Noname manuscript No.
(will be inserted by the editor)

A Comprehensive Meta-Analysis of Cryptographic Security Mechanisms for Cloud Computing

Mehmet Sabır Kiraz

Received: date / Accepted: date

Abstract The concept of cloud computing offers measurable computational or information resources as a service over the Internet. The major motivation behind the cloud setup is economic benefits, because it assures the reduction in expenditure for operational and infrastructural purposes. To transform it into a reality there are some impediments and hurdles which are required to be tackled, most profound of which are cloud security, privacy and reliability. As the user data is revealed to the cloud, it departs the protection-sphere of the data owner. However, this brings new security and privacy concerns. This work focuses on the security and privacy of various cloud service and deployment models by spotlighting their major challenges. While the classical cryptography is an ancient discipline, modern cryptography, which has been mostly developed in the last few decades, is the subject of study who needs to implement strong security and privacy mechanisms in today's real-world scenarios. The technological solutions, short and long term research goals of the cloud security will be described and addressed using various classical as well as modern cryptographic mechanisms. This work explores the new directions in cloud computing security, while highlighting the correct selection of these fundamental technologies from cryptographic point of view.

Keywords Cloud Computing · Security · Privacy · Cryptographic Algorithms and Protocols

Mehmet Sabır Kiraz
Mathematical and Computational Sciences Labs
TÜBİTAK BİLGEM, Turkey
Tel.: +90-262-6481945
Fax: +90-262-6481000
E-mail: mehmet.kiraz@tubitak.gov.tr

1 Introduction

Even though cloud computing applications are not new, a whole new terminology is being introduced and many changes are being made by the industry as well as research community, highlighted by Gartner [38]. Whether realized them or not, cloud-based resources (data, software, storage, infrastructure, security) are part of our daily life, e.g. when we use an e-mail service such as Gmail, watch a movie on YouTube, store files to DropBox, or shop at Amazon.com. As a marketing strategy, each solution provider brands almost the same product with a different name. Larry Ellison, CEO of Oracle, describes computer industry as more fashion driven than women's fashion [57]. For example, Microsoft has named its platform service as Azure, Google has named a similar service as Google AppEngine. To resolve terminology and achieve interoperability issues National Institute of Standards and Technology (NIST) has formed a cloud computing program. In [102], NIST defines cloud computing as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". This cloud model is generally consists of 5 essential characteristics, 3 service models, and 4 deployment models (see Figure 1).

The 5 characteristics are "On-demand self-service", "Ubiquitous network access", "Resource pooling", "Rapid elasticity" and "Measured service". The 3 service models are "Cloud Software as a Service" (which uses cloud service providers' applications over a network like Saleforce's applications, Microsoft's 365-suite and Google's applications like

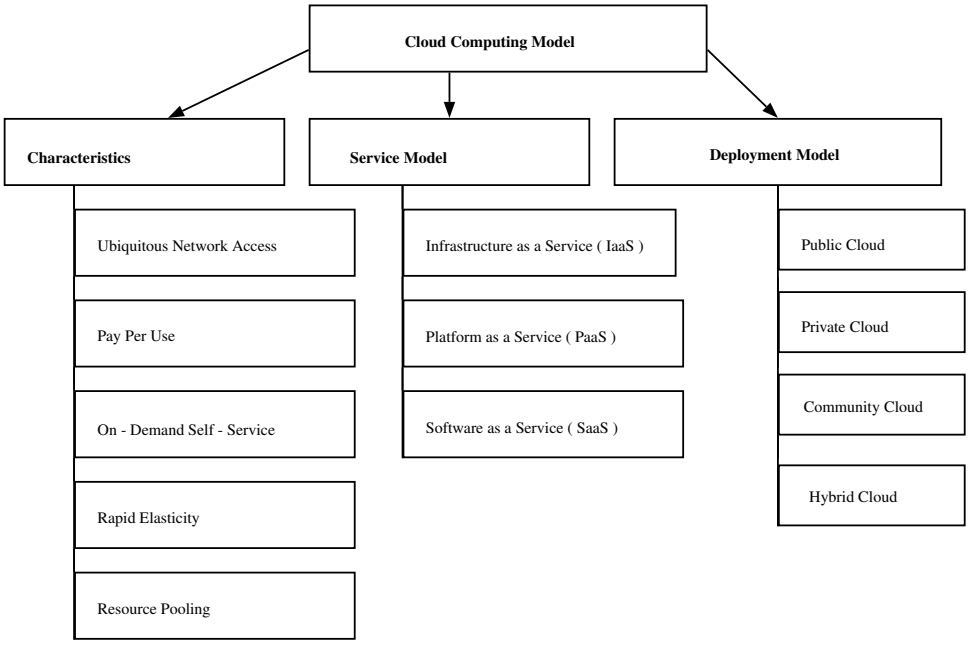

Fig. 1 Cloud Computing Model [102]

email, calendar), "Cloud Platform as a Service" (which deploys customer-created applications to the cloud like database and Java/PHP engine), and "Cloud Infrastructure as a Service" (which rents processing, network capacity, storage, and other fundamental computing resources). The four deployment models are "Private Cloud-Enterprise" (owned or leased), "Community Cloud-Shared Infrastructure" (for specific community), "Public Cloud" (sold to the public and constituting a mega scale infrastructure) and "Hybrid Cloud" (which is a composition of two or more clouds).

Many big companies such as Google, Apple, Microsoft, IBM, Amazon, Yahoo have launched their own cloud computing infrastructures, e.g. Amazon's Elastic Computing Cloud (EC2) and S3, IBM's Blue Cloud, Google's Google Apps, HP's CloudStart, iPhone apps, Microsoft's Azure, OpenStack and CloudStack. Public cloud distributes services to anyone on the Internet. On the other hand, a private cloud is a closed network that supplies hosted services to the users of this network. Furthermore, if a cloud service provider (CSP) utilizes public cloud resources to create their own private cloud it is called virtual private cloud. However, various issues are hampering the practical manifestation of this innovative concept; security being the most critical one and continues to be a top issue in the cloud computing model. Further definitions and terminology can be found in [102].

Big data is also another area of cloud that has its own set of challenges. Cloud Security Alliance (CSA) proposed the top ten security and privacy challenges for big data [47] and NIST also described security and privacy requirements for big data [9]. In [158, 124] surveys on the security and privacy concerns of various CSPs are offered. According to the Trusted Cloud Europe Report in 2014 [7], many sectors, including the general public sector, taxation and social security, health care and legal services, media and entertainment, financial services, national archiving and manufacturing/consumer have various information security concerns. YYYIn this respect, security is still being one of the main reasons discouraging users from employing cloud platforms because the loss of their sensitive information may lead to the loss of interest for enterprise organizations.YYY

Many researchers have been working on such areas during the previous decade, yet, one cannot

find a single source where most such research findings have been gathered [20]. Production of a single source, which gathers a variety of problems hindering implementations of the cloud concept, would really be an innovative one. Hence, while keeping major focus on security related issues in cloud computing as viewed by the original researchers during the past decade, we would also give a broad overview of cloud security mostly based on the use of cryptography. In the bargain, we also try to shed light on the potentials of modern cryptography in cloud computing and try to describe how it can be utilized in solving issues pertaining to implementation of "Cloud" (and considering all of them as a Cryptography-as-a-service model). Then elaborating the cryptographic aspect, we proceed further to describe new directions in cloud security by exploring certain hardware token scenarios, and make an effort to categorize and highlight the important ones by comparing advantages and disadvantages of various types of hardware tokens.

(a) Scope of the Paper

This paper discusses security issues of cloud computing and their broad solutions with respect to classical as well as modern cryptography. This paper aims at combining most such research carried out on these scenarios with suggested solutions that the reader can access in a single source of knowledge. Hence, most of the relevant research has been gathered in this paper in order to facilitate any individual interested in issues related to cloud computing security. We also make an effort to address certain new directions in cloud computing to the future of these concepts. Security is considered as the most important impediment hindering the adoption of cloud computing concept in practical use [134, 117, 76, 62, 20]. Therefore, an utmost effort has been made to highlight this issue by throwing light on the work of various researchers [52, 58, 19, 103].

It is fundamentally not secure to outsource sensitive information storage to anywhere, especially to outside of an organization. The trivial solution would be to use standard security technologies in-house before sending the data to the CSP (e.g., symmetric and asymmetric encryption algorithms such as AES, RSA; one way hash functions such as SHA families; digital signatures such as RSA, DSA; cryptographic protocols like SSL/TLS, VPN, IPSec). However, these classical solutions introduce the potential risk of system DBAs (database administrators) abusing their privileges because the systems and applications must possess the encryption keys. Although cloud computing has its various advantages and benefits over the current IT infrastructure it brings its own additional security threats, vulnerabilities and risks which have to be mitigated. Therefore, it is highly crucial to protect the sensitive information and systems in the cloud setting in order to ensure the privacy of its users/clients. We have also highlighted the importance of the use of modern cryptography to solve various security and privacy problems within the realm of cloud computing. While talking about new directions in cloud computing, we also focus on futuristic concepts such as token-based cloud computing (a type of Trusted Computing). The solutions provided in this paper are not merely the research of the author(s), but are attributed to a number of individuals who have carried out research in this topic and put forward their suggestions in order to solve various security issues in cloud computing.

(b) Roadmap

The paper is structured as follows. Section 2 throws light on major dilemmas in cloud computing, highlighting "Security" as the main obstacle in the implementation of the concept. Various categories of cloud security requirements are touched upon in Section 3, describing their detailed security needs. Certain contemporary solutions derived from classical as well as modern cryptography with some new directions in cloud computing have been described in Section 4. Section 5 discusses the further non-cryptographic security issues. Section 6 concludes the paper with a focus on the future of the cloud computing security.

2 The Dilemma with Cloud: Security as a Major Issue

Computation has changed radically from centralized to distributed systems, computing systems are returning back to centralized, virtualization technology in data centers which is called today as cloud computing [52]. Cloud computing essentially describes the development of existing and evolving technologies in a pervasive computing environment [14]. At the same time, there have been various challenges faced by the organizations who wish to implement these models. In particular, cloud security has become the most critical barrier to have a widespread usage of cloud computing. Consider the following questions to highlight some possible security concerns:

- YYYIs encryption of data or access to data more important?
- Can data be encrypted when stored in the cloud? Who holds the encryption keys?
- Who has access to the data? What are the access-control policies?
- Is data encrypted during transfer from the internal network to the cloud? What is the encryption algorithm?
- If you encrypt your entire data and send it to a CSP then how can it efficiently query the data without decryption? More generally, is it possible to enable computation over encrypted data?
- How does a CSP distribute the decryption keys to her dynamic clients? More specifically, how do they handle user revocation?
- Does the CSP have security breach investigation capabilities?
- How can you be sure that your data is not modified or deleted without your permission?
- If you want to delete your data in the cloud then how can you be sure that it is indeed deleted and cannot be recovered?
- Do performing operations in cloud lead to privacy issues?
- How can the CSP assure and provide the following services all in one: confidentiality, integrity, authentication, privacy, untraceability, delegated authentication, accountability, transparency, and access control?
- What is the data-backup process? Who has access to the backup data? Where is the backup data stored?
- What is the data-recovery process? How long does data recovery take? investigation process?
- Location of your data storage and processing is utmost important because each country has its own legal requirements, therefore how can you be sure that your data is stored on the locations that you prefer? YYY

YYY Furthermore, and most importantly, a well-known but often ignored fact is that if data is compromised then the CSP will be the only source of information for an investigation and auditing. YYY

YYY The cloud can provide exactly the same technologies as a traditional on-premise (non-cloud) infrastructure and the main difference is that each of these technologies is provided as a service. These services can be accessible over a cloud management interface layer, which provides access over Representational State Transfer (REST)/Simple Object Access Protocol (SOAP) Application Programming Interfaces (APIs) or a management console website. Therefore, in addition to the existing security issues for traditional setting such as access control, secure communication, data confidentiality, integrity, availability, and privacy cloud systems bring new cloud specific security challenges [20]. We refer to Table 1 to give an overview of a comparison between non-cloud (on-premise) and cloud systems from the security threat perspective. And going beyond that, with the 2013 disclosures of special NSA programs by Edward Snowden and existing real-life serious security issues like the Heartbleed, BEAST, POODLE, FREAK, Logjam, and DROWN attacks [61, 17, 50, 33, 15, 34, 25], we are now more concerned about the software and the hardware of the cloud systems than before. YYY

We emphasize that serious weaknesses have been found in many well-known cryptographic algorithms, cryptographic protocols and in their implementations including the underlying algorithms for key generation [27]. Therefore, developers of the cloud system must use cryptography in a cor-

	Security Risks & Threats	On-Premise (Non-Cloud) Systems	Cloud Based Systems
Confidentiality	Data Control & Privileges (Physical access, Credentials, Identity and Access Management)	Locally managed with own resources, full controlled under organization's responsibility. Breach remains on premises, more controllable, insider attacks are applicable, requires additional capital investment.	Since limited control and access on the data a trust to the CSP should be established. Data is distributed to many servers, Location independent resource pooling, Multiple clients affected by the shared services, Strong authentication and access controls are needed.
	Data Leakage (Secrecy, Privacy)	More easily managed as access to data is monitored and controlled	No control over the access data. Need to trust CSP. Data classification policies and processes needed. Service Legal Agreements (SLAs) between clients and CSPs are essential. Privacy preserving mechanisms also be provided.
	Development Environment	Easier to manage, easier to monitor, easier to train personnel on use	No control on source code accessibility and verifiability, disclosure to security bugs on the integrated development environments (e.g, no longer control what version of a software application that clients are going to use).
	Virtualization (Managing images, Monitoring virtual machines, Virtualized networking, Mobility, VM-Level, Malware, Availability)	On premises environment, no or limited effect by virtual machine vulnerabilities	Hypervisor is the main target of hackers. Hypervisor vulnerabilities can affect multi-tenants of the shared platform. Allocation and deallocation of memory, storage and other resources.
	Network security (APTs and Malicious outsiders, Protocols and Standards, Web Services, Web Technologies, Availability, Mobile Platforms, Perimeter Security, Spoofing, Denial of Service (DoS), DDos)	Organization's responsibility and it is expensive	CSP is responsible for maintenance. There are more system services and drivers which means more attack surface compared to traditional computing model. Secure the data with the state-of-the-art security technologies by expert security teams. Security patches/updates are updated continuously across all resources, therefore it has better security management.
Integrity	Tampering	Data are stored on promises under organization's responsibility	All data transmission is done through the internet thus may lead to loss of data.
	Integrity/Data Loss	Conventional mechanisms are used (e.g., digital signature, HMAC), No concern due to locality except insider attacks	Conventional mechanisms do not scale certain malleability needed (e.g., linear authenticators, short signatures, integrity auditing mechanisms, secure deletion).
Availability	Data Backup/Recovery	Not paying attention may lead to data loss.	Assured under Service Level Agrements, CSP is responsible.
	Multi-tenancy	Hard	Easy
	Disaster recovery	Hard	Easy
	Interoperability	Managed and processed by organization	SLA, and CSA, NIST, International Organization for Standardization (ISO) Regulations.
	Downtime risk/Availability (Network, Hardware, Power, and Sofware Failures)	Moderate & Expensive	Low
Trust Model	Reliability (Moving to the Cloud, Human Factor, Reputation, Auditability, Privacy, Anonymization)	Managed and processed by organization	Requires chain of trust agreement
	Compliance & Legality (Forensics, Acts, Legal Problems, Accountability, Governance)	Managed and processed by organization	Problem of the verification of authorization and authentication records, and also to check the compliances with predefined standards and policies. SLA, Cloud policies and controls, Data classification policies and processes.
Complexity	Storage	Increasing storage for an on-premise service likely requires an extra hardware, but that means clients have extra and unnecessary hardware if it is no longer needed. Furthermore, block or object (file) level deduplication can be performed on the client side (is not essential for an organization).	Cloud services are flexible and allows for increases and decreases as necessary (pay-as-you-go or on-demand usage service model). Security and privacy issues still exist with resource pooling where clients share resources, allowing CSP to distribute the storage cost savings to all of its clients (multi-tenancy). Also, deduplication decreases the storage cost significantly but also compromises secrecy & privacy, however it is essential for the CSP to minimize the the cost.
	Bandwidth/Communication	Managed and processed by organization (conventional mechanisms are used such as email, portable hard disk, and shared pooling)	Deduplication decreases the communication cost significantly but also compromises secrecy & privacy, however it is also essential for the CSP to minimize the the cost.

Table 1 Comparison: Traditional On-Premise (Non-Cloud) Systems vs Cloud Specific Threats

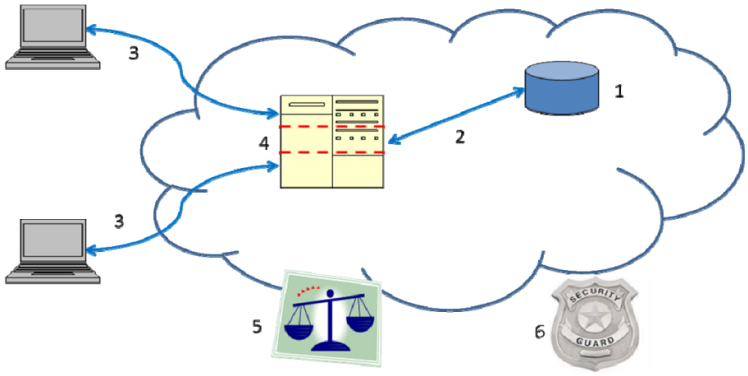

Fig. 2 Areas for security concerns in cloud computing: (1) Data-at-rest and Data-in-process, (2) Data-in-transit, (3) Authentication, (4) Separation between customers, (5) Cloud legal and regulatory issues and (6) Incident response [1].

rect manner and must have the ability to fix and correct weak/vulnerable cryptographic implementations. For example, developers need to know how to use high quality random number generation process which is the most crucial parts of an encryption system, instead of using weak random number generation functions like Rand() in C or C++, java.util.Random in Java, or System.Random in C# or VB.NET. On the other hand, kleptographic attacks (also called as cryptographic trojans) proposed by Young and Yung [154] can also be rather serious in the cloud setting. It is also crucial to consider kleptographic attacks/backdoors in the cloud security architecture by being suspicious to executable codes and checking the correctness of the source codes.

There are a number of problems/issues encountered which have been addressed by various suggested solutions, however, addressing various major issues has been cumbersome. In particular, the main areas of security concerns are (1) Data-at-rest and Data-in-process, (2) Data-in-transit, (3) Authentication, (4) Separation between customers, (5) Cloud legal and regulatory issues and, (6) Incident response (see Figure 2). In order to have a structural insight, a security framework of cloud computing should the cover following main categories: Security and risk management, compliance, identity management and access control, security and privacy data/information, security of applications/processes, network/server/end-user security, software and hardware security, and security of physical infrastructure.

In Figure 3, we see the Cloud Security Alliance model which shows the mapping cloud model to the security control and compliance check. This mapping can be used to analyze the gaps between cloud architecture and compliance framework, and the corresponding security control strategies of CSPs, customers or third parties. Note that modeling cloud, security, and compliance helps to decide whether the security risks of cloud computing are at acceptable level or they should be mitigated.

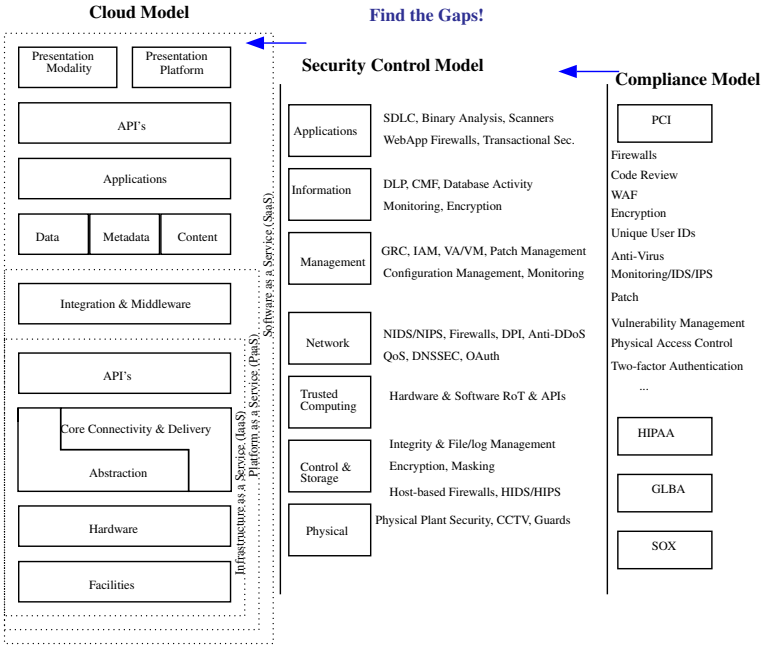

Fig. 3 Mapping the Cloud Model to the Security Control & Compliance [2]

(a) Potential Adversaries in Cloud Computing

Potential attackers in a typical cloud computing scenario can be either internal or external. Internal adversaries may consist of

- a cloud user
- employees inside the cloud user's organization with authorized access privileges
- employees inside the CSP with authorized access privileges
- the CSP itself
- a third party organization

The internal attackers may have motivation to learn other cloud users' passwords or other valuable information for bypassing authentication, gaining control of the virtual machines, logging the communication of all other cloud users, or misusing the access privileges to help unauthorized malicious users gain access to sensitive information.

External adversaries may be within the CSP which are not customers or third party provider organizations, and have no authorized access to cloud services, customer data or applications. Cloud computing can be vulnerable to various adversarial attacks from the Internet like any other open network systems. Any attacker who provides a valid credit card information can register to many CSPs and can easily launch passive or active attacks. Therefore, external attackers can easily do existing passive/active attacks like eavesdropping the network traffic, phishing legitimate users' credential, manipulating network traffic, and probing the cloud structure. Depending on the capabilities of an external attacker, it can launch severe attacks by taking the advantage of the knowledge of the system. Furthermore, the capabilities of attackers can also be on a different level: random (uses simple techniques), weak (uses publicly available tools), strong (organized, well-financed, skilled) and substantial (motivated, not easily detected, greater intelligence).

Before we move to the real issues, we note that cloud security is already an evolving research area of computer and network security, or more generally, information security. Information security basically comprises security rules and procedures, policies, technologies, and controls deployed to protect data, processes, and the associated infrastructure of cloud computing. As cloud computing is

achieving increased popularity, security and privacy concerns are being raised through adoption of this new model.

3 Cloud Security and Privacy Requirement Categories

After having gone through the description regarding dilemmas with the cloud leading to security issues, we move forward towards categorization of these security and privacy issues so as to analyze each one with an aim to shed light on the available solutions. Asset, threat, and vulnerability risk assessment matrices have to be implemented accordingly in order to compare/prioritize risks and to determine the correct security controls. Cloud security concerns can be classified into any number of categories, for example Gartner [38] classified into seven security issues while Cloud Security Alliance [2] identified thirteen domains of concern. All these categories can be aggregated into three main areas: 1) Security and Privacy, 2) Compliance, and 3) Legal or Contractual Issues. YYYIn this section, we will elaborate and discuss the information security requirements in the cloud setting and summarize the state-of-the-art existing and scalable solutions. YYYIn Table 2 we give an overview of threats, vulnerabilities, and countermeasures of cloud systems.YYY

(a) Identity Management

Identity Management basically consists of creation, management and removal of a digital identity. An identity management mechanism (IDM) is used to authenticate users and/or services based on their credentials and characteristics. A subtle issue of IDMs in cloud is interoperability and coherent security when different identity tokens and identity negotiation protocols are used. For example, existing password-based authentication solutions have limitations and significant security risks.

Most organizations have to manage the identity life cycle of their employees, their business partners, and their customers. These parties can change frequently and the relationship has to be updated, which requires an administrative action. Federated Identity Management provides the policies, processes and mechanisms to manage identity and trusted access to systems and to reduce identity management costs across organizations. This allows for reuse of users' identities and authentication methods across organizational boundaries, and ensures efficient user identity lifecycle management, compliance, and congruence of relevant user information between partner organizations without excessive administrative overhead. Most organizations or companies which are involved in the business have to formulate their privately owned IDM in order to have information access and computing resources control. Such organizations providing cloud services could either integrate Customer's IDM into their own infrastructure by the use of federation or Single-Sign-On (SSO) technology; or they would have to put forward their own identity management solutions [43, 137, 40]. For example,

- Amazon Web Services (AWS) EC2 uses Amazon Identity and Access Management (IAM) which allow users to create multiple accounts and manage the permissions for each of the new users based on the role and responsibility within the Amazon account. In this setting, a new user is an identity with unique security credentials which is used to access AWS services.
- Microsoft Azure uses Azure Platform AppFabric Access Control Service (ACS) to manage the security of user access. ACS integrates with Windows Identity Foundation (WIF) and supports for popular web identity providers including Google, Yahoo, and Facebook. ACS also supports Active Directory Federation Services 2.0 and OAuth 2.0. Note that an access token is provided by OAuth in order to grant access to a protected resource on behalf of the resource owner. Credentials of the resource owner like a password are never shared during OAuth.
- Rackspace uses client authentication called "Cloud Authentication Service" which is known as *Auth*. Auth allows each user to obtain an authentication token in order to use various services available in the cloud. Users authenticate with

Security Risks & Threats	Vulnerability	Incidents	Methodologies and Countermeasures
Identification & Authentication	Passive and active attacks, Unauthorized access to cloud resources and applications, Data-related vulnerabilities like SQL injection, Man-in-the-middle kinds of attacks, Phishing, Human accidents, Social engineering, Key loggers, Eavesdroppers, Malicious code	Compromise of secrecy & privacy Multiple clients affected by the shared services.	Username or other public information, Strong authentication, Multi-Factor Authentication, Forward Secrecy, Single-sign-on, ID Federation, Security policies, Monitoring, Auditing logs, Encryption methods, Firewalls, Intrusion Detection system, Antivirus, Client awareness. Biometrics, passwords, passphrase, token, or other private information, Security on the client-side (web security, malware), Cryptographic keys, Digital signatures [80, 110, 150, 13, 141, 96, 144, 109].
Authorization (Access Control, Privileges, Physical access)	Insider attacks, dictionary attacks, brute-force attacks, and spoofing at logon	Unauthorized users can access the data and can monitor and record all attempts made to access a system.	Consistent security policies and procedures, Separation of duties, Job rotation, Strong password policies, Strong authentication (forward secrecy), Intrusion detection and prevention, Penetration testing, Minimum necessary information provided, Monitoring, Mandatory and Discretionary Access Control, Attribute and Role based access control, Undeniable logging protocol, Multi-user access policies, Data access management, OAuth [101, 135, 120].
Account or service hijacking	Insecure Interfaces and APIs	An attacker can use the victim's account (by means of phishing attacks, fraud, exploitation of software vulnerabilities, reused credentials, and passwords) in order to get access to the target's resources.	Security policies, Strong authentication, Activity monitoring, Two-factor authentication, Forward secrecy, Single-sign-on.
Data scavenging	Unlimited allocation of resources	Data can be co-located with the data of unknown owners (competitors, or intruders) with a weak separation	Proof of location, SLA, Auditing, Tenant isolation [86, 35, 8, 108, 93].
Integrity	Correctness and completeness of data, SQL injection, Unpatched databases, Loss of encryption key, Privilege escalation, Malicious insiders	CSP could return incorrect and incomplete results in response to its clients' queries, Secure deletion is an issue in the cloud setting where data cannot be completely removed.	Integrity preserving mechanisms, Privacy-preserving public auditing, digital signatures, Replication of data, Error-correcting codes [145, 148, 88, 16].
Cloud Data Deduplication	Privacy leakage while performing block or file level deduplication on both the server and client sides.	Server-side deduplication compromises secrecy & privacy	Secure deduplication, Proof-of-ownership [28, 94, 157, 115].
Privacy leakage	Unpatched databases, Loss of encryption key, Privilege escalation, Malicious insiders	Data is often stored, processed, and transferred in clear plaintext. Therefore, data secrecy & privacy must be ensured during its journey.	Encryption (data in-transit & at-rest & in-process), Homomorphic encryption, Threshold cryptography, Digital signature, ID based encryption, Bring your own device (BYOD), Strong password, Strong key management, Privacy and integrity preserving mechanisms, Digital signatures, Attribute based Proxy Re-Encryption, Access Controls [138, 85, 99, 21, 141, 1, 78, 142, 45, 127, 139, 73, 65, 72, 51, 120, 16, 74, 146].
Virtualization (Rootkit in hypervisor and VM escape)	VM hopping, Malicious VM creation, Insecure VM migration, Sniffing/Spoofing virtual networks, VM Sprawl, Virtualization backup and recovery.	Complex Hypervisor code, Unrestricted allocation and deallocation of resources with VMs, a malicious VM image in a public repository, Lack of robust sniffer/tracking/firewalling tools for virtualized network, Cross-VM side channel attacks due to the sharing of physical resources (e.g., a single core CPU, cache), Possible covert channels in the co-location of VMs, Uncontrolled Migration (from one server to another server due to fault tolerance, load balance, or hardware maintenance), Uncontrolled VM snapshots (due to flexibility)	Strong isolation, Physical and functional hierarchical, IDS (Instruction Detection System) and IPS (Intrusion Prevention System), Reconfigurable distributed virtual machine, Hypervisor security, Migration should be included in the SLA [86, 80, 6, 8, 108, 89, 16, 120].
Continuous Availability (Data recovery and disaster recovery issues)	Nature disasters, Malicious insiders, Availability concerns	Possibility of data corruption or failure of any mission-critical functions [44].	Recovery planning, Access control in buildings, Disaster & Recovery planning, Backup strategies, Multi facility provisioning, Data dispersion, and Data replication should be included in the SLA [86, 112, 13, 8, 108], Fault Tolerance, Byzantine quorum protocols, Secret sharing, Erasure codes.
Network security (APTs and Malicious outsiders, Protocols and Standards, Web Services, Web Technologies, Mobile Platforms, Perimeter Security, Spoofing, Session hijacking, Distributed Denial of Service (DDoS))	Ensuring the continuous availability, Security & privacy leakage.	Services and applications could be inaccessible from remote locations in a cloud environment. Continuous availability of cloud services can be disrupted. DDoS and signature wrapping kinds of attacks could create data transmission risks in the cloud network. Invisible network created by virtual servers makes it difficult to monitor network traffic and performance.	Information security risk management framework, Monitoring, Provisioning, Intrusion detection systems, Isolation, Netflow, Access control list (ACL), Authentication, Security policies, Network security for virtual machines, Network security sandbox, CSPs offer services that can be accessed through APIs (SOAP, REST, or HTTP with XML/JSON). The security of the cloud depends upon the security of these interfaces. Cloud APIs are still immature due to its frequent update (e.g., a fixed bug can introduce another security hole in the application). [16, 140].
Web security (Key-loggers, Malware, Spyware, Bot network, Phishing, Virus, Spam, Bandwidth consumption)	Malicious attack on low-level security applications, Malicious attacks on insecure browsers, Potentially higher cost of real time monitoring, Encrypted traffic	Confidentiality, Integrity, and Availability can be compromised	Email server, Anti-virus, Anti-spam, Anti-phishing, Web filtering, Monitoring, Strong authentication (SSL/TLS/VPN), Forward secrecy, Access controls, Encryption methods, Web application scanners, Firewalls, DLP, Email security, Security policy [54, 70, 129, 97, 128, 60, 36, 159, 113, 91, 133, 100, 63].
Trust & Reliability (Moving to the Cloud, Human Factor, Reputation, Auditability, Anonymization)	Continuous availability, Security & privacy leakage	Important aspects of decision making for Internet applications fo depth and assurance of confidence, otherwise it may lead to privacy leakage. Trust can be established based on many properties including security, reliability, and availability.	Concrete trust model should be established, SLA [86, 13, 8, 108], Threshold Cryptography, Homomorphic Encryption, ID/Pairing Based Encryption, Attribute-Based Encryption, Secure Multi-Party Computation
Cloud Standards, Compliance, Interoperability	Security & Privacy & Availability concerns	Need to measure the security and privacy needs to demand from the cloud services and to verify they comply with their security and compliance requirements.	IEEE Cloud Computing Standard Study Group, ITU Cloud Computing Focus Group, CSA, NIST, ISO [75**?** , 77, 10].

Table 2 Threats, Vulnerabilities, and Possible Methodologies and Countermeasures

their credentials and can create/delete containers and objects within that account.

An ideal IAM should be able to protect private and sensitive information of users and processes. Designing and developing of a new robust authentication and identity management protocols and improving existing lines is one of the most critical requirements for cloud computing. Access control systems for cloud computing should be flexible enough to ensure dynamic, attribute or credential-based access requirements [110]. Some other technologies implement federated identity solutions (e.g., Security Assertion Markup Language (SAML) [107]- an open source implementation of the SAML standard called Shibboleth [125], authentication and attributes; Service Provisioning Markup Language (SPML) [105]; eXtensible Access Control Markup Language (XACML) [106]).

(b) Authorization and Access Management

Authorization and access management forms a broad category of services that is required to ensure security in the cloud. Access Management includes the authorization of access only to data that an entity needs to access to perform required duties efficiently and effectively. While authorization determines the user's right to access a certain resource, access management in general has the responsibility to enforce that users access to a given resource is managed with respect to the users credentials and attributes associated with the identity policies. Provisioning and deprovisioning are critical aspects of Access Management. Provisioning is the process of creating accounts that allow users to access appropriate systems and resources in the cloud. The goal of user provisioning is to streamline account creation and provide a consistent framework for providing access to end users. Deprovisioning is the process whereby a user account is disabled when the user no longer requires access. This may be due to users leaving an organization, transferring within an organization, a change in role, etc. In the cloud computing environment, deprovisioning refers to the termination or disabling of user accounts in cloud platforms, or those managed by the cloud-based IAM service.

(c) Authentication

Authentication is the process of verifying the credentials of an entity trying to access a protected resource (which can be seen as a defense mechanism for spoofing identity and denial of service). Authentication is the most critical part of cloud environments because the cloud is accessible to anyone over the Internet. Therefore, authentication must be done in a secure, trustworthy, and manageable manner. For accounts that require higher levels of security, multiple factor authentication may be required such as password (what you know), card/token (what you have) or biometric (what you are). It is also sometimes called strong authentication which requires the use of two or more of the three types of authentication factors. In the cloud platform, authentication services should include strong authentication mechanisms for validating the credentials.

Some cloud services (like dynamically personalized based on location, calendar, social network) may also require privacy to be taken into account. On one hand, the client should authenticate herself and use the services, and on the other hand, it must be ensured that the cloud cannot know her identity. At the same time, the system should be accountable so as to detect malicious actions. Anonymous authentication mechanisms validate the clients without learning any information about their identity.

Delegatable authentication can also be an effective mechanism for cloud which allows CSPs to delegate the client authentication to the identity providers. Hence, clients do not have to register for each website in this scenario. Namely, they only use their identifier which will be sufficient to validate the authentication through the identity provider.

(d) Trust

In general, CSPs have access and control the user data. Therefore, minimizing the trust placed on the

CSPs is one of the most desired requirements. Minimizing required trust can be achieved with advanced cryptographic techniques such as homomorphic encryption, secret sharing and threshold cryptosystems (in order to distribute the trust), bilinear pairing, secure multi-party computation and fully homomorphic encryption. While some of them are ready for deployment others still need long-term research.

(e) Physical and Personnel security

The organizations providing cloud services are required to ensure physical security of the hardware and personnel involved in the overall cloud set up. Moreover, they are also required to ensure that access to such machines and relevant customer data is restricted. Access to data belonging to customers, by various personnel involved in provisioning services, should be well.

(f) Virtualization, Hypervisors and Multi-Tenancy

A hypervisor, also called virtual machine manager (VMM), is one of many hardware-assisted virtualization techniques allowing multiple operating systems to run concurrently on a host computer. A CSP utilizes virtual machines (VMs) and hypervisors to separate users. Virtual machines (VM) are the key aspects of cloud computing where different users (called tenants) access the same physical hardware. However, a corrupted hypervisor can damage all systems that it hosts. One of the best proposed solutions is to utilize Trusted Platform Modules (TPM) which can provide hardware-based verification of hypervisor and VM integrity in order to guarantee network separation and security. However, there are various VM-attacks where malicious adversaries who are on the same physical hardware learn private data from other virtual machines on the same hardware (e.g., cache based side-channel attacks [155, 76, 46]. Moreover, a malicious adversary can gain access not only on a user's data or applications but also other users' data and applications by simply attacking multi-tenancy design or insecure APIs. This is considered one of the most serious security threats in cloud computing [48].

(g) Application Security

CSPs need to assure that their applications available as a service are indeed secure. This can be verified by implementation, test and acceptance procedures for applications codes. These procedures aim to detect a possible vulnerability and find out its point of origin in the application code. It is important to highlight that the requirements for the security of an application are different based on the cloud deployment models IaaS, PaaS and SaaS [3].

(h) Privacy

Privacy is an important requirement for cloud computing, both in terms of legal compliance and user trust. This is one of the major cloud issue which should be considered at every phase of designing a cloud architecture. There are many types of sensitive information: personal, corporate/governmental financial information, health, biometric information, religious or racial information, sexual orientation, job performance information, collections of surveillance, taking photos/videos in public places, usage data collected from various devices (computers, printers, smart phones), behavioral information (e.g., viewing digital content), visited websites, product usage history, IP addresses, RFID tags, and unique hardware identities.CSPs are required to assure that all critical data (personal as well as commercial data) are masked against malicious adversaries and that only authentic and authorized users have access to the data. Moreover, electronic identities and credentials must be protected in the cloud together with additional data or activity that the CSP collects or produces [69].

The concept of differential privacy [56] considers the case that no information of an individual should be leaked from the database that cannot be learned without access to it. Namely, differential privacy guarantees that the addition or removal of

a database item does not lead to leak the private data of an individual.

We would like to highlight that the key requirement for privacy is to ensure notice, openness, and transparency. Notices basically should include information regarding a user's privacy policies. More concretely, organizations should provide notices to the users regarding traceability, collection, usage and dissemination of personally identifiable information.

(i) Data Integrity

Data integrity requirement means that data should be correctly stored on the cloud server without any unexpected modification, and any violations (e.g., if data gets lost, modified or compromised) should be easily detected [84]. In general, integrity is solved using either message authentication codes (MACs) or electronic signatures. However, the private key should be kept secret by the client, otherwise anyone who has the private key can easily modify the signed data without being detected. Data integrity of large files in cloud servers is not that efficient and easy to solve. This is because the utility computing of the resource are not run in-house, these services are outsourced to third parties and clients have no real guarantee that the CSP is performing the computation what they indeed claim. Therefore, it is important to assure the client of the integrity of the cloud data from any unauthorized modification. The trivial solution is to first download all the files and check their integrity but this requires high transmission cost. Hence, to guarantee the data integrity and to minimize the verification cost (in terms of communication complexity and storage) a stateless third party auditor is generally used as an assistance who checks the integrity for the cloud user by querying a random subset of the data portions [41, 149, 123].

Proof of data possession (or sometimes called proof of retrievability [83]) is a challenge-response protocol enabling a client to verify whether its data stored on the cloud is available and has not been modified without detection. It is also rather important to utilize stateless and semi-honest third party verifiers (i.e., auditors) [131, 12]. In general, existing schemes consist of four procedures: preprocess for signature generation by the client, challenges by the third party auditor, proof by the cloud and verification by the third party auditor. YYY-More concretely, a user outsources her data to the cloud and delegates a semi-honest third party auditor to perform the integrity checking process on behalf of the owner without leaking any information on the data [145, 148, 88] (see Figure 4 to see the public auditing model)YYY.

Secure cloud data storage services is a growing need which leads the cloud community to have an innovative solution for proof of data possession (e.g., Dropbox, OwnCloud, TeamDrive, Box, OneDrive (formerly SkyDrive), Google Drive, DepSky, and SugarSync). Nowadays, clients store (possibly substantially large) files to servers and want to be sure that they remain stored in their original form and no information is leaked to any party including CSPs [84]. Of course, for usability reasons the proposed solutions should also be efficient in terms of storage overhead, computation (including number of reads), and bandwidth. For example, DepSky system [31] which is a virtual cloud storage system to assure availability and confidentiality of data stored on combination of different clouds to form a cloud-of-clouds.The DepSky uses multi CSPs and each of them has own interface and cryptographic key which is distributed by a cryptographic secret sharing algorithm and erasure codes to prevent it from insider attackers. Multiclouds Database Model (MCDB) [18] is another solution to mitigate possible security threats on the data integrity and availability using multi CSPs.

(j) Secure Data Disposal

Secure data disposal is a critical risk from the customer point of view, because one cannot verify that data was indeed deleted and irrecoverable within the cloud [147]. If the cloud does not satisfy secure disposal mechanisms, then this may lead to sensitive information leakage. This vital obstacle prevents a certain deployment of industry and government.

(k) Guarantee of Image

Infrastructure-as-a-Service clouds provide virtual machines to clients to run their software on remote resources. Giving full control to CSPs the user is not actually sure that the correct image is executed, hence it may lead to security issues. Trusted Computing (TC) is a security mechanism for IaaS where execution of a virtual machine (VM) instance can be run securely. On the other hand, the virtual machines need to be up to date from the view of both the CSP and the customer side [114, 39].

(l) Secure Deduplication

Deduplication allows multiple uploads of the same content with storage space of a single upload. Namely, it ensures more optimal usage of the resources at the cloud side (e.g., Bitcasa, Dropbox). Unfortunately, encryption of data with classical cryptosystems cannot be deduplicated. Hence, current existing applications have either security or storage efficiency issues. Secure deduplication schemes aim to to reduce storage, communication, and computation overhead of the cloud storage server and clients while providing efficiency and security against malicious servers. There are four different deduplication categories: deduplication happens at the client side (i.e. before the upload) or at the server side, and whether deduplication happens at a block level or at a file level. The server-side deduplication which is run by the server is to check whether two files stored in the server are identical. That means, the client must upload a file first, then the server can determine whether it is a duplicate. In this scenario, clients are not aware of deduplication. Trivial solutions have been proposed for server-side deduplication. However, the client does not want to upload the file if it exists in the server, and that is why storage service providers employ the client-side deduplication. The client-side deduplication run by the client and the server interactively is to check whether the uploading file exists in the server. In this case, duplicate files are not uploaded. The most practical deduplication scheme considers the client side, because it also saves upload bandwidth. For these reasons, deduplication is a critical enabler for a number of popular and successful storage services that offer cheap, remote storage to the broad public by performing client-side deduplication, thus saving both the network bandwidth and storage costs. Indeed, data deduplication is arguably one of the main reasons why the prices for cloud storage and cloud backup services have dropped so sharply.

Message-locked encryption schemes aim to ensure secure deduplication [11, 29, 30, 92, 132]. Bellare, Keelveedhi, and Ristenpart in [29] formalized the convergent encryption which they called message-locked encryption. The basic idea of message-locked encryption algorithms is to encrypt the data under a symmetric key which is obtained as a function of the message (i.e., $sk = F(M)$). Therefore, same plaintexts will lead to same ciphertexts. Furthermore, the user only needs to store the key sk and her file M in cloud to be able to retrieve it and then decrypt it via the key sk. The scheme of Bellare *et al.* still provide security for only unpredictable messages which fails to achieve semantic security. Bellare *et al.* later presented a server-aided encryption mechanism for deduplicated storage, called DupLESS [29].

(m) Proof of Data Location

Users may not know where their data is physically located, which may result in an undesired leakage of users data to local authorities (due to local regulations) [143, 104].

(n) Data Security and Protection

Whenever user data leaves user's end-point, it starts travelling via a public network to be stored in the cloud. Therefore, the data can be intercepted and modified by (internal or external) malicious adversaries during transmission. To mitigate these kinds of attacks, strong encryption, access controls, and authentications are required. Most significant problems is data security in the field of cloud Security [122, 150].

In many existing cloud environments, important data, files and records are entrusted to a third

party, which enables data security to become the main security issue of cloud computing. Data loss and data leakage are both serious threats in a cloud scenario [4]. For instance, Google's amount of customer information was leaked in 2009. Security and privacy of data can be achieved with the use of cryptographic protocols and the underlying encryption and authentication mechanisms. Additionally, trusted platform modules (TPM) can also be used for migration of data between virtual machines and physical machines.

(o) Data Storage Security

Cloud storage services require strong cryptographic mechanism because in a cloud computing scenario the management of data and services may not be trustworthy and traditional security methods for securing an in-house data center cannot be directly adopted due to the loss of control over data (e.g., Dropbox, OwnCloud, TeamDrive, Box, OneDrive (formerly SkyDrive), Google Drive, and SugarSync). Therefore, long-term solutions have to be investigated for the verification of security and privacy properties.

Furthermore, cloud users frequently update their data stored in the cloud including insertion, deletion and modification. Therefore, to guarantee the security and the privacy of dynamic data and storage correctness is of at most importance. Here, it is crucial to highlight that verification of correct data storage has to be conducted in cloud without explicit knowledge of cloud data. Cloud computing is hosted, deployed and managed by cloud vendors running in a cooperated, simultaneous and distributed manner, therefore distributed cryptographic protocols play an important role in achieving a secure cloud data storage system as they are intended to assure security and privacy and avoid root of trust problems [138, 78, 142].

(p) Reliability

The clients outsource their personal or businesses data to CSPs. Therefore, the CSPs should be able to provide the reliability and trust to their clients that their data and running applications are indeed secure [26]. We note that an important component of reliability is to assure availability which is achieved with a good backup strategy. It is also crucial for the CSPs to meet Service Level Agreements which provide clients with clear information about controls and security of the CSP [5].

(q) Availability

CSPs need to give assurance to their customers that they shall regularize every access to customer data as well as applications. It is one of the most critical cloud properties in terms of usability. In 2014, Azure Virtual Machines experienced about 43 hours of downtime globally [44]. Furthermore, the analysis from CloudHarmony showed that Amazon Web Services (AWS) experienced fewer than 5 hours downtime in total for its storage services, and Elastic Cloud Computing (EC2) and Simple Storage Service (S3) had 35 outages. Googles cloud storage was down for less than 15 minutes in 2014. Therefore, preventing various attacks like denial of service (DoS) and ensuring robustness for the integrity of the overall system are of utmost importance. Backup, recovery schemes, fault tolerance, and replication techniques can be used to achieve the availability property.

(r) Legal and Regulatory Issues

CSPs as well as customers need to consider legal issues, such as Contracts and e-Discovery, and the related laws according to the country of origin of both parties. Users must have legal and regulatory experts in order to understand CSPs' policies and practices for resolving security issues (e.g., data security, auditing, data retention and deletion, trusted storage techniques).

4 Minimize the Threats: Modern Cryptography-as-a-Service

In this section, we focus on another dimension pertaining to cloud computing security which will shift

us from the classical concept of securing communication to the modern concept of securing computation. These directions will play a pivotal role in re-vitalization and maturity of the cloud computing concept in the future. We take a look at those specific directions in cloud computing, focusing on how these techniques can help us resolve various issues associated with cloud; and last but not least, we analyze certain proposed solutions under the umbrella of these new techniques.

Many researchers have proposed diverse solutions for resolving cloud security issues. However, in order to understand the problems surrounding the issue and have a realization of the impediments hindering the implementation of the cloud concept; first we need to divide the entire security issues into broad areas and then evaluate the suggested solutions so as to transform the cloud scenario into a reality.

Mathematical and cryptographic solutions are the most straightforward approaches for many of the given threats. Although they are useful mechanisms, they require careful implementation, because cryptography alone does not guarantee complete security. Many cryptographic primitives rely on some hard problems where for a probabilistic polynomial-time Turing machine it is computationally infeasible to solve secret values (e.g., *prime factorization of large numbers, intractability of the discrete logarithm, subset sum problem, learning with errors problem*). However, poor implementations or bad password choices can easily enable malicious adversaries to mount brute-force attacks that go through all possible combinations [130]. The brute-force attack is a growing threat because of its easier disclosure in terms of time complexity by either evolving technology (multi-core CPU, Graphics Processing Units (GPU) with high clock rates) and password cracking methods.

(a) Searchable Encryption in the Cloud

While transferring data to the cloud, privacy leakages may occur in existing database structures. For example, when a user wants to search for some data or files, he will query certain keywords from the cloud. The cloud service will execute the query and return the requested data or files to the user. This process may leak some information to a passive adversary about which files a user is looking for by just observing the query and the files returned to the user. To solve this privacy issue, private/encrypted search algorithms have been proposed [32, 21, 65, 72, 74, 146]. In a searchable encryption scheme, the user will send an encrypted query to the cloud so that the cloud will never learn what keywords the user is searching for. There are in fact three distinct security models for searchable encryption: 1) searching on private-key encrypted data, 2) searching on public-key encrypted data, and 3) single-database private information retrieval [49].

CryptDB can be given as a practical and efficient example which enables a large set of standard SQL queries to be executed in an encrypted database [111]. We note that the cloud has to execute the encrypted query on every data or file because otherwise the cloud may learn some information about the user's data in case some files are omitted to be taken into account. However, this will lead to performance issues because the cloud has to process all queries of each individual user over all data or files. Liu *et al.* [95] proposed a private search protocol called Cooperative Private Searching (COPS). This protocol aims to reduce the computation and communication costs while preserving privacy individual data. Their simulation results show that the computational costs can be reduced by 80% and bandwidth cost by 37%.

(b) Privacy Preserving Protocols Based Secure Multi-Party Computation

Secure multi-party computation (SMPC), also known as secure function evaluation, is one of the most fundamental problems in cryptography. In this scenario, two parties, possibly not trusting each other, wants to compute a joint function without revealing any information about their inputs except output of the function. In the ideal case, this problem can be easily achieved by using a trusted third party. "Millionaires' problem" is a typical example for SMPC problem: two parties are interested

	Generation (sec)	Evaluation (sec)	Communication (byte)
Oblivious Transfer	$19.73\pm 0.5\%$	$5.26 \pm 0.4\%$	1.7×10^8
	$1.1\pm 6\%$	$15.6 \pm 0.6\%$	1.7×10^8
Cut & Choose	$1.1\pm 0.8\%$	–	6.5×10^7
	–	$1.5 \pm 2\%$	6.5×10^7
Generation/ Evaluation	$24,400\pm 1\%$	$14,600\pm 3\%$	1.8×10^{13}
	$4,900\pm 1\%$	$14,700\pm 2\%$	1.8×10^{13}
Input Consistency	$0.6\pm 20\%$	–	8.5×10^6
	$0.4\pm 40\%$	$0.60\pm 20\%$	8.5×10^6
Total	$24,400\pm 1\%$	$14,600\pm 3\%$	1.8×10^{13}
	$4,900\pm 1\%$	$14,700\pm 2\%$	1.8×10^{13}

Table 3 The experiment of Kreuter *et al.* [90] for secure computation in the presence of malicious adversaries. The result of $(x, y) \rightarrow (\perp, EDT\text{-}4095(x, y))$ where $x, y \in \{0, 1\}^{4095}$ and EDT-4095 denotes 4095-bit edit distance circuit. The circuit has 5.9 billion gates and 2.4 billion of those are non-XOR. Each party is comprised of 256 cores in a cluster. The upper row in each stage is the computation time and the lower is the communication time .

to learn who is richer without revealing any information about their wealth. Secret Sharing scheme is a fundamental building block of many cryptographic protocols including secure-multi-party computation of general functions [87]. At the first stage, the input bits are distributed to all parties and then the parties evaluate the gates starting from the top to bottom (the circuit is consist of addition and multiplication gates where addition gate can be seen as addition modulo 2 and multiplication gate can be seen as multiplication modulo 2.).

Andrew Yao presented a solution to this problem in his seminal work, which is known as Yao's garbled circuit [152]: Any arbitrary computation $f(x, y)$ can be represented with primitive logical gates. One of the parties, say Bob, does the following for circuit creation. In the garbling phase each input wire or intermediate wire x, y, z is expanded to some t-tuple random vectors. The truth table values are encrypted by a symmetric algorithm and the outputs are permuted. The permuted truth table values is sent to circuit evaluator, which is Alice. Alice starts an Oblivious Transfer (OT) protocol for each input wire and obtains the inputs of Bob (in an OT protocol Alice has two values x_0, x_1 and Bob has $b \in \{0, 1\}$, at the end of OT Alice obtains x_b and Bob learns nothing.). After this phase Alice does not need any interaction. She evaluates the circuit and obtains her output. This setting is secure under semi-honest adversary model, i.e., each party behaves according to the protocol.

The details of the first implementation which includes a high level description language for circuit preparation and oblivious transfer primitives are introduced in Fairplay [98]. Their work starts with a high level function description language, SFDL, in which any function is described in a user friendly way. Later, this description is converted into gates. This gate representation is similar to register-transfer level (RTL) netlist of a digital circuit. After circuit creation the parties communicate with each other over TCP/IP network. Recently, various major optimizations have been proposed which improve either Yao's garbled circuit construction or the bandwidth efficiency (see [90] for the complete optimization solutions). With these recent improvements, Yao's garbled circuit approach is now believed to be one of the most feasible solutions for real-life secure computation problems. Fairplay [98] and his successors applied Yao's garbled circuit implementation to different privacy preserving algorithms [116, 51].

In one of the latest implementations of Yao's garbled circuits, Huang *et al.* [79] combined several optimizations together with their own enhancements. These include free XOR gate optimization, garbled circuit optimizations, oblivious transfer extension and pipelined circuit creation and evaluation. It was previously reported that when a large function is evaluated in garbled circuit memory overflows and very high run-time durations are observed. This was reported as a discouraging case

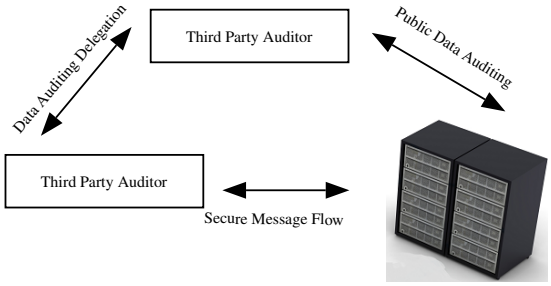

Fig. 4 Public Auditing

for secure function evaluation. Huang *et al.* created an implementation such that the protocol starts execution without having all the garbled circuit. Their implementation saves only intermediate level gates and the gates whose outputs will not be used again are erased from the memory. With this observation they made it possible to evaluate arbitrarily large functions with garbled circuits. More recently, Kreuter *et al.* in [90] improved previous implementation of Yao's garbled circuit schemes and give a summary of their implementation as illustrated in Table 3. They show that by utilizing state of the art optimization techniques and parallelizing almost all steps, evaluation of billion-gate circuits is practically feasible in the malicious model.

(c) Ensuring Data Integrity on the Cloud

To mitigate the security threats in cloud data storage, the subject of study focuses on either on single server scenario or distributed protocols. In particular, distributed cryptographic protocols aim to ensure data integrity and storage correctness across multiple servers. However, most of proposed solutions do not consider dynamic data operations which limits their usability in cloud. An efficient and practical distributed mechanism has to be modelled for cloud data storage in such a way that it ensures storage correctness, fast localization of data error (i.e., identification of misbehaving server), dynamic data support, and dependability. Note that these solutions assume that communication channels between every CSP and customer are already authenticated and reliable. Wang *et al.* [138] proposed a method to assure data storage security and supports dynamic operations on data blocks utilizing homomorphic token with distributed verification of erasure-coded data.

Proof-of-Storage schemes (PoS) are proposed to allow the clients to verify that their remote files are unchanged on the cloud side even though they do not possess any local copy of these files. In this context, there are mainly two approaches: Provable Data Possession (PDP)and Proof of Retrievability (PoR). PDP schemes basically verify that the cloud server receives a file correctly. More concretely, in order to generate some metadata to store it locally, the data owner first processes the data file and then it is sent to the server. The data owner later verifies the possession of the file by executing a challenge-response protocol. Finally, the data owner deletes the local copy of the file. POR schemes, on the other hand, ensure the data owner that the file is retrievable correctly without any loss or corruption. These schemes have been later extended in several ways in [123, 53, 24, 145, 148].

In a PDP scheme, the users sign the file blocks with authentication. While verifying, the user challenges the cloud by randomly choosing file blocks. In order to guarantee the proof possession, the cloud returns a short proof of possession. The main idea is that the complexity of the response from the cloud is constant, because of the homomorphic property of authentication tags reduces them into a short string. In this scenario, any malicious data modification or deletion will be detected with very high probability. Additionally in POR, the remote file blocks will contain the error correction codes. The CSP also provides a proof that the entire data or

file can be recovered in case of an error during the communication.

The proof-of-storage schemes (PoS) can catch a misbehaving CSP only after the targeted files which have already been changed. The main obstacle of proof-of-storage schemes is that an audit can verify the integrity of a file only at certain time. Therefore, the storage must be challenged regularly in order to ensure the integrity of the files that they still indeed exist and are not modified. Ateniese *et al.* [23] considered a different approach based on making changing or deleting files extremely troublesome for the CSP. By making the clients encode all their files into a single digital clew c, an entangled encoding, that can be used as a representation of all files and be stored on the cloud. The goal is to ensure that any small change to c will disrupt the content of all files. This approach of data entanglement was originally proposed by Aspnes *et al.* [22]. Unfortunately, in the original model of Aspnes *et al.* , a trusted authority is needed to create the entanglement. And the files can be retrieved only through the trusted authority which make the schemes within their framework are not suitable for cloud computing. Ateniese *et al.* in [23], the authors focused on storage schemes where the entanglement is collectively created by all clients and files can also be retrieved without interacting with any trusted entity. They referred to their framework as entangled storage.

(d) Partially and Fully Homomorphic Encryption Schemes

In most current existing systems data is transferred to the Cloud using standard encryption mechanisms (symmetric and asymmetric encryption) to ensure the security of the system and the secrecy of the storage. Standard encryption schemes allow to encrypt the data before it is sent to the CSP and it requires to decrypt the data at every operation. In order to do these operations, the user needs to provide the private key to the CSP for decryption which results in potential confidentiality and privacy leakage of the data stored in the Cloud. Namely, moving to the cloud gives over the control or possession of the user data and its computations. Because user looses the physical control over the computations in the cloud, the client faces new security concerns especially high-value assets such as cryptographic keys. After a decade of research, it is widely accepted that the most profound effect on cloud, as far as the solutions to various problems restraining its secure implementation is concerned, may come from the applications based on secure multi-party computation (SMPC) and from general fully homomorphic encryption (FHE) schemes. In the following discussion, we will look at schemes proposed by cryptographers with reference to cloud computing and try to analyze the propounded solutions.

Homomorphic encryption schemes can be used to perform operations on encrypted data without knowing the private key or plaintext data. Decrypting the result of any computation gives the same results as as the operation on the plain form of the data. An encryption scheme is homomorphic when having only $Enc(x)$ and $Enc(y)$, it is possible to compute $Enc(f,(x,y))$ without using the private key, where f is $+$, or $\oplus$ (e.g., Paillier Encryption is additive homomorphic, Goldwasser-Micali encryption is XOR-homomorphic, textbook RSA and ElGamal encryptions are multiplicative). FHE schemes provides the user with the ability to carry out arbitrary computation over encrypted data without being decryption, which prevents the CSP from learning anything about the user's data, i.e., it evaluates circuits over encrypted data without decryption (see Figure 5 for the FHE model in the cloud setting.). From a mathematical point of view, partial homomorphic (either additive or multiplicative) encryptions are actually group homomorphic and FHEs are actually ring homomorphic which satisfies both addition and multiplication. Because XOR (addition in modulo 2 as in $\mathcal{F}_2$) and AND (multiplication in modulo 2) are Turing complete (i.e., any function can be written as a combination of only XOR and AND gates), having an efficient FHE is the most ideal candidate for cloud security [71, 156, 153].

An FHE cryptosystem consists of four algorithms: KeyGen, Enc, Dec, and Eval algorithm. The first three algorithms are defined exactly the same as in any public-key cryptosystem. Given a secu-

rity parameter ℓ, the $\mathsf{KeyGen}(1^\ell)$ algorithm produces a key-pair (pk, sk). Moreover, Eval takes as input pk, a circuit C and a set of ciphertexts $\varphi = (\phi_1, \ldots, \phi_k)$ with ϕ_i=$\mathsf{Enc}(pk, m_i)$, $i = 1, \ldots, k$ and outputs a ciphertext ϕ. The scheme is homomorphic if for any key-pair (pk, sk) generated by $\mathsf{KeyGen}(1^\ell)$, any circuit C, any plaintexts $m_1, \ldots, m_k$, and any ciphertexts $\varphi = (\phi_1, \ldots, \phi_k)$ with $\phi_i = \mathsf{Enc}(pk, m_i)$, we have that

$\phi = \mathsf{Eval}(pk, C, \varphi)$ implies $C(m_1, \ldots, m_k) = \mathsf{Dec}(sk, \phi)$.

The first FHE scheme was introduced by Craig Gentry in 2009 [66]. He constructed a somewhat homomorphic encryption scheme (SWHE) using ideal lattices that is limited to evaluate polynomials of low-degree over encrypted data (the limitation of low-degree polynomials caused by the noise of ciphertexts, which grows slightly during homomorphic addition, and explosively during homomorphic multiplication). Then he modified the SWHE scheme to make it evaluate its own decryption circuit making it bootstrappable (bootstrapping runs the decryption function on the ciphertext homomorphically, using an encrypted secret key, which will reduce noise). Last but not least, he showed that FHE can be implemented through a recursive, self embedded, bootstrappable homomorphic encryption scheme (SWHE). Researchers have since suggested variants/improvements to Gentry's model [45, 127, 139]. Use of FHE schemes solves many of the cloud data security problems discussed in Section 3 since the client can send his request as encrypted, the CSP handles this request also under encryption and sends the encrypted results back to the client who can decrypt the result. The CSP does not see what has happened. Although there are significant improvements the existing implementations are unfortunately not yet to be practical in the cloud setting [119, 73].

Using Principal-Ideal Lattices of Prime Determinant. Some variants of Gentry's scheme yield a smaller key size, a simpler encryption/decryption algorithm and can be described without resorting to lattices [45, 127]. However, it was not a practical FHE scheme because they were not able to support large enough dimension to make Gentry's squashing technique work, so it was not easy to implement the bootstrapping functionality that is needed to get the complete scheme to work. Later Gentry and Halevi [67] presented a work in the same direction of Smart and Vercauteren's implementation [126]. They showed a number of optimizations that allow implementing all aspects of the scheme, including the bootstrapping functionality.

Brakerski *et al.* [37] presented a new way of constructing leveled FHE schemes which can evaluate arbitrary polynomial-size circuits without Gentry's bootstrapping procedure. The proposed scheme is based on the learning with error (LWE) or ring-LWE (RLWE) problems that have 2^λ security against known attacks ($\lambda \in \{0,1\}^*$ is a security parameter). For RLWE, the authors achieved that

- A leveled FHE scheme that does not use the bootstrapping procedure, can evaluate L-level arithmetic circuits with $\widetilde{O}(\lambda \cdot L^3)$ per-gate computation. The security of the underlying mechanism is based on RLWE for an approximation factor exponential in L.
- A leveled FHE scheme that uses the bootstrapping procedure, can evaluate the circuits with $\widetilde{O}(\lambda^2)$ the per-gate computation including the bootstrapping technique. Note that the complexity is independent of L and the security of the underlying mechanism is based on the hardness of RLWE for quasi-polynomial factors. Their scheme has improved the previous existing schemes which required sub-exponential factors.

The authors found similar results for LWE as well but they said that the performance were worse than using RLWE.

(e) Token-Based Cloud Computing

Traditional security measures include user authentication to enable data access. After user authentication transmission security needs to be established. The Secure Sockets Layer protocol handles authentication, key exchange and message confidentiality through symmetric and asymmetric cryptographic mechanisms. The underlying protocol is

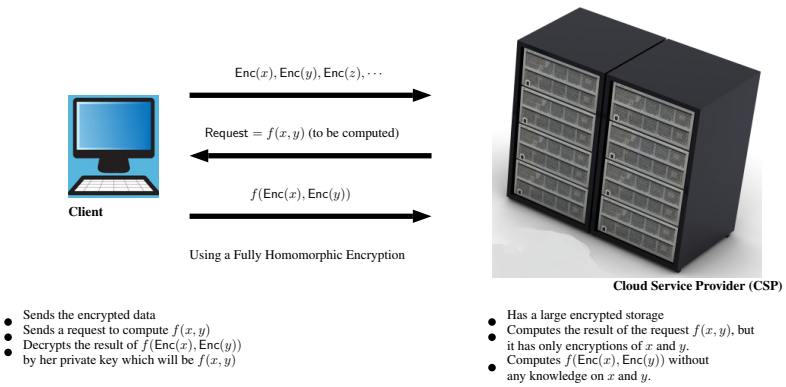

Fig. 5 Fully Homomorphic Encryption Applied to the Cloud Computing

mature and also used extremely widely in network applications. What is new and crucial about cloud computing is that sensitive user information leaves the secure premises of an organization and is saved in the CSP's data centers. When sensitive user data is saved in the cloud, the CSP has a degree of control over users' data. Unfortunately current cloud models are not transparent after this point. Data owners do not know if their data is abused or leaked. Current security approaches try to make the data secure in devices and in networks where it travels. This model is defined as device-centric or network-centric.

For cloud computing and other new information technologies information-centric protection of data was introduced by Chong *et al.* [42]. In this concept, data needs to be self-defending and self-describing. Namely, when the data is accessed, it should consult its describers and attempt to re-create a secure environment and reveal only if the environment is reliable through the use of Trusted Computing. Since information-centric security is not easily realizable yet, device-centric security solutions have been adopted for cloud computing. In some of these approaches data owners have a degree of control over cloud hardware. This control is made available through a trusted computing solution. An independent trusted monitor at the cloud server which audits all data access on the cloud server may assure data owner that certain access policies are not violated.

Trusted computing has another potential role in securing user data at cloud hardware which is described as token-based cloud computing [121]. A hardware token is a device which can perform cryptographic operations in a tamper-proof and leakage-free way (e.g., standard smart cards, USB devices). In cloud computing context a hardware token can reside either at user side or at the CSP. Depending on the protocol there can be only one token at one of the communicating sides or there can be two tokens at each of the sides. For example, Sadeghi *et al.* in [121] proposed to use a hardware token in order to outsource computation to an untrusted CSP. Roughly speaking, the token generates Yao's garbled circuits of a given functionality during the setup phase and then forwards them to the CSP to be evaluated without leaking information about the user's data. In the following, we will mention different aspects of token-based cloud implementations.

1. **Functionality.** Hardware security token can be used for various purposes. Strong two-factor authentication, oblivious database search (ODBS), verifiable encryption, and secure function evaluation (SFE) are some of the functionalities that hardware tokens are used for. It is clear that essentially anyone who obtains the correct credentials can access the sensitive data because sensitive data of a client is not stored on a local computer but in the cloud. Therefore, strong two-factor authentication is one of the most vital requirements for organizations. Hardware tokens provide the most mature solutions to replace the static passwords and mitigate the risks.

2. **Weight.** The underlying cryptographic algorithms in a token may be lightweight or expensive depending on the complexity of the cryptographic primitives. For example, public key cryptographic computations are relatively expensive whereas a lightweight token can only perform less complex operations such as executing symmetric algorithms. In general, tamper-proof tokens have physical protection measures to prevent key exposure, reverse engineering and other attacks. As an example of an expensive token we can consider WebAlps application of Jiang *et al.* [82]. They proposed to employ a secure coprocessor together with the existing cloud infrastructure. This coprocessor is chosen as IBM-4758 secure coprocessor platform. Sensitive applications are hosted at the trusted co-processor. When the client wants to access these secure applications, an authenticated SSL connection between the client and secure co-processor is established. In this example the client needs to trust in secure co-processor. Here the root of trust comes from a third party provider, namely IBM.
   TrustedPals is another secure multi-party computation solution which uses smart cards in each party [64]. They use Java Card Technology enabled smart cards as trusted tokens. The upper layer of the protocol is implemented as a standard Java application. Smart cards are used to encrypt and decrypt all protocol messages. The host application is used to handle the communication parts of the protocol. In this example two light tokens are used. Root of trust comes again from a third party smart card provider.
3. **Hardware-Based Trust.** Depending on particular security concerns, a protocol might require a hardware-based root of trust and therefore hardware tokens can be provided for both the client and the CSP. Note that secure hardware token processors can support expensive asymmetric cryptographic algorithms and clients can access to their data and run applications securely, and can even add an extra protection with a PIN.
   Cryptography-as-as-a-Service (CaaS) model performs cryptographic operations (like encryption/decryption) on behalf of a device via web services APIs. In order to send a message M from a client C_A to another client C_B securely, then C_A sends M to the CaaS provider, CaaS encrypts as Encrypt(M) and sends it back to C_A. Next, C_A forwards the encryption $\mathsf{Encrypt}(\mathsf{M})$ to C_B which also sends to CaaS for decryption. CaaS decrypts and sends the message M to C_B. In this model, the clients (i.e., devices) never learn the cryptographic keys because they are only stored in the CaaS provider.
4. **State Property.** Secure two-party communications need strong random generators such as tamper-proof hardware tokens for encryption of protocol messages. Furthermore, Goldreich and Ostrovsky shows that these tokens can also be used for software protection [68]. Running a secure multi-party computation protocol on a tamper-proof hardware is considered with either stateful or stateless tokens. If a cryptographic algorithm is used inside the token and the token needs to preserve the state of the algorithm, those tokens are called as "stateful tokens" [55]. Therefore, the memory of stateful tokens must be updatable which may lead to reset attacks. If the algorithm is used in a protocol but the protocol does not need to recall a previous state they are called as "stateless tokens". By definition stateful tokens are required to be tamper-proof, therefore stateless tokens are more being preferred.

In order to thwart the attacks against the insecure devices such as API, the security model of the CSP should be carefully analyzed and the implementation of strong authentication and access controls via secure transmission should be developed and verified in a correct manner.

YYY We illustrate the state-of-the-art technology readiness of modern cryptographic mechanisms in Table 4. In the following sections, we will focus on other environmental security, compliance and legal/contractual issues. YYY

	Ready for Deployment	Short-Term Research Needed	Longer-Term Research Needed
Secure Multi-Party Computation	✓	✓	✗
Searchable Encryption	✗	✓	✓
Message Locked Encryption	✓	✓	✗
Fully Homomorphic Encryption	✗	✗	✓
Order Preserving Encryption	✗	✗	✓
Attribute Based Encryption	✗	✓	✗
Delegated/Federated Authentication	✓	✓	✗
Delegated Computation	✗	✗	✓
Proof of Data Possession	✓	✓	✗
Cloud Key Management	✓	✓	✗
Data Integrity	✓	✓	✗
Access Policy Based Encryption	✓	✓	✗
Secure Data Dissemination	✓	✓	✗

Table 4 Technology Readiness of Modern Cryptographic Mechanisms (partly taken from [59] report)

5 Cloud Computing Security Framework: Other Security Issues and Existing Methodologies

Cloud computing security framework, aimed at ensuring customer information security, is comprised of two types of systems: Cloud computing server system and cloud computing standard system. Cloud computing server system comprises cloud security application services, cloud fundamental security services and credible cloud infrastructure services. This cloud security framework offers new ideas for solving the cloud security issues, which is an important practical methodology for guaranteeing the data security.

(a) Side-Channel Attacks

One of the most important emerging concerns are the side-channel attacks against virtualization platforms. Many of these attacks are using cache-timing attacks where the memory access pattern reveals information about confidential data. An attacker may put her virtual machine (VM) on the same physical machine as another targeted user's VM in order to gain confidential knowledge from targeted VM. In [118], Risten *et al.* show a case study for the Amazon EC2 service and call this type of attack a *cross-VM side-channel attack*. Crane *et al.* in [46] proposed a mechanism to thwart cache-based side-channel attacks without any hardware changes.

(b) Social Networking Attacks

Use of popular corporate and personal social networking sites increase the risk of advanced social engineering attack. Many employees of CSPs, suppliers, and vendors will be listed on social networking sites and are probably connected to each other. Because of these relationships, malicious adversaries can easily set up fake identities in order to gain honest parties' trust, and use their private information (e.g., personal interests, roles, knowledge) to prepare their attacks. In [136] Timm and Perez highlights the most dangerous attacks to social networks like Facebook, Twitter, and Myspace. They also propose possible solutions to mitigate and defend against such attacks in details.

(c) Bring Your Own Device (BYOD): Mobile Device Attacks

Cloud enterprises has a lot interest in the "Bring your own Device" (BYOD) trend. They are basically not providing any hardware/software to their customers, instead by adopting BYOD policies, each customer can choose their own computers or mobile devices (like smart phones). Note that in this

case the customers can work more efficiently and effectively because they are already familiar with these devices [99]. With the increasing use of smart phones cloud connectivity is now no longer limited to laptops or desktop PCs. Attacks on mobile devices are now emerging because similar vulnerabilities exist as traditionally associated with laptops and desktop PCs.

As mobile devices now have equivalent features as in traditional devices, Internet-based spyware, worms or even physical attacks are more likely to occur against mobile devices.

(d) Compliance

There are various regulations related to the storage and use of data, e.g., Payment Card Industry Data Security Standard (which regulates any organization that stores, transmits, or transacts credit card data), the Health Insurance Portability and Accountability Act (include requirements about security and privacy of health data), and the Sarbanes-Oxley Act (which aims to protect shareholders and the general public from accounting errors, fraudulent practices, and business disclosures). Standard reporting and audit trails have to be performed with such regulations and CSPs should be forced to allow their customers to comply appropriately with these regulations.

- **Business Continuity and Data Recovery.** Business continuity and data recovery plans have to be modelled properly and be in place by CSPs in order to ensure that cloud services can still be sustainable and data loss can be recovered in case of a disaster or an emergency. These plans should be shared with the users and reviewed by them as well [112].
- **Logs and Audit Trails.** Logs and audit trails should be kept continuously and CSPs should work with their customers to assure that these logs and audit trails are properly secured and maintained. Whenever a customer requests these logs and trails, it should be accessible for the purposes of inspection (e.g., e-Discovery) [112]. Performing internal and external audits regularly in order to monitor CSP's compliance to agreed terms, conditions, standards and regulations is considered good practice.
  An identity and access management mechanism should be able to track and satisfy all the basic audit requirements (e.g., track who has access to what information, check whether the access is required to perform the job, monitor usage logs and report them properly. Monitoring the use of cloud resources from user access point of view is critical, as it is needed to identify and prevent access violations. Furthermore, the system should be designed in such a way that risk exposure is quantifiable in order to reduce the residual risk. In order to manage the audit correctly, separation of duties and role-based access control mechanisms should also be developed.
- **Unique Compliance Requirements.** CSPs supply data center services which may also be subject to compliance requirements. Because customer data may not remain in the same data center or the same provider's cloud, using a CSP can lead to additional security and privacy concerns considering data jurisdiction. Note that data jurisdiction is a crucial concern for all private or public cloud providers.

(e) Legal and Contractual Issues

CSPs and their customers would have to negotiate terms considering liability for any risks they undertake, intellectual property, and end-of-service. Note that it is important to specify the requirements about how to resolve in case of data loss or compromise. Similarly, the end-of-service should cover when the CSP can send the customer's data and applications back to her [81, 151].

(f) Service Level Agreement

The Service Level Agreements (Cloud SLAs) define the legal relationship between a cloud service customer and a CSP of a cloud service. They are in fact crucial components which help the parties agree upon certain points and protect both sides.

There are typically five different cloud security levels: 1) Server access security, 2) Internet access security, 3) Database access security, 4) Data privacy, and 5) Program access security. Cloud SLA additionally defines the relationship between the CSP and its clouds, which includes definition of services, performance management, problem management, customer duties and responsibilities, warranties and remedies, disaster recovery, and business continuity. Cloud SLA also discusses about other issues like security policies, security methods and their implementations [86, 13, 8]. Note that Cloud SLAs will be naturally different because of distinct cloud services and deployment models, which makes SLAs more complicated. Therefore, Cloud SLAs will not often be the same for different CSPs and customers will not easily compare cloud services.

6 Conclusion

Because of flexibility and lower cost advantages cloud computing offers attractive alternatives to IT departments. Since the concept of cloud computing was proposed in its modern form around 2006, cloud security has become the most critical obstacle to have widespread usage. When considering solutions to the cloud computing problems, it is important to highlight that the main security issues for the cloud setting are essentially old problems. Main security problems in cloud computing can be classified into data security and privacy, data storage security and adversarial attacks. Many researchers have developed various models or schemes for solving such problems. Modern cryptographic mechanisms are expected to enhance data privacy and in general strengthen cloud computing security in the near future.

After a thorough discussion on various problems/solutions related to cloud computing, derived from classical as well as modern cryptography, respectively; we emphasize that it is very important to develop and transform these for the real setting we have today. As in all engineering problems, there are trade-offs to decide on cloud computing. While new innovations emerge, they introduce new problems and new innovations are found to solve these new problems. Innovations of cars caused traffic accidents, but later traffic lights were introduced to solve that problem. Nowadays electric cars are being introduced as a solution to environmental problems. Analogously cloud computing is creating new opportunities but causing new problems, some of which are serious security concerns. Modern cryptographic approaches are trying to overcome those deficiencies.

Markets keep developing new software applications, platforms, and infrastructure as a service over the “Cloud”. These services are already accessible on a pay-per-use basis and provide great flexibility and alternatives to reduce capital costs. Open source clouds such as the Ubuntu cloud, OpenStack, Eucalyptus, Open-Nebula, and CloudStack offer various services can give the chance to try out the benefits of cloud computing. A real opportunity will exist for the future once the world achieves a mature cloud computing technology. People can all benefit from countless advantages technology offers. The cloud technology is already widely in use today and with the advent of impressive results this will certainly continue to grow.

Acknowledgements This research is supported by a grant from Ministry of Development of Turkey provided to the Cloud Computing and Big Data Research Lab Project. The author would like to thank Markku-Juhani Saarinen, Devrim Ünal, Thomas-Brochmann Pedersen, İsa Sertkaya and Osmanbey Uzunkol for their time, insightful comments, and support.

References

1. Trusted Computing Group-Cloud Computing and Security A Natural Match (2010). Available at `http://www.trustedcomputinggroup.org/` (Accessed on; April 2015)
2. Cloud Security Alliance-Security Guidance for Critical Areas of Focus in Cloud Computing (v3.0) (2011). Available at `https://cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf`
3. Cloud Standards Customer Council-Security for Cloud Computing 10 Steps to Ensure Success (2012). Available at

http://www.cloud-council.org/Security_for_Cloud_Computing-Final_080912.pdf (Accessed on; April 2015)
4. Cloud Security Alliance-Cloud Vulnerabilities Working Group-Cloud Computing Vulnerability Incidents: A Statistical Overview (2013). Available at https://cloudsecurityalliance.org/download/cloud-computing-vulnerability-incidents-a-statistical-overview/
5. Digital Agenda For Europe-Cloud Service Level Agreement Standardisation Guidelines (2014). Available at https://ec.europa.eu/digital-agenda/en/news/cloud-service-level-agreement-standardisation-guidelines
6. DRAFT NIST Special Publication 800-125-A-Security Recommendations for Hypervisor Deployment (2014). Available at http://csrc.nist.gov/publications/drafts/800-125a/sp800-125a_draft.pdf (Accessed on; May 2016)
7. European Cloud Partnership-Establishing a Trusted Cloud Europe: A policy vision document by the Steering Board of the European Cloud Partnership (2014). Available at http://ec.europa.eu/digital-agenda/en/news/trusted-cloud-europe
8. European Strategy- A Europe 2020 Initiative-Cloud Service Level Agreement Standardisation Guidelines (2014). URL https://ec.europa.eu/digital-agenda/en/news/cloud-service-level-agreement-standardisation-guidelines
9. NIST Big Data Public Working Group-Security and Privacy Requirements (2014). Available at http://jtc1bigdatasg.nist.gov/_workshop2/10_NBD_SnP.pdf
10. Cloud Standards Customer Council-Security for Cloud Computing Ten Steps to Ensure Success (2015). Available at http://www.cloud-council.org/deliverables/CSCC-Security-for-Cloud-Computing-10-Steps-to-Ensure-Success.pdf
11. Abadi, M., Boneh, D., Mironov, I., Raghunathan, A., Segev, G.: Message-locked encryption for lock-dependent messages. In: Advances in Cryptology CRYPTO 2013, *Lecture Notes in Computer Science*, vol. 8042, pp. 374–391. Springer Berlin Heidelberg (2013)
12. Abo-alian, A., Badr, N., Tolba, M.: Auditing-as-a-Service for Cloud Storage. In: Intelligent Systems'2014, *Advances in Intelligent Systems and Computing*, vol. 322, pp. 559–568. Springer International Publishing (2015)
13. Aceto, G., Botta, A., De Donato, W., Pescapè, A.: Survey Cloud Monitoring: A Survey. Computer Networks **57**(9), 2093–2115 (2013)
14. Adamov, A., Erguvan, M.: The truth about cloud computing as new paradigm in it. In: Application of Information and Communication Technologies, 2009. AICT 2009. International Conference on, pp. 1–3. IEEE (2009)
15. Adrian, D., Bhargavan, K., Durumeric, Z., Gaudry, P., Green, M., Halderman, J.A., Heninger, N., Springall, D., Thomé, E., Valenta, L., VanderSloot, B., Wustrow, E., Zanella-Béguelin, S., Zimmermann, P.: Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice. In: Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, pp. 5–17. ACM (2015)
16. AlBelooshi, B., Salah, K., Martin, T., Damiani, E.: Securing Cryptographic Keys in the IaaS Cloud Model. In: 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC), pp. 397–401 (2015)
17. AlFardan, N.J., Bernstein, D.J., Paterson, K.G., Poettering, B., Schuldt, J.C.N.: On the Security of RC4 in TLS. In: Proceedings of

the 22Nd USENIX Conference on Security, SEC'13, pp. 305–320. USENIX Association (2013)
18. AlZain, M., Soh, B., Pardede, E.: MCDB: Using Multi-clouds to Ensure Security in Cloud Computing. In: Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on, pp. 784–791 (2011)
19. Amazon: Amazon Web Services: Risk and Compliance (2014). URL http://media.amazonwebservices.com/AWS_Risk_and_Compliance_Whitepaper.pdf
20. Ardagna, C.A., Asal, R., Damiani, E., Vu, Q.H.: From Security to Assurance in the Cloud: A Survey. ACM Computing Surveys **48**(1), 2:1–2:50 (2015)
21. Asghar, M.R., Russello, G., Crispo, B., Ion, M.: Supporting Complex Queries and Access Policies for Multi-user Encrypted Databases. In: Proceedings of the 2013 ACM Workshop on Cloud Computing Security Workshop, CCSW '13, pp. 77–88. ACM (2013)
22. Aspnes, J., Feigenbaum, J., Yampolskiy, A., Zhong, S.: Towards a Theory of Data Entanglement. Theoretical Computer Science **389**(1), 26–43 (2007)
23. Ateniese, G., Dagdelen, z., Damgard, I., Venturi, D.: Entangled Cloud Storage. IACR Cryptology ePrint Archive (2012). URL http://eprint.iacr.org/2012/511
24. Ateniese, G., Kamara, S., Katz, J.: Proofs of Storage from Homomorphic Identification Protocols. In: Advances in Cryptology ASIACRYPT 2009, *Lecture Notes in Computer Science*, vol. 5912, pp. 319–333. Springer Berlin Heidelberg (2009)
25. Aviram, N., Schinzel, S., Somorovsky, J., Heninger, N., Dankel, M., Steube, J., Valenta, L., Adrian, D., Halderman, J.A., Dukhovni, V., Käsper, E., Cohney, S., Engels, S., Paar, C., Shavitt, Y.: DROWN: Breaking TLS with SSLv2. In: Proc. 25th USENIX Security Symposium (2016)
26. Bauer, E., Adams, R.: Reliability and Availability of Cloud Computing, 1st edn. Wiley-IEEE Press (2012)
27. Belk, M., Coles, M., Goldschmidt, C., Howard, M., Randolph, K., Saario, M., Sondhi, R., Tarandach, I., Vaha-Sipila, A., Yonchev, Y.: SAFECode Whitepaper: Fundamental Practices for Secure Software Development 2nd Edition. In: ISSE 2014 Securing Electronic Business Processes: Highlights of the Information Security Solutions Europe 2014 Conference, pp. 1–32. Springer Fachmedien Wiesbaden (2014)
28. Bellare, M., Keelveedhi, S.: Interactive Message-Locked Encryption and Secure Deduplication, chap. Public-Key Cryptography – PKC 2015: 18th IACR International Conference on Practice and Theory in Public-Key Cryptography, Gaithersburg, MD, USA, March 30 – April 1, 2015, Proceedings, pp. 516–538. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)
29. Bellare, M., Keelveedhi, S., Ristenpart, T.: DupLESS: Server-aided Encryption for Deduplicated Storage. In: Proceedings of the 22Nd USENIX Conference on Security, SEC'13, pp. 179–194. USENIX Association (2013)
30. Bellare, M., Keelveedhi, S., Ristenpart, T.: Message-Locked Encryption and Secure Deduplication. In: Advances in Cryptology EUROCRYPT 2013, *Lecture Notes in Computer Science*, vol. 7881, pp. 296–312. Springer Berlin Heidelberg (2013)
31. Bessani, A., Correia, M., Quaresma, B., André, F., Sousa, P.: DepSky: Dependable and Secure Storage in a Cloud-of-Clouds. In: Proceedings of the Sixth Conference on Computer Systems, EuroSys '11, pp. 31–46. ACM, New York, NY, USA (2011)
32. Bethencourt, J., Song, D., Waters, B.: New techniques for private stream searching. ACM Transactions on Information and System Security (TISSEC) **12**(3), 16 (2009)
33. Beurdouche, B., Bhargavan, K., Delignat-Lavaud, A., Fournet, C., Kohlweiss, M., Pironti, A., Strub, P.Y., Zinzindohoue, J.K.:

A Messy State of the Union: Taming the Composite State Machines of TLS. In: 2015 IEEE Symposium on Security and Privacy, pp. 535–552 (2015)
34. Beurdouche, B., Bhargavan, K., Delignat-Lavaud, A., Fournet, C., Kohlweiss, M., Pironti, A., Strub, P.Y., Zinzindohoue, J.K.: A Messy State of the Union: Taming the Composite State Machines of TLS. In: 2015 IEEE Symposium on Security and Privacy, pp. 535–552 (2015)
35. Bleikertz, S., Bugiel, S., Ideler, H., Nürnberger, S., Sadeghi, A.R.: Client-Controlled Cryptography-as-a-Service in the Cloud, chap. Applied Cryptography and Network Security: 11th International Conference, ACNS 2013, Banff, AB, Canada, June 25-28, 2013. Proceedings, pp. 19–36. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
36. Blome, A., Ochoa, M., Li, K., Peroli, M., Dashti, M.T.: VERA: A Flexible Model-Based Vulnerability Testing Tool. In: 2013 IEEE Sixth International Conference on Software Testing, Verification and Validation, pp. 471–478 (2013)
37. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) Fully Homomorphic Encryption Without Bootstrapping. ACM Trans. Comput. Theory **6**(3), 13:1–13:36 (2014)
38. j. Brodkin: Seven Cloud-Computing Security Risks. www.infoworld.com/d/security-central/gartner-seven-cloud-computing-security-risks-853. (2008)
39. Canard, S., Pointcheval, D., Sanders, O.: Efficient Delegation of Zero-Knowledge Proofs of Knowledge in a Pairing-Friendly Setting. In: Public-Key Cryptography PKC 2014, *Lecture Notes in Computer Science*, vol. 8383, pp. 167–184. Springer Berlin Heidelberg (2014)
40. Chadwick, D., Siu, K., Lee, C., Fouillat, Y., Germonville, D.: Adding Federated Identity Management to OpenStack. Journal of Grid Computing **12**(1), 3–27 (2014)
41. Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., Molina, J.: Controlling Data in the Cloud: Outsourcing Computation Without Outsourcing Control. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security, CCSW '09, pp. 85–90. ACM (2009)
42. Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., Molina, J.: Controlling Data in the Cloud: Outsourcing Computation Without Outsourcing Control. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security, CCSW '09, pp. 85–90. ACM (2009)
43. Chow, S.S.M., He, Y.J., Hui, L.C.K., Yiu, S.M.: SPICE: Simple Privacy-Preserving Identity-Management for Cloud Environment. In: Proceedings of the 10th International Conference on Applied Cryptography and Network Security, ACNS'12, pp. 526–543. Springer-Verlag (2012)
44. ComputerWeekly.com: Microsoft Azure had more downtime in 2014 than main cloud rivals (2015). Available at `http://www.computerweekly.com/news/2240238379/Microsoft-Azure-had-more-downtime-than-main-cloud-rivals`
45. Coron, J.S., Lepoint, T., Tibouchi, M.: Scale-Invariant Fully Homomorphic Encryption over the Integers. In: Public-Key Cryptography PKC 2014, *Lecture Notes in Computer Science*, vol. 8383, pp. 311–328. Springer Berlin Heidelberg (2014)
46. Crane, S., Homescu, A., Brunthaler, S., Larsen, P., Franz, M.: Thwarting Cache Side-Channel Attacks Through Dynamic Software Diversity. In: 22nd Annual Network and Distributed System Security Symposium, NDSS 2015, California, USA. The Internet Society (2015). URL `http://www.internetsociety.org/doc/thwarting-cache-side-channel-attacks-through-dynamic-software-diversity`
47. CSA: Top Ten Big Data Security and Privacy Challenges (2012). Available at `https://downloads.cloudsecurityalliance.org/initiatives/bdwg/`

Big_Data_Top_Ten_v1.pdf
48. CSA: Cloud Security Alliance. (2013). The notorious nine: Cloud computing top threats in 2013. http://www.cloudsecurityalliance.org/topthreats (2013)
49. Curtmola, R., Garay, J., Kamara, S., Ostrovsky, R.: Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS '06, pp. 79–88. ACM (2006)
50. De Ruiter, J., Poll, E.: Protocol state fuzzing of tls implementations. In: Proceedings of the 24th USENIX Conference on Security Symposium, SEC'15, pp. 193–206. USENIX Association, Berkeley, CA, USA (2015). URL http://dl.acm.org/citation.cfm?id=2831143.2831156
51. Demmler, D., Schneider, T., Zohner, M.: ABY - A Framework for Efficient Mixed-Protocol Secure Two-Party Computation. In: Network and Distributed System Security Symposium (NDSS'15) (2015)
52. Dikaiakos, M., Katsaros, D., Mehra, P., Pallis, G., Vakali, A.: Cloud computing: Distributed internet computing for it and scientific research. Internet Computing, IEEE **13**(5), 10–13 (2009)
53. Dodis, Y., Vadhan, S., Wichs, D.: Proofs of retrievability via hardness amplification. In: Proceedings of the 6th Theory of Cryptography Conference on Theory of Cryptography, TCC '09, pp. 109–127. Springer-Verlag, Berlin, Heidelberg (2009)
54. Doroodchi, M., Iranmehr, A., Pouriyeh, S.A.: An investigation on integrating XML-based security into Web services. In: GCC Conference Exhibition, 2009 5th IEEE, pp. 1–5 (2009)
55. Dttling, N., Kraschewski, D., Mller-Quade, J.: Unconditional and Composable Security Using a Single Stateful Tamper-Proof Hardware Token. In: Theory of Cryptography, *Lecture Notes in Computer Science*, vol. 6597, pp. 164–181. Springer Berlin Heidelberg (2011)
56. Dwork, C.: Differential privacy: A survey of results. In: Theory and Applications of Models of Computation, *Lecture Notes in Computer Science*, vol. 4978, pp. 1–19. Springer Berlin Heidelberg (2008)
57. Ellison, L.: What the Hell is Cloud Computing (2012). Available at http://www.youtube.com/watch?v=0FacYAI6DY0
58. ENISA: Algorithms, key size and parameters report (2014). Available at https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-size-and-parameters-report-2014
59. ENISA: Study on Cryptographic Protocols (2014). Available at https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/study-on-cryptographic-protocols
60. Falkenberg, A., Mainka, C., Somorovsky, J., Schwenk, J.: A New Approach towards DoS Penetration Testing on Web Services. In: IEEE 20th International Conference on Web Services (ICWS), pp. 491–498 (2013)
61. Fardan, N.J.A., Paterson, K.G.: Lucky Thirteen: Breaking the TLS and DTLS Record Protocols. In: Security and Privacy (SP), 2013 IEEE Symposium on, pp. 526–540 (2013)
62. Fernandes, D.A., Soares, L.F., Gomes, J.a.V., Freire, M.M., Inácio, P.R.: Security Issues in Cloud Environments: A Survey. Int. J. Inf. Secur. **13**(2), 113–170 (2014)
63. Fett, D., Küsters, R., Schmitz, G.: SPRESSO: A Secure, Privacy-Respecting Single Sign-On System for the Web. In: Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, pp. 1358–1369. ACM (2015)
64. Fort, M., Freiling, F., Penso, L., Benenson, Z., Kesdogan, D.: TrustedPals: Secure Multi-

Party Computation Implemented with Smart Cards. Computer Security–ESORICS 2006 pp. 34–48 (2006)
65. Fujinoki, H.: Designs, analyses, and optimizations for attribute-shuffling obfuscation to protect information from malicious cloud administrators. Security and Communication Networks (2015)
66. Gentry, C.: A Fully Homomorphic Encryption Scheme. Ph.D. thesis, Stanford University, Stanford, CA, USA (2009)
67. Gentry, C., Halevi, S.: Implementing Gentrys Fully-Homomorphic Encryption Scheme. In: Advances in Cryptology EUROCRYPT 2011, *Lecture Notes in Computer Science*, vol. 6632, pp. 129–148. Springer Berlin Heidelberg (2011)
68. Goldreich, O., Ostrovsky, R.: Software Protection and Simulation on Oblivious RAMs. Journal of ACM **43**(3), 431–473 (1996)
69. Grobauer, B., Walloschek, T., Stocker, E.: Understanding cloud computing vulnerabilities. Security and Privacy, IEEE **9**(2), 50–57 (2011)
70. Gruschka, N., Iacono, L.L.: Vulnerable Cloud: SOAP Message Security Validation Revisited. In: IEEE International Conference on Web Services, ICWS '09, pp. 625–631 (2009)
71. Guellier, A.: Can Homomorphic Cryptography ensure Privacy? (2014). Available at https://hal.inria.fr/hal-01052509v1/document
72. Hadavi, M., Jalili, R., Damiani, E., Cimato, S.: Security and searchability in secret sharing-based data outsourcing. International Journal of Information Security pp. 1–17 (2015)
73. Halevi, S., Shoup, V.: Bootstrapping for HElib. IACR Cryptology ePrint Archive (2015). URL https://eprint.iacr.org/2014/873
74. Han, F., Qin, J., Hu, J.: Secure searches in the cloud: A survey. ElSevier-Future Generation Computer Systems (2016). URL http://www.sciencedirect.com/science/article/pii/S0167739X16000091
75. Harsh, P., Dudouet, F., Cascella, R.G., Jgou, Y., Morin, C.: Using open standards for interoperability - issues, solutions, and challenges facing cloud computing. CoRR **abs/1207.5949** (2012). URL https://hal.inria.fr/hal-00720636/file/Contrail-VEP-Interoperability.pdf
76. Hashizume, K., Rosado, D., Fernndez-Medina, E., Fernandez, E.: An Analysis of Security Issues for Cloud Computing. Journal of Internet Services and Applications **4**(1), 5 (2013)
77. Hendre, A., Joshi, K.P.: A semantic approach to cloud security and compliance. In: 2015 IEEE 8th International Conference on Cloud Computing, pp. 1081–1084 (2015)
78. Herranz, J., Ruiz, A., Sáez, G.: New Results and Applications for Multi-secret Sharing Schemes. Design Codes and Cryptography **73**(3), 841–864 (2014)
79. Huang, Y., Evans, D., Katz, J., Malka, L.: Faster secure two-party computation using garbled circuits. In: USENIX Security Symposium (2011)
80. a.w. Ideler, H.: Cryptography as a service in a cloud computing environment. Master's thesis, Endhoven University of Technology, the Netherlands (2012)
81. Jensen, M., Schwenk, J., Gruschka, N., Iacono, L.: On technical security issues in cloud computing. In: Cloud Computing, 2009. CLOUD'09. IEEE International Conference on, pp. 109–116. IEEE (2009)
82. Jiang, S., Smith, S., Minami, K.: Securing Web Servers Against Attack. In: Computer Security Applications Conference, 2001. ACSAC 2001. Proceedings 17th Annual, pp. 265–276. IEEE (2001)
83. Juels, A., Kaliski Jr, B.: PORs: Proofs of Retrievability for Large Files. In: Proceedings of the 14th ACM conference on Computer and communications security, pp. 584–597. ACM (2007)
84. Kaaniche, N.: Cloud data storage security based on cryptographic mechanisms. Theses, Institut National

des Télécommunications (2014). URL https://tel.archives-ouvertes.fr/tel-01146029

85. Kamara, S., Lauter, K.: Cryptographic Cloud Storage, chap. Financial Cryptography and Data Security: FC 2010 Workshops, RLCPS, WECSR, and WLC 2010, Tenerife, Canary Islands, Spain, January 25-28, 2010, Revised Selected Papers, pp. 136–149. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
86. Kandukuri, B., Paturi, V., Rakshit, A.: Cloud Security Issues. In: IEEE International Conference on Services Computing, SCC'09, pp. 517–520. IEEE (2009)
87. Kiraz, M.S.: Secure and Fair Two-Party Computation. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, the Netherlands (2008)
88. Kiraz, M.S., İsa Sertkaya, Uzunkol, O.: An Efficient ID-Based Message Recoverable Privacy-Preserving Auditing Scheme. In: Privacy, Security and Trust (PST), 2015 13th Annual Conference on, pp. 117–124. IEEE (2015)
89. Klemperer, P.F.: Efficient Hypervisor Based Malware Detection. Theses, Carnegie Mellon University (2015). URL http://repository.cmu.edu/dissertations/466/
90. Kreuter, B., Shelat, A., Shen, C.H.: Billion-gate secure computation with malicious adversaries. In: Proceedings of the 21st USENIX Conference on Security Symposium, Security'12, pp. 14–14. USENIX Association, Berkeley, CA, USA (2012)
91. Kupser, D., Mainka, C., Schwenk, J., Somorovsky, J.: How to Break XML Encryption – Automatically. In: 9th USENIX Workshop on Offensive Technologies (WOOT 15). USENIX Association, Washington, D.C. (2015). URL https://www.usenix.org/conference/woot15/workshop-program/presentation/kupser
92. Li, J., Chen, X., Li, M., Li, J., Lee, P., Lou, W.: Secure deduplication with efficient and reliable convergent key management. Parallel and Distributed Systems, IEEE Transactions on **25**(6), 1615–1625 (2014)
93. Li, J., Squicciarini, A., Lin, D., Liang, S., Jia, C.: SecLoc: Securing Location-Sensitive Storage in the Cloud. In: Proceedings of the 20th ACM Symposium on Access Control Models and Technologies, SACMAT '15, pp. 51–61. ACM (2015)
94. Liu, J., Asokan, N., Pinkas, B.: Secure Deduplication of Encrypted Data Without Additional Independent Servers. In: Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, pp. 874–885. ACM (2015)
95. Liu, Q., Tan, C., Wu, J., Wang, G.: Cooperative Private Searching in Clouds. Journal of Parallel and Distributed Computing (2012)
96. Liu, Q., Wang, G., Wu, J.: Time-based Proxy Re-encryption Scheme for Secure Data Sharing in a Cloud Environment. Information Sciences **258**, 355–370 (2014)
97. Mainka, C., Somorovsky, J., Schwenk, J.: Penetration Testing Tool for Web Services Security. In: 2012 IEEE Eighth World Congress on Services, pp. 163–170 (2012)
98. Malkhi, D., Nisan, N., Pinkas, B., Sella, Y.: Fairplay-A Secure Two-Party Computation System. In: USENIX Security Symposium, pp. 287–302 (2004)
99. Mansfield-Devine, S.: Interview: Byod and the enterprise network. Computer Fraud & Security **2012**(4), 14 – 17 (2012)
100. Masood, A., Java, J.: Static Analysis for Web Service Security - Tools Amp; Techniques for a Secure Development Life Cycle. In: Technologies for Homeland Security (HST), 2015 IEEE International Symposium on, pp. 1–6 (2015)
101. Masood, R., Shibli, M.A., Ghazi, Y., Kanwal, A., Ali, A.: Cloud authorization: exploring techniques and approach towards effective access control framework. Frontiers of Computer Science **9**(2), 297–321 (2015)
102. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. NIST Special Publication 800-145 (2011)

103. Nanavati, M., Colp, P., Aiello, B., Warfield, A.: Cloud Security: A Gathering Storm. Commun. ACM **57**(5), 70–79 (2014)
104. Noman, A., Adams, C.: Providing a Data Location Assurance Service for Cloud Storage Environments. Journal of Mobile Multimedia **8**(4), 265–286 (2012)
105. OASIS: Service Provisioning Markup Language (SPML) Version 1.0, OASIS (2003). Available at https://www.oasis-open.org/committees/download.php/4137/os-pstc-spml-core-1.0.pdf
106. OASIS: eXtensible Access Control Markup Language (XACML), OASIS (2005). Available at http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf
107. OASIS: WS-Security Profile of the OASIS Security Assertion Markup Language (SAML), OASIS (2012). URL https://www.oasis-open.org/standards
108. Opara-Martins, J., Sahandi, R., Tian, F.: Implications of Integration and Interoperability for Enterprise Cloud-based Applications. In: 6th International Conference on Cloud Computing. Springer-Verlag in the Lecture Notes of ICST (LNICST) (2015). URL http://eprints.bournemouth.ac.uk/22895/
109. Oriyano, S.P.: CEH v9: Certified Ethical Hacker Version 9 Study Guide-3rd Edition (2016). URL http://www.amazon.com/CEH-v9-Certified-Ethical-Version/dp/1119252245
110. Parno, B., Raykova, M., Vaikuntanathan, V.: How to Delegate and Verify in Public: Verifiable Computation from Attribute-Based Encryption. In: Theory of Cryptography, *Lecture Notes in Computer Science*, vol. 7194, pp. 422–439. Springer Berlin Heidelberg (2012)
111. Popa, R.A., Redfield, C.M.S., Zeldovich, N., Balakrishnan, H.: CryptDB: Processing Queries on an Encrypted Database. Commun. ACM **55**(9), 103–111 (2012)
112. Popovic, K., Hocenski, Z.: Cloud Computing Security Issues and Challenges. In: MIPRO, 2010 Proceedings of the 33rd International Convention, pp. 344–349. IEEE (2010)
113. Prabadevi, B., Jeyanthi, N.: Distributed Denial of Service Attacks and its Effects on Cloud Environment- A Survey. In: Networks, Computers and Communications, The 2014 International Symposium on, pp. 1–5 (2014)
114. Proudler, G., Chen, L., Dalton, C.: Direct Anonymous Attestation (DAA) in More Depth. In: Trusted Computing Platforms, pp. 339–352. Springer International Publishing (2014)
115. Rabotka, V., Mannan, M.: An Evaluation of Recent Secure Deduplication Proposals. Journal of Information Security and Applications **2728**, 3 – 18 (2016). Special Issues on Security and Privacy in Cloud Computing
116. Rastogi, A., Hammer, M.A., Hicks, M.: Wysteria: A Programming Language for Generic, Mixed-Mode Multi-Party Computations. In: Proceedings of the 2014 IEEE Symposium on Security and Privacy, SP '14, pp. 655–670. IEEE Computer Society (2014)
117. Ren, K., Wang, C., Wang, Q.: Security Challenges for the Public Cloud. Internet Computing, IEEE **16**(1), 69–73 (2012)
118. Ristenpart, T., Tromer, E., Shacham, H., Savage, S.: Hey, You, Get off of My Cloud: Exploring Information Leakage in Third-party Compute Clouds. In: Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS '09, pp. 199–212. ACM, New York, NY, USA (2009)
119. Rohloff, K., Cousins, D.B.: A Scalable Implementation of Somewhat Homomorphic Encryption Built on NTRU. In: WAHC'14 - 2nd Workshop on Applied Homomorphic Cryptography and Encrypted Computing (2014)
120. Roy, A., Sarkar, S., Ganesan, R., Goel, G.: Secure the Cloud: From the Perspective of a Service-Oriented Organization. ACM Comput. Surv. **47**(3), 41:1–41:30 (2015)

121. Sadeghi, A.R., Schneider, T., Winandy, M.: Token-based cloud computing **6101**, 417–429 (2010)
122. Samarati, P., di Vimercati, S.D.C.: Data Protection in Outsourcing Scenarios: Issues and Directions. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ASIACCS '10, pp. 1–14. ACM, New York, NY, USA (2010)
123. Shacham, H., Waters, B.: Compact Proofs of Retrievability. Journal of Cryptology **26**(3), 442–483 (2013)
124. Shankarwar, M., Pawar, A.: Security and Privacy in Cloud Computing: A Survey. In: Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, *Advances in Intelligent Systems and Computing*, vol. 328, pp. 1–11. Springer International Publishing (2015)
125. Shibboleth: The Shibboleth Consortium (2003). Available at `http://shibboleth.net`
126. Smart, N., Vercauteren, F.: Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In: Public Key Cryptography PKC 2010, *Lecture Notes in Computer Science*, vol. 6056, pp. 420–443. Springer Berlin Heidelberg (2010)
127. Smart, N., Vercauteren, F.: Fully homomorphic SIMD operations. Designs, Codes and Cryptography **71**(1), 57–81 (2014)
128. Somorovsky, J., Mayer, A., Schwenk, J., Kampmann, M., Jensen, M.: On Breaking SAML: Be Whoever You Want to Be. In: Presented as part of the 21st USENIX Security Symposium (USENIX Security 12), pp. 397–412. USENIX (2012). URL `https://www.usenix.org/conference/usenixsecurity12/technical-sessions/presentation/somorovsky`
129. Somorovsky, J., Schwenk, J.: Technical Analysis of Countermeasures against Attack on XML Encryption – or – Just Another Motivation for Authenticated Encryption. In: 2012 IEEE Eighth World Congress on Services, pp. 171–178 (2012)
130. Sood, S.K.: A Combined Approach to Ensure Data Security in Cloud Computing. Journal of Network and Computer Applications **35**(6), 1831–1838 (2012)
131. Sookhak, M., Talebian, H., Ahmed, E., Gani, A., Khan, M.K.: A review on remote data auditing in single cloud server: Taxonomy and open issues. Journal of Network and Computer Applications **43**(0), 121 – 141 (2014)
132. Stanek, J., Sorniotti, A., Androulaki, E., Kencl, L.: A Secure Data Deduplication Scheme for Cloud Storage. In: Financial Cryptography and Data Security, *Lecture Notes in Computer Science*, vol. 8437, pp. 99–118. Springer Berlin Heidelberg (2014)
133. Subashini, S., Kavitha, V.: Review: A Survey on Security Issues in Service Delivery Models of Cloud Computing. Journal of Network and Computer Applications **34**(1), 1–11 (2011)
134. Takabi, H., Joshi, J., Ahn, G.J.: Security and Privacy Challenges in Cloud Computing Environments. Security Privacy, IEEE **8**(6), 24–31 (2010)
135. Tep, K.S., Martini, B., Hunt, R., Choo, K.K.R.: A Taxonomy of Cloud Attack Consequences and Mitigation Strategies: The Role of Access Control and Privileged Access Management. In: Trustcom/BigDataSE/ISPA, 2015 IEEE, vol. 1, pp. 1073–1080 (2015)
136. Timm, C., Perez, R.: Seven Deadliest Social Network Attacks. Syngressg, Boston, USA (2010)
137. Vossaert, J., Lapon, J., De Decker, B., Naessens, V.: Trusted Computing to Increase Security and Privacy in eID Authentication. In: ICT Systems Security and Privacy Protection, *IFIP Advances in Information and Communication Technology*, vol. 428, pp. 485–492. Springer Berlin Heidelberg (2014)
138. Wang, C., Wang, Q., Ren, K., Lou, W.: Ensuring data storage security in Cloud Computing. In: Quality of Service, 2009. IWQoS. 17th International Workshop on, pp. 1–9 (2009)

139. Wang, F., Wang, K., Li, B.: LWE-Based FHE with Better Parameters. In: Advances in Information and Computer Security, *Lecture Notes in Computer Science*, vol. 9241, pp. 175–192. Springer International Publishing (2015)
140. Wang, J., Kissel, Z.A.: Introduction to Network Security: Theory and Practice, 2nd Edition. John Wiley & Sons, Boston, USA (2015)
141. Wang, L.l., Chen, K.f., Mao, X.p., Wang, Y.t.: Efficient and Provably-Secure Certificateless Proxy Re-Encryption Scheme for Secure Cloud Data Sharing. Journal of Shanghai Jiaotong University (Science) **19**(4), 398–405 (2014)
142. Wang, Y., Wong, D., Wu, Q., Chow, S., Qin, B., Liu, J.: Practical Distributed Signatures in the Standard Model. In: Topics in Cryptology CT-RSA 2014, *Lecture Notes in Computer Science*, vol. 8366, pp. 307–326. Springer International Publishing (2014)
143. Watson, G.J., Safavi-Naini, R., Alimomeni, M., Locasto, M.E., Narayan, S.: LoSt: Location Based Storage. In: Proceedings of the 2012 ACM Workshop on Cloud Computing Security Workshop, CCSW '12, pp. 59–70. ACM, New York, NY, USA (2012)
144. Wei, L., Zhu, H., Cao, Z., Dong, X., Jia, W., Chen, Y., Vasilakos, A.V.: Security and Privacy for Storage and Computation in Cloud Computing. Information Sciences **258**, 371–386 (2014)
145. Worku, S.G., Xu, C., Zhao, J., He, X.: Secure and Efficient Privacy-Preserving Public Auditing Scheme for Cloud Storage. Computers and Electrical Engineering **40**(5), 1703 – 1713 (2014)
146. Xiang, T., Li, X., Chen, F., Guo, S., Yang, Y.: Processing Secure, Verifiable and Efficient SQL Over Outsourced Database. Information Sciences **348**, 163 – 178 (2016). URL `http://www.sciencedirect.com/science/article/pii/S002002551630072X`
147. Xiong, J., Liu, X., Yao, Z., Ma, J., Li, Q., Geng, K., Chen, P.: A Secure Data Self-Destructing Scheme in Cloud Computing. Cloud Computing, IEEE Transactions on **2**(4), 448–458 (2014)
148. Xu, C., Zhang, Y., Yu, Y., Zhang, X., Wen, J.: An Efficient Provable Secure Public Auditing Scheme for Cloud Storage. KSII Transactions on Internet and Information Systems **8**(11), 4226 – 4241 (2014)
149. Xu, G., Chen, C., Wang, H., Zang, Z., Pang, M., Jiang, P.: Two-Level Verification of Data Integrity for Data Storage in Cloud Computing. In: Advanced Research on Electronic Commerce, Web Application, and Communication, *Communications in Computer and Information Science*, vol. 143, pp. 439–445. Springer Berlin Heidelberg (2011)
150. Yang, C., Lai, J.: Protecting Data Privacy and Security for Cloud Computing Based on Secret Sharing. In: International Symposium on Biometrics and Security Technologies, ISBAST 2013, 2-5 July, 2013, Chengdu, Sichuan, China, pp. 259–266 (2013)
151. Yang, J., Chen, Z.: Cloud Computing Research and Security Issues. IEEE, Computational Intelligence and Software Engineering (CiSE) (2010)
152. Yao, A.: How to generate and exchange secrets. In: Foundations of Computer Science, 1986., 27th Annual Symposium on, pp. 162–167. IEEE (1986)
153. Yi, X., Paulet, R., Bertino, E.: Homomorphic Encryption and Applications. Springer Briefs in Computer Science. Springer (2014)
154. Young, A., Yung, M.: Malicious Cryptography: Exposing Cryptovirology. John Wiley & Sons (2004)
155. Zhang, Y., Juels, A., Reiter, M.K., Ristenpart, T.: Cross-VM Side Channels and Their Use to Extract Private Keys. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS '12, pp. 305–316. ACM, New York, NY, USA (2012)
156. Zhao, F., Li, C., Liu, C.F.: A Cloud Computing Security Solution Based on Fully Homomorphic Encryption. In: Advanced Communication Technology (ICACT), 2014 16th International Conference on, pp. 485–488

(2014)
157. Zheng, Y., Yuan, X., Wang, X., Jiang, J., Wang, C., Gui, X.: Enabling Encrypted Cloud Media Center with Secure Deduplication. In: Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '15, pp. 63–72. ACM (2015)
158. Zhou, M., Zhang, R., Xie, W., Qian, W., Zhou, A.: Security and Privacy in Cloud Computing: A Survey. In: Semantics Knowledge and Grid (SKG), 2010 Sixth International Conference on, pp. 105–112 (2010)
159. Zhou, Y., Evans, D.: Ssoscan: Automated testing of web applications for single sign-on vulnerabilities. In: 23rd USENIX Security Symposium (USENIX Security 14), pp. 495–510. USENIX Association, San Diego, CA (2014). URL https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/zhou