness of abstraction layers is also analogous to OSI and TCP/IP networking layers. Let us now discuss the research issues, scope and scale of each TrustCloud framework layer:

B. System Layer

The lowest TrustCloud layer is the system layer. The system layer tracks data containers by performing **file-centric logging** within the following three components:

1) Operating Systems (OS)

OS system and event logs are the most common type of logs associated with cloud computing at the moment. However, these logs are not the main contributing factor to accountability of **data** in the cloud, but a supporting factor. This is because in traditional physical server environments housed within companies, the emphasis was on server health, system status and ensuring uptime, as server resources are limited and expensive to maintain. In cloud computing, resources are relatively inexpensive and appear to end-users as though they were unlimited. OS logs, while important, are no longer the top concern of customers.

2) File Systems

Even though the file system is technically part of the OS, we explicitly include it as a major component in a file-centric system layer. This is because, in order to know, trace and record the exact file life cycles, we often have to track system read/write calls to the file system. From the system read/write calls, we can also extract the files' virtual and physical memory locations, providing more information for further forensics. The file-centric perspective [11] is also the area which is less emphasized by current tools.

3) Cloud's Internal Network

As clouds are vast networks of physical and virtual servers over a large number of locations, we need to also monitor network logs within the cloud. Network logs [12, 13] are logs specific to data being sent and received over the network. Our team is currently working on techniques which perform logging, and tracing of file life cycles (i.e. *creation, modification, duplication* and *destruction*) within clouds.

C. Data Layer

The data layer supports the data abstraction and facilitates **data-centric logging** through the following components:

1) Provenance Logger

To enable reasoning about the origins, collection or creation, evolution, and use of data, it is essential to track the history of data, i.e., its provenance. Provenance information is often viewed as the foundation for any reasonable model of privacy and trust. It enables validation of processes involved in generating/obtaining the data and the detection of unusual behavior. We also need to detect attempts to falsify provenance data; to protect data owners as well as data providers from exposing sensitive, important information indirectly through provenance logs; and to enable efficient querying of provenance data. Cloud computing-based provenance logging must fulfill the following criteria: (1) be secure and privacy-aware (to ensure that the logs themselves cannot be tempered with or be a source for knowledge inference); (2) be (eventually) consistent and complete (similar to the ACID properties known from database transaction processing); (3) be transparent/non-invasive; (4) be scalable, e.g. avoid exponential explosion of provenance data through application of summarization techniques; (5) be persistent over the long term; (6) allow for multiple tailored views (to permit access based on roles with different access privileges); and (7) be efficiently accessible.

2) Consistency Logger

While current cloud providers typically support a weaker notion of consistency, i.e., eventual consistency, it is important to have mechanisms to allow for rollback, recovery, replay, backup, and restoring of data. Such functionality is usually enabled by using operational and/or transactional logs, which assist with ensuring atomicity, consistency, and durability properties. Logs have also been proven useful for monitoring operational anomalies. While these concepts are well established in the database domain, cloud computing's characteristics such as eventual consistency, "unlimited" scale, and multi-tenancy pose new challenges. In addition, secure, privacy-aware mechanisms must be devised not only for consistency logs but also for their backups.

D. Workflow Layer

The workflow layer focuses on **audit trails and audit-related data found in the software services in the cloud.**

1) Governance in the Cloud

When cloud computing experiences an increase in uptake and usage, there will be mandated needs for auditability, proper prevention and tracking of fraudulent activities, irregularities and control loopholes in the business processes. The workflow layer of the TrustCloud framework, together with the policy layer, is concerned with how clouds can achieve high auditability via compliance to regulations such as Sarbanes-Oxley (SOX) [14] and Health and Human Services Health Insurance Portability and Accountability Act (HIPAA) (e.g. Title II: Preventing Healthcare Fraud and Abuse) regulations [15], and/ or benchmarking against information security standards such as the ISO 27000 suite.

2) *Automated Continuous Auditing*

With the promise of high performance computing power from cloud architectures, we foresee automated auditing of financial and business process transactions in the cloud. Auditability is a prerequisite for such a step. However, achieving auditability via methods such as continuous auditing [16] within a highly virtualized environment is a very difficult and complex task. There needs to be consideration not only of the auditing of business logic and control flows, but also of the applications.

3) *Patch Management Auditing*

There is also a need for auditing of the management of virtual machine image bug fixes, patching and upgrades in a cloud environment [17, 18]. The scale of patching and deployment within the cloud environment is massive, and the associated logs need to be highly auditable for proper troubleshooting, playbacks and accountability of the technical staff performing these activities.

4) *Accountability of Services*

With cloud computing, the source of services may or may not be trustworthy, which presents a major problem in cloud computing. Some services may be malicious (e.g. manipulate data passing through) and violate contractual agreements. We believe that logging can help achieve accountability of services. Logging should assist with addressing the following concerns about a service component:

a) *Input or pre-processing*, whether the component takes in adequate input to perform the required function.

b) *Processing*, whether the component is designed to do what is expected. Is there any extra and unexpected processing that occurs during the production of the requested result?

c) *Post processing*, whether the component properly disposes of the input and intermediary results of the processing.

E. Policy, Law and Regulations

Policies and laws require information to be logged on what data items are processed, accessed, stored or transmitted. They may also require information on *why, when, where, how* and *by whom* this processing takes place.

What: Data classification is important, as in general there will be different policies and legal rules affecting different classes of data items. Possible classes include non-PII data, anonymised data, pseudonymised data, PII, sensitive PII, and PCI-regulated data. When new data is created (either by a user, or as the result of automated copying or processing of already-existing data) this creation may need to be logged together with its classification and/or the policies associated.

Why: The purpose of a data processing action, and the purposes for which the processing of a given data item (e.g. PII) is permitted, may need to be recorded.

When: Logs usually include timestamps. Timing information is also necessary for compliance to laws and policies concerned with data retention: it is necessary to have a data retention and destruction plan for all data storage systems. Timing considerations may also reduce the information that needs to be recorded, as transient data that is only stored for the purpose of the current transaction and then deleted has minimal privacy implications.

Where: Geographical location matters from a legal point of view – different laws may apply depending on where information exists, and there are some restrictions on trans-border data flows. It can be difficult to ascertain within the cloud where data is, and there may be multiple copies. So the physical location of storage and the occurrence of cross-border data transfers may need to be recorded.

How: Some laws and policies restrict how data is handled. For example, the processing of PCI-regulated data may require encryption and other safeguards. Information on how such data has been handled therefore needs to be recorded for auditability.

Who: Policies may restrict access to a data item to a particular set of authorized users, identified either as individuals or by role. There is a need to record the corporate identity of partners or CSPs to which data is transmitted, as part of due diligence, and to assist actions required by policies if a provider goes out of business or is acquired, or has a data breach.

V. RELATED WORK

A. Governance, Risk Management and Compliance (GRC) Stack of the Cloud Security Alliance (CSA)

The Cloud Security Alliance (CSA), comprised of many subject matter experts from academia and leading organizations, is a non-profit organization formed to promote best practices for providing security assurance within Cloud Computing, and provide education on Cloud Computing. Two projects from the CSA's Governance, Risk Management and Compliance (GRC) Stack are relevant: *CloudAudit* [19] and the *Trusted Cloud Initiative* [20].

B. HP Labs – Cloud and Security Lab

Pearson and Mowbray have done research on technical and procedural methods for promoting cloud privacy [21, 22]. Recently, Ko, Lee and Pearson established the case for accountability in [10], via a short paper covering scenarios and concerns of accountability within the cloud.

C. University of Pennsylvania/ Max Planck Institute for Software Systems

Haeberlen et al. were one of the first researchers to call for awareness in an accountable cloud [23]. In [23], they assumed a primitive *AUDIT* with considerations of *agreement*, *service* and *timestamps*. However, *AUDIT* did not have a clear explanation of the scope, scale, phases and layers of abstraction of accountability. It is our aim to complement their work. They also proposed an approach for accountable VMs [24], and discussed a case study on the application to detect cheats in an online multi-player game

Counterstrike. This non-cloud based game was not a practical business scenario for accountability, and did not address the needs of logging virtual-to-physical mapping.

D. HyTrust Appliance [8]

Recently, HyTrust, a startup focusing on cloud auditing and accountability, has released a hypervisor consolidated log report and policy enforcement tool for VM accountability management. HyTrust Appliance addresses the *System layer* (recall Section IV) of cloud accountability. It focuses on the virtual layers and does not log virtual-to-physical complexities. It also views accountability from a system perspective and not a file-centric perspective.

E. Accountability of Services by CSIRO

Chen and Wang of CSIRO currently have a team looking at “accountability as a service” for the cloud [25, 26]. Their work presented a prototype which enforces accountability of service providers whose services are deployed in the cloud. This is achieved by making the service providers responsible for faulty services and a technique which allows identification of the cause of faults in binding Web services.

F. Provenance in Clouds

Muniswamy-Reddy et al. [27] discuss the main challenges of provenance adoption for cloud computing and suggest four properties (data coupling, multi-object casual ordering, data-independent persistence, and efficient querying) that make provenance systems truly useful. Secure provenance [28] and privacy-aware provenance [29] have also been proposed for cloud computing systems, as provenance information may contain or expose sensitive, confidential or proprietary information directly or indirectly.

VI. Concluding Remarks

In this paper, we establish the urgent need for research in cloud accountability. We propose detective rather than preventive approaches to increasing accountability. Detective approaches complement preventive approaches as they are non-invasive, and enable the investigation not only of external risks, but also risks from within the CSP. With the shift in end-users’ concerns from *system health and performance* to the *integrity and accountability of data* stored in the cloud, we require a file-centric perspective, on top of the usual system-centric perspective for logging. Using the abstraction layers defined via the TrustCloud framework, we were able to list several cloud accountability issues previously not mentioned in cloud computing literature.

We intend to develop a system based on the TrustCloud framework that gives cloud users a single point of view for accountability of the CSP. We are currently researching and developing solutions for each accountability layer, with one example being a logging mechanism for the system layer.

References

[1] Fujitsu Research Institute. (2010). *Personal data in the cloud: A global survey of consumer attitudes*. Available: http://www.fujitsu.com/downloads/SOL/fai/reports/fujitsu_personal-data-in-the-cloud.pdf

[2] D. Catteddu and G. Hogben, "Cloud Computing Risk Assessment," European Network and Information Security Agency (ENISA) 2009.

[3] Cloud Security Alliance, "Top Threats to to Cloud Computing Report (Ver.1.0)," 2010.

[4] S. Pearson and A. Benameur, "Privacy, Security and Trust Issues Arising from Cloud Computing," in *The 2nd International Conference on Cloud Computing 2010*, Indiana, USA, 2010, pp. 693-702.

[5] US House of Representatives, "The Best Practices Act of 2010 and Other Privacy Legislation," 2010.

[6] M. Armbrust*, et al.*, "A view of cloud computing," *Communications of the ACM,* vol. 53, pp. 50-58, 2010.

[7] A. Baldwin, S. Shiu, and Y. Beres, "Auditing in shared distributed virtualized environments," *HP Technical Reports,* 2008.

[8] HyTrust. (2010). *HyTrust Appliance*. Available: http://www.hytrust.com/product/overview/

[9] J. Shende. (2010). *Live Forensics and the Cloud - Part 1*. Available: http://cloudcomputing.sys-con.com/node/1547944

[10] R. K. L. Ko, B. S. Lee, and S. Pearson, "Towards Achieving Accountability, Auditability and Trust in Cloud Computing," presented at the International workshop on Cloud Computing: Architecture, Algorithms and Applications (CloudComp2011), Kochi, India, 2011.

[11] M. Rosenblum and J. Ousterhout, "The design and implementation of a log-structured file system," *ACM Transactions on Computer Systems (TOCS),* vol. 10, pp. 26-52, 1992.

[12] A. Slagell, J. Wang, and W. Yurcik, "Network log anonymization: Application of crypto-pan to cisco netflows," 2004.

[13] A. Slagell and W. Yurcik, "Sharing computer network logs for security and privacy: A motivation for new methodologies of anonymization," 2006, pp. 80-89.

[14] Sarbanes-Oxley Act, "Public Law No. 107-204," in *Washington, DC: Government Printing Office* vol. 107th US Congress ed, 2002.

[15] *Health Insurance Portability and Accountability Act (HIPAA) of 1996 (P.L. 104-191)*, 1996.

[16] Z. Rezaee*, et al.*, "Continuous auditing: Building automated auditing capability," *Auditing,* vol. 21, pp. 147-164, 2002.

[17] W. Z. P. Ning*, et al.*, "Always Up-to-date–Scalable Offline Patching of VM Images in a Compute Cloud," 2010.

[18] J. Wei*, et al.*, "Managing security of virtual machine images in a cloud environment," 2009, pp. 91-96.

[19] Cloud Security Alliance. (2010). *CloudAudit (A6 - The Automated Audit, Assertion, Assessment, and Assurance API)* Available: http://cloudaudit.org/

[20] Cloud Security Alliance. (2010). *Trusted Cloud Initiative*. Available: http://www.cloudsecurityalliance.org/trustedcloud.html

[21] M. Mowbray, S. Pearson, and Y. Shen, "Enhancing privacy in cloud computing via policy-based obfuscation," *The Journal of Supercomputing,* pp. 1-25, 2010.

[22] S. Pearson and A. Charlesworth, "Accountability as a way forward for privacy protection in the cloud," *Cloud Computing,* pp. 131-144, 2009.

[23] A. Haeberlen, "A case for the accountable cloud," *ACM SIGOPS Operating Systems Review,* vol. 44, pp. 52-57, 2010.

[24] A. Haeberlen*, et al.*, "Accountable virtual machines," *9th OSDI,* 2010.

[25] S. Chen and C. Wang, "Accountability as a Service for the Cloud: From Concept to Implementation with BPEL," in *6th IEEE World Congress on Services (SERVICES-1)*, 2010, pp. 91-98.

[26] J. Yao*, et al.*, "Accountability as a Service for the Cloud," in *IEEE Service Computing Conference 2010 (SCC 2010)*, 2010, pp. 81-88.

[27] K. K. Muniswamy-Reddy, P. Macko, and M. Seltzer, "Provenance for the Cloud," in *Proceedings of the 8th USENIX Conference on File and Storage Technologies*, 2010, pp. 197-210.

[28] R. Lu*, et al.*, "Secure provenance: the essential of bread and butter of data forensics in cloud computing," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, 2010, pp. 282-292.

[29] S. B. Davidson*, et al.*, "On provenance and privacy," in *Proceedings of the 14th International Conference on Database Theory (ICDT)*, 2011, pp. 3-10.