The Cloud Security for SPI model

Chetna Kachhwaha
Department of Computer Application
Jodhpur National University
Jodhpur, India
chetna_1978@yahoo.com

***Abstract*—The main concerns which call for a sophisticated threat model is data breach, data loss, account hijacking, insecure API's, and denial of service in the post PC era where mobility, social, big data, cloud is creating boundary less organizations. The persistent and evolving threat landscape has created a need for smarter security solutions, and organizations are looking at security compliance very seriously than ever.**

A threat model helps in analyzing a security problem, design mitigation strategies, and evaluate solutions. The key security constructs on the basis of which security policies for a given threat model will be defined and enforced are infrastructure, information, identity, and end-user devices. Residing on a combination of public clouds and on-premise virtualized infrastructure, workloads are decoupled from their underlying infrastructure. In the borderless enterprise, flexible and secure information controls require policies that use rich information classification models, federated identities, and context–based authorization.

By building on what is already in place today, it is possible to forge a path to public, private, and hybrid clouds—often with a greater ability to both assert information controls and validate their effectiveness. Each stage in the journey to the cloud has a unique set of concerns and recommendations for addressing them. People, processes, and technical controls must evolve consistently to address these concerns. Security and compliance must be addressed in a unified and consistent fashion across physical, virtual and cloud environments. Security regains control and visibility as we move from virtualization towards public cloud and so is the efficiency and agility. The security can be implemented/ hardened by embracing secure-by-design approach, implementing an active monitoring solution and managing identities.

Keywords—Cloud security;SaaS;PaaS; IaaS

I. INTRODUCTION.

Cloud by definition is a model for enabling convenient, on demand network access to shared pool of configurable resources like networks, servers, storage, application and services that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Cloud security is a combination of set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing. The security issues can be broadly classified into two broad categories:

- Security issues faced by cloud providers (organizations providing software-as-a-service, platform-as-a-service, infrastructure-as-a-service via the cloud)
- Security issues faced by customers/ clients

.

II .SECURITY MECHANISM USED FOR SAAS CLOUD SERVICE MODEL

SaaS users have less control over security among the three fundamental delivery models IaaS, PaaS and SaaS in the cloud. The security concerns with SaaS are:

A.Data Security

Data security means ensuring data against corruption and controlling access to data. In a traditional on-premise application deployment model, the sensitive data of each enterprise resides within the enterprise boundary which provides its physical, logical and personnel security and access control policies. But, in the SaaS model, the enterprise data is stored outside the enterprise boundary and the interactions are beyond the firewall. This enforces SaaS vendor to adopt additional security checks to ensure data security and prevent threats due to security vulnerabilities in the application or through malicious users.

SaaS provider uses strong encryption and fine-grained authorization techniques to control access to data during processing. All access to data, including administrative access, should be logged and routinely audited. To encrypt data, information must be encoded in such a way that only the customer or computer with a key can decode it. Two different types of encryption can be used:

Symmetric-key

Public-key

For instance, the administrators of Amazon Cloud (Elastic Compute Cloud (EC2) does not have access to customer instances and cannot log into the Guest OS. EC2 Administrators are required to use their individual cryptographically Strong Secure Shell (SSH) keys to gain access to a host .On the other hand the data at rest in Simple Storage Service (S3) is not encrypted by default, users can encrypt their data before it is uploaded to Amazon S3, so that it is not accessed or tampered with by any unauthorized party.

B.Data Segregation:

The application instances and data stores are shared across multiple enterprises in a multi-tenant SaaS architecture. In such a situation, data of various users reside at the same location. Intrusion of data of one user by another is possible in such an environment. This intrusion can be done either by hacking through the loop holes in the application or by injecting client code into the SaaS system.

Thus sufficient security checks need to be adopted to ensure data security and prevent unauthorized access to data of one tenant by users from other tenants. This is achieved by hardening the data store as well as the application to ensure the data segregation. A SaaS model should therefore ensure a clear boundary for each user's data. The boundary must be ensured not only at the physical level but also at the application level. The service should be intelligent enough to segregate the data from different users

Amazon's S3 APIs provide both bucket-level and object-level access controls, with defaults that only permit authenticatedaccess by the bucket and/or object creator. Write and Delete permission is controlled by an Access Control List (ACL) associated with the bucket. Permission to modify the bucket's ACL is itself controlled by an ACL, and it defaults to creator-only access. Therefore, the customer maintains full control over who has access to their data.

C. Deployment Model:

SaaS security also depends upon the deployment model being used by the vendor. SaaS vendors may choose to deploy the solution either by using a public cloud vendor or host it themselves. Dedicated public cloud providers such as Amazon helps to build secure SaaS solutions by providing infrastructure services that aid in ensuring perimeter and environment security. This involves the use of firewalls, intrusion detection systems, etc. These, however, should be configured to ensure maximum security. A self-hosted SaaS deployment, however, requires the vendor to assemble the services like firewalls, intrusion detection systems, etc and hardens it themselves.

D. Deployment Environment:

The SaaS applications that are deployed on the public cloud should harden their application security settings to conform to the best practices recommended by the public cloud vendor. A third-party security audit of the SaaS application deployment can be conducted regularly to identify any security issues or threats to ensure the safety of the enterprise data.

Cloud providers like Amazon and Google help facilitate building secure SaaS applications by providing infrastructure services that aid in ensuring data security, network security, data segregation, etc.

E.Network Security:

In a SaaS deployment model, sensitive data is obtained from the enterprises, processed by the SaaS application and stored at the SaaS vendor end. This data which flows over the network must be secured in order to prevent leakage of sensitive information. Sufficient safeguards should be adopted against network security issues such as man-in-the-middle (MITM) attacks, IP spoofing, port scanning, packet sniffing, etc. The safeguards involve the use of strong network traffic encryption techniques such as Secure Socket Layer [SSL] and the Transport Layer Security [TLS] for security.

For Amazon Web Services [AWS], the network layer provides significant protection against traditional network security issues. To provide maximum security, Amazon S3 is accessible via SSL encrypted endpoints. These encrypted endpoints are accessible by the Internet and from within Amazon EC2, ensuring that data is transferred securely both within AWS and to and from sources outside of AWS

F. Regulatory Compliance:

The periodic third-party governance, risk, and compliance (GRC) audit of the SaaS application is critical for assessing the conformance to regulatory standards. This helps in identifying any compliance issues and also in ensuring that correct business processes are in place.

The SAS 70 standard includes operating procedures for physical and perimeter security of data centers and service providers. Access, storage, and processing of sensitive data needs to be carefully controlled and is governed under various regulations such as ISO-27001, Sarbanes-Oxley Act [SOX], Gramm-Leach-Bliley Act [GLBA], Health Insurance Portability and Accountability Act [HIPAA] and industry standards like Payment Card Industry Data Security Standard [PCI-DSS].

Data privacy is another significant challenge. Different countries have different privacy regulations about how data needs to be secured and stored. These regulations might lead to conflicts when the enterprise data of one country is stored in data centers located in another country.

G. Availability:

The SaaS application needs to ensure that enterprises are provided with service around the clock. This involves making architectural changes at the application and infrastructural levels so that they are resilient to hardware/software failures as well as to denial of service attacks. A multi-tier architecture needs to be adopted, supported by a load-balanced farm of application instances, running on a variable number of servers. Resiliency to hardware/software failures, as well as to denial of service attacks, needs to be built from the ground up within the application.

At the same time, an appropriate action plan for business continuity [BC] and disaster recovery [DR] needs to be considered for any unplanned emergencies. This is essential to ensure the safety of the enterprise data and minimal downtime for enterprises.

For Amazon, the AWS API endpoints are hosted on the same Internet-scale, world-class infrastructure that supports the Amazon.com retail site. Standard Distributed Denial of Service [DDoS] mitigation techniques such as synchronized

cookies and connection limiting are used. To further mitigate the effect of potential DDoS attacks, Amazon maintains internal bandwidth that exceeds its provider-supplied Internet bandwidth.

H. Backup and Recovery :

Enterprises must assure that SLA contains guarantees for covering the backup and recovery services. To facilitate disaster recovery and mitigate risks against the loss of sensitive data due to failures it is necessary that the proven backup and recovery services at the SaaS application, infrastructure and the cloud level are present.

The SaaS vendor must ensure that all sensitive enterprise data is regularly backed up to facilitate quick recovery in case of disasters. This backup data should be protected, similar to operational data, using strong encryption mechanism which helps to reduce the risk of unauthorized access and leakage of sensitive data.

Cloud vendors such as Amazon do not encrypt the data at rest in S3 by default. It is user's responsibility to protect the data by separately encrypting their data and backups so that it cannot be accessed or tampered with by unauthorized parties.

I. IdM and SSO

SaaS vendors provide the complete stack of Identity management IdM and sign-on services. For this all the user information, passwords, etc., are completely maintained at the SaaS vendor's site and must be securely stored and processed. The vendor should support the password strength and password expiration policies of the enterprise to comply with regulatory demands.

Identity management (IdM) and sign-on components ensure services for user-account processing, password management, and secure authentication. The security challenges differ depending upon the IdM and sign-on model used.

Alternatively, a SaaS vendor can also provide support for user account and credential replication. For this scenario the user account processing is done separately by each provider's customer within the customer's own boundary. Relevant portions of user-account information are replicated to the SaaS vendor for authentication and authorization capabilities. The SaaS vendor needs to ensure the sanctity of these credentials and prevent their leakage.

In a federated IdM model, the entire user-account information (including credentials) is managed and stored independently by each customer. The user authentication occurs within the enterprise boundary. The identity of the user as well as certain attributes are propagated to the SaaS vendor using federation for authentication and access control on demand .To assure secured federation of user identities the SaaS vendor and tenants need to ensure that proper trust relationships and validations are established.

III .SECURITY MECHANISM USED FOR PAAS CLOUD SERVICE MODEL

PaaS is a category of cloud computing that provides a platform and environment to allow developers to build applications and services over the internet. PaaS does offer security concerns that include:

- Access and authorization issues
- Distributed application issues
- Storage and data security issue

A. Authentication, Access Control and Authorization (AAA):

PaaS provides a shared development environment, thus authentication, access control, and authorization mechanisms are combined to assure that customers are kept completely separate from each other. A strong and effective authentication framework is necessary to ensure that individual users can be correctly identified without the authentication system succumbing to the numerous possible attacks.

Two factor authentications such as smartcards and biometric mechanisms also help to provide increased protection from various attacks but at the expense of greater complexity and longer provisioning cycles. Mostly PaaS vendors rely on user name and passwords for authentication and then implement a mechanism that provides access control to data and application-level authorization based on verification of the given credentials. They may also use some technique for enhancing the security of the authentication process, such as only requesting three characters from the password or answering a "secret question" or identifying a pre-agreed upon image.Irrespective of the authentication mechanism used, it is necessary that an end-to-end encryption is applied to the logon sequence. This authentication can be done by using a cryptographic hashing mechanism so that the password itself is never exposed.

Authorization is applicable at the application level which provides a confirmation that a user, computer, device or assembly has the required permission to carry out an operation. Authorization is carried out through a roles-based framework, where in user accounts are assigned to roles. Role-based assignment provides flexibility, to dynamically assign user accounts to different roles .Thus as users move around the organization, they automatically receive the rights they need to carry out their roles.

B. Distributed Application:

A well-architected distributed application should provide the transition from on-premises environment to cloud-based configuration without any issues. But poor architecture and design can result in performance degradation and week security which may result into compromise of few or all components of the application.

IJERT

Performance degradation comes from design errors like querying of parameters excessively or by using object properties heavily .In cloud-based environments, such issues cause the application to perform very slowly.

Rigorous enforcement of security standards are applied to distributed applications operating entirely within a corporate network, still these security standards for legacy systems are often less than ideal. While the risk/benefit analysis applied legacy designs are considered acceptable within the confines of the corporate network, but such implementations are considered unsupportable in the cloud.

The cloud applications need more wide-reaching authentication setup, with requirements for a federated, claims-based mechanism to establish identity and then using those claims-based identities to authenticate to the different tiers of the application. Thus an application ported into the cloud must manage exceptions and provide a mechanism for error reporting, to avoid the diagnostic challenges.

Any distributed application that is to be ported to cloud must be thoroughly assessed for design and security flaws. The cloud providers provide advantages of scalability and agility; still the applications need to be tuned, instrumented, synchronized and secured according to the realities of the PaaS cloud environment.

C. Storage and Data Security:
Data is stored in encrypted form with a Public Cloud Vendor, transported in encrypted form, processed behind the firewall of the customer and stored on the Public Cloud without unencrypted data being exposed on the Internet at any time. Thus the primary risks come at the point of processing the data, not at the point of storage.

So it is recommended that the sensitive data should not be sent to, or processed in, the Public Cloud to avoid the vulnerability during data processing. Organizations like, IBM have developed a homomorphic encryption algorithm that performsthe mathematical operations on the data without the software knowing the unencrypted data at any point.

*IV.*SECURITY MECHANISM USED FOR IAAS CLOUD SERVICE MODEL:

IaaS is another service model which provides neither a finished service nor a development platform. Rather, IaaS provides thecore compute, network and storage infrastructure which helps to build our own PaaS or SaaS environments. IaaS provides a way to deploy virtualized servers in the cloud by taking an advantage of server virtualization and automation. IaaS helps to reduce the physical server footprint in our datacenter and save on energy costs also. But along with all this there are security implications with IaaS model.

The various security issues with IaaS are:

- Data leakage protection and usage monitoring
- Authentication and authorization
- End to end logging and reporting
- Infrastructure hardening
- End to end encryption

A. Data leakage protection and usage monitoring issue:

Data can be stored in Iaas infrastructure in both public and private clouds, but when we are deploying IaaS in the public cloud we need to know who is accessing the information, how the information was accessed (from what type of device), the location from which it was accessed (source IP address), and what happened to that information after it was accessed (was it forwarded to another user or copied to another site)?

To solve such problems we can use modern Rights Management services and can apply restrictions to all information that is considered business critical. We can create various policies for such critical information and then deploy those policies such that it doesn't require any user intervention. Additionally, we should create a transparent process that can control the activities like who can see the critical information and then create a "self-destruct" policy for sensitive information that does not need to live indefinitely outside of the confines of the corporate datacenter.

B. Authentication and authorization:
For an effective Data Loss Prevention (DLP) solution we need to have robust authentication and authorization methods. We know that user name and password is not the most secure authentication mechanism. We can consider two factor or multi-factor authentication for all information that has to be restricted. Also we can consider tiering our access policies based on the level of trust we have for each identity provider for our IaaS cloud solutions. Finally we can integrate this authorization tiering into your DLP solution.

C. End to end logging and reporting
Effective deployment of IaaS, both in the private and the public cloud, demands for a comprehensive logging and reporting in place. Virtual machines spun up automatically and are moved between servers in an array dynamically over time, we never know where the information is present, at any place in time. Thus to keep a track of where the information is residing , who accesses it, which machines are handing it, and which storagearrays are responsible for it, we require a robust logging and reporting solutions.

The logging and reporting solutions are critical for service management and optimization, also there criticality rises in the event of a security breach. Logging is critical for incident response and forensics – and the reports and findings after the incident are going to depend heavily on our logging infrastructure. We must assure that all compute, network, memory and storage activity is logged and that the logs are stored in multiple, secure locations with extremely limited

IJERT

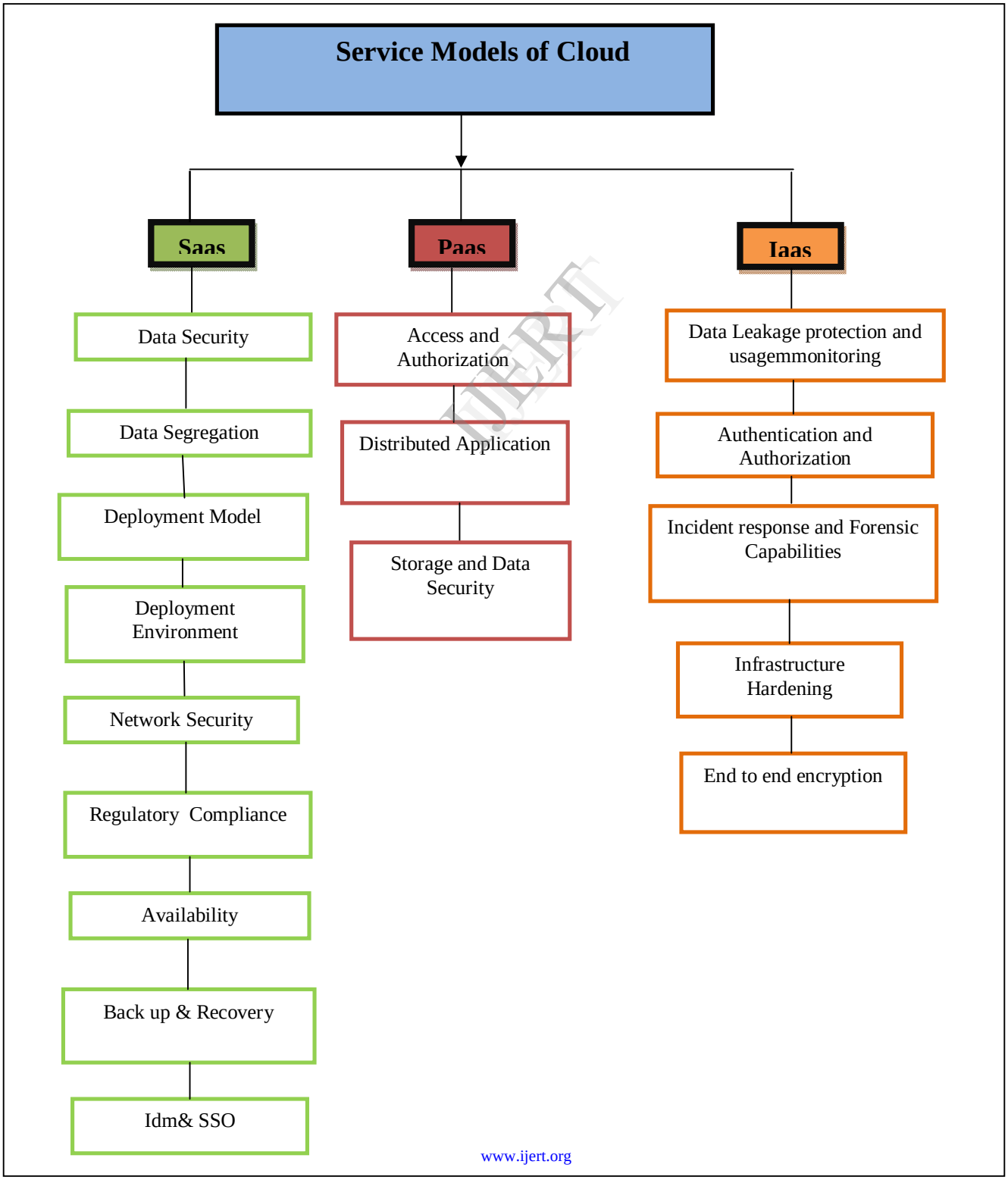
access. Also we should ensure that the principle of least privilegedrives our log creation and management activities.

D. Infrastructure hardening:

We must assure that our "golden image" virtual machines and VM templates are hardened and clean. This can be done with initial system hardening when we create the images, also we can take advantage of technologies that enables us to update the images offline with the latest service and security updates. We must assure that we have a process in place to test the security of these master images on a regular basis to confirm that there has been no drift from our desired configuration, either due to malicious or non-malicious changes from the original configuration.

E. End to end encryption:

IaaS as a service, both in public and private clouds takes the advantage of encryption from end-to-end. We must assure thatat we are using whole disk encryption, which ensures that all data on the disk, not only the user data files, are encrypted. This also prevents offline attacks. Additionally we must assure that all communications to host operating systems and virtual machines in the IaaS infrastructure are encrypted. This can be done over SSL/TLS or IPsec. This includes not only communications from management stations, but also communications between the virtual machines themselves .Also we should deploy mechanisms such as homomorphic encryption to keep end-user communications safe and secure. This is a form of encryption that allows complex calculations to be performed on the encrypted data also.



REFERENCES

[1] http://en.wikipedia.org/wiki/Cloud_computing_security

[2] http://www.infosectoday.com/Articles/Securing_SaaS_Applications.htm

[3] Guide for companies on cloud security and privacy implilcations.pdf

[4] http://searchcompliance.techtarget.com/definition/cloud-computing-security

[5] http://www.saas-tenant.com/white-paper/Securing-SaaS-Applications.htm

[6] http://esj.com/Articles/2010/02/09/Cloud-SaaS-Security.aspx?Page=3

[7] Cloud Security A comprehensive guide to Secure Cloud Computing by Ronald L. Kurtz and Russell Dean Vines

[8] Moving to The Cloud byDinkarSitaram and Geetha Manjunath.

[9] http://social.technet.microsoft.com/wiki/contents/articles/3808.security-considerations-for-infrastructure-as-a-service-iaas.aspx

[10] http://www.windowsecurity.com/articles-tutorials/Cloud_computing/Security-Considerations-Infrastructure-Service-Cloud-Computing-Model.html

IJERT