International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 1 | ISSN : 2456-3307

Cloud Computing and SaaS (ERP) Implementation

Sayali A. Ambavane, Ajay S. Pawar, Vivek H. Verma, Pallavi Marathe

ABSTRACT

During the last two decades, the use of Internet has been changing every domain of technology. It has also led to the tremendous development and implementation of cloud computing from the last few years. But the shared nature of data in the cloud makes it prone to security attacks. Different security techniques should be implemented to prevent security breaches. Authentication is one such technique which plays a major role in Cloud Computing security. Various possible security attacks on the Cloud Service Providers (CSP) are prevented by applying different authentication mechanisms, which verifies a user's identity when a user demands services from cloud servers. There are multiple authentication technologies for verifying the identity of a user before granting access to resources. In this paper we have discussed services provided by cloud and its brief analysis. Our mainly focus is on SaaS service which includes security issues and its solutions. We have mentioned various security prevention techniques which need to be considered when we want to implement SaaS.

I. INTRODUCTION

In a cloud based computing infrastructure, [1] the resources are normally in someone else's premise or network and accessed remotely by the cloud users. Processing is done remotely implying the fact that the data and other elements from a person need to be transmitted to the cloud infrastructure or server for processing; and the output is returned upon completion of required processing. The study represented in this paper is based on the services provided by the cloud. Also we have discussed various approaches to cloud computing as well as the security issues and concerns that must be included in the deployment towards a cloud based infrastructure. Cloud computing can serve a diverse range of functions over the Internet like storage and virtual servers; applications and authorization for desktop applications. By taking advantage of resource sharing, cloud computing is able to achieve consistency and economies of scale. The types of cloud computing are classified based on two models. Cloud computing service models and cloud computing deployment models.

II. CHARACTERISTICS OF CLOUD COMPUTING

[1]The essential characteristics of the cloud computing model were defined by the National Institute of Standards and technology (NIST) and have since been redefined by number of architects and experts. [3]According to NIST, Cloud computing is 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.

1. **On-demand self-service** - A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

2. **Broad network access** - Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
3. **Resource pooling** - The provider's computing resources are pooled to serve multiple consumers using a multi- tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacentre). Examples of resources include storage, processing, memory, and network bandwidth.
4. **Rapid elasticity** - Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
5. **Measured service –** [2] Cloud systems automatically control and optimize resource use by leveraging a metering capability1 at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

Cloud Computing offers great benefits for organization and individuals by reducing cost and increasing flexibility. [3] A good and robust Cloud Computing model should consist of all these five essential characteristics. There are also privacy and security concerns. If you are building Cloud service or considering public Cloud, you should think also about how your organization and your customer's data can be protected. Carefully review the terms of service or contracts, and challenge the Cloud provider to meet your needs.

III. CLOUD DEPLOYMENTS MODELS

In the cloud deployment model, networking, platform, storage, and software infrastructure are provided as services. The Cloud Computing model has three main deployment models which are:

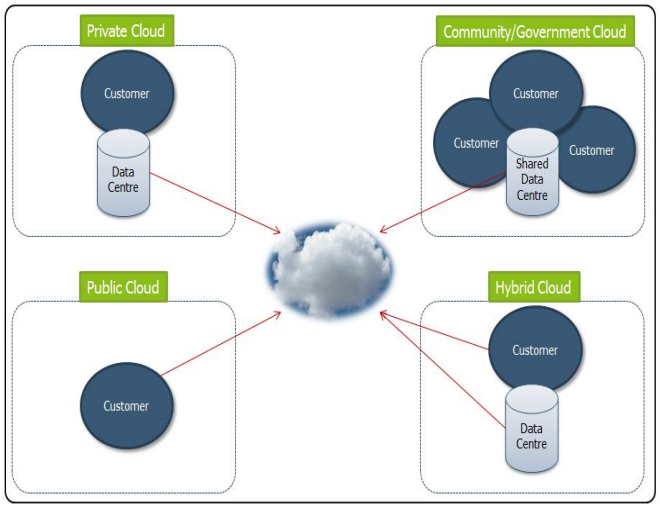

Figure 1. Cloud Computing Deployment Models

Public cloud

The most common and well-known deployment model is Public Cloud. A Public Cloud is a huge data centre that offers the same services to all its users. [2]The services are accessible for everyone and much used for the consumer segment. Examples of public services are Facebook, Google and LinkedIn. For consumers, Public Cloud offerings are usually free of charge, for professionals there is usually a per-per-use (or user) pricing model. [2]The Public Cloud is always hosted by a professional Cloud supplier. Public clouds are less secure than the other cloud models because it places an additional burden of ensuring all applications and data accessed on the public cloud are not subjected to malicious attacks.

Private cloud

The other commonly used deployment model is Private Clouds. There are lots of discussions for how strict the definition of Private Clouds should be.

[3]In general a customer's internally hosted data centre is regarded as a Private Cloud. If we add virtualization and automation, such a setup may very well be regarded as a Private Cloud. A professional Cloud vendor may also offer a Private Cloud to their customers by supporting a separate hardware environment in the data centre.

In the private cloud, scalable resources and virtual applications provided by the cloud vendor are pooled together and available for cloud users to share and use. It differs from the public cloud in that all the cloud resources and applications are managed by the organization itself, similar to Intranet functionality. [4]Utilization on the private cloud can be much more secure than that of the public cloud because of its specified internal exposure. Only the organization and designated stakeholders may have access to operate on a specific Private cloud. A Private Cloud is therefore mostly suited for sensitive data, where the customer is dependent on a certain degree of security. Private Clouds, to a certain degree, loose the economy of scale compared to a Public Cloud.

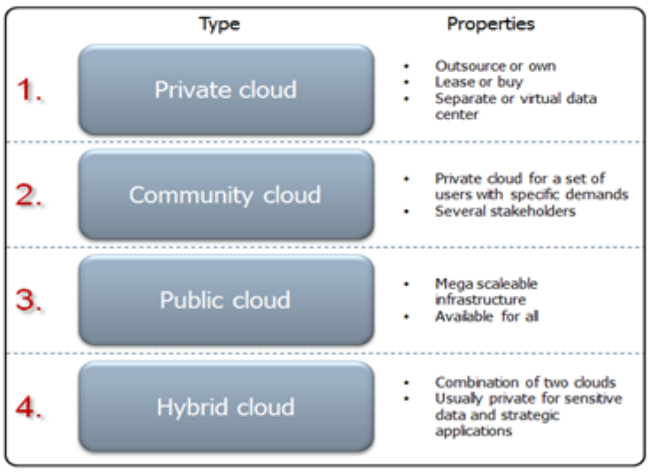

Figure 1(a). Cloud Deployment Model Properties

Hybrid cloud

The Hybrid Cloud is a combination of both Private and Public. This is a setup that is much used for large companies. [4]Hybrid cloud is a private cloud linked to one or more external cloud services, centrally managed, provisioned as a single unit, and circumscribed by a secure network. Hybrid Cloud provides more secure control of the data and applications and allows various parties to access information over the Internet. It also has an open architecture that allows interfaces with other management systems.

Hybrid cloud can describe configuration combining a local device, such as a Plug computer with cloud services. It can also describe configurations combining virtual and physical, collocated assets -for example, a mostly virtualized environment that requires physical servers, routers, or other hardware such as a network appliance acting as a firewall or spam filter. [5]Vital data is usually preferred in a Private Cloud and supporting services in Public, for instance search, email, blogs, CRM etc. In other words strategic applications are run separately.

Virtual Private

A Private Cloud offspring is the Virtual Private Cloud. [2]This is a virtual, and not physically, separated Cloud offering normally run in a Public Cloud centre. Access is given through a secure connection, i.e. VPN, and access may also be restricted by the physical location of the user, i.e. within the customer's firewalls.

IV. SERVICE MODELS

Infrastructure as a Service (IaaS)

Infrastructure as a service (IaaS) refers to online services that provide high-level APIs used to dereference various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc. A hypervisor, such as Xen, Oracle Virtual Box, Oracle VM, KVM, VMware ESX/ESXi, or Hyper-V, LXD, runs the virtual machines as guests. Pools of hypervisors within the cloud operational system can support large numbers of virtual machines and the ability to scale services up and down according to customers' varying requirements. [6]The capability provided to the consumer is the provision of grids or clusters or

virtualized servers, processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems. The highest profile example is Amazon's Elastic Compute Cloud (EC2) and Simple Storage Service, but IBM and other traditional IT vendors are also offering services, as is telecom-and-more provider Verizon Business.

Platform as a Service (PaaS)

Cloud computing has evolved to include platforms for building and running custom web-based applications, a concept known as Platform-as-a-Service. PaaS is an outgrowth of the SaaS application delivery model. [7]The PaaS model makes all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet, all with no software downloads or installation for developers, IT managers, or end users. Examples include Microsoft's Azure and Salesforce's Force.com.

Software as a Service (SaaS)

The traditional model of software distribution, in which software is purchased for and installed on personal computers, sometimes referred to as Software-as-a-Product. [7]Software-as-a-Service is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. SaaS is becoming an increasingly prevalent delivery model as underlying technologies that support web services and service-oriented architecture (SOA) mature and new developmental approaches become popular. SaaS is also often associated with a pay-as-you-go subscription licensing model. Mean-while, broadband service has become increasingly available to support user access from more areas around the world. [8]SaaS is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the internet. Examples include G Suite - formerly Google Apps- Microsoft Office 365, Salesforce and Workday.

In the Software as a Service (SaaS) approach, applications are delivered over the Internet in the form of service. Rather than installing and maintaining the software, one simply needs to access the software over the Internet.

Any SaaS model should have the following key characteristics:

- **Multitenant Architecture** – In a multitenant architecture, multiple users and applications share a common source code. This source code is maintained centrally in one location.
- **Customization** – Since the source code is maintained in a single place, it becomes easier to customize the application based on the business needs of the customer. SaaS is designed and organized in such a fashion that these customizations can easily be managed and maintained per customer.
- **Accessibility** – SaaS provides better access to data over the Internet. This makes it easier to manage privileges or monitor data usage. It also ensures that same information is available to all users at any point of time.

Figure 2. SaaS (Cloud Service)

When implementing the SaaS model, the following steps should be followed:

Understand the business requirements

Irrespective of the technology or the model, it is very important to have a clear understanding of the business requirements. Without this, we won't be

able to design and develop any system or application. [9]In order to achieve better results, it is important to identify the goals and objectives at a very early stage. The investigation and discovery process should be robust enough to set the goals and directives at a very early stage. The investigation process should determine the following:

- How should the application be designed to run?
- What are the different categories of users who will access the application?
- How should the application respond to:
  - ✓Scalability
  - ✓Security
  - ✓Failover issues

It is very important to identify and understand the characteristics of the application at a very early stage. Not only that, we should pay equal attention to identifying the challenges that the existing application, system or the process is facing.

Identify the team to take on the task

Again, irrespective of the technology or the model, it is very important that the team assigned to take on the task is well versed in the technology and the concepts. In the SaaS model we should have a team comprised of seasoned developers who understand the concept of SaaS in depth. The team should have members who have the expertise of multiple technologies and also should be well aware of the best practices that are followed in the industry.

Design a scalable infrastructure

Once the team has the complete understanding of the business requirements, the next step is to build the infrastructure along with the following components:

- ✓ Data canter
- ✓ Network infrastructure – connectivity and security
- ✓ Hardware – both systems and storage
- ✓ Backup and monitoring tools

[10]On top of these, there should be internal reviews to evaluate the cost-benefit-related issues while building the infrastructure. While finalizing decisions on infrastructure, one must take the following into consideration:

- Service level agreement (SLA)
- Scalability, availability and other performance factors
- Customer support and incident reporting
- Disaster recovery
- Network bandwidth
- Security management

Finalize the bandwidth requirement and hosting facility

It is very important that the infrastructure is hosted within a facility which has a public connectivity and maintains consistency to ensure positive user experience. While reviewing the bandwidth, we must think of the demographics of our application, e.g. the connectivity factor for a user sitting in an office where high bandwidth network speed is available would be different from a user who is connecting from home. It is also important that we place the infrastructure as close as possible to ensure fewer network hops. We should have multiple network connections to our data canter, thus eliminating network bottlenecks. If we decide to outsource the data canter infrastructure, we should consider the following:

- ✓ Is the data canter available 24×7×365?
- ✓ Testing frequency
- ✓ Availability of redundant systems for power and other hardware failures
- ✓ Physical security of the campus

Procuring the infrastructure components

Once the infrastructure design is complete, we need to use components which have proven reliability and functionality. This step is critical in order to ensure high availability. While evaluating these hardware components, we must also ensure that the selected hardware is delivered within the timelines of our business needs.

Deployment of the SaaS delivery infrastructure

Once the infrastructure components are available, the operation team should start building and deploying the SaaS components. Servers should be racked, configured and subsequently the operating systems should be installed as per the need. Security devices should be upgraded with the latest versions of IDS. The firewall should also be configured as per the user access policy of the business.

Plan for disaster recovery and continuity

Now that the application is ready to be used over the SaaS platform, we must plan for disaster recovery and ensure continuity of the application. The following questions need to be answered in this regard:

- ✓ How do we respond to a disaster condition?
- ✓ How do we bring back the application in a limited time frame?

Integration of a monitoring solution

A monitoring subsystem is vital. It helps to ensure timely intervention and avoid disasters. The system monitoring should be done based on the following parameters:

- ✓ Memory and CPU usages
- ✓ Event logs from the operating system and the application
- ✓ Different application components (TCP layer, database, application servers, etc.)

Prepare the customer support call centre

Once the application is out on the market, it must have a customer support call canter. The call centre should be well connected and equipped to manage an appropriate ticketing system. [10]Customer support is a key component to ensure success of any model or application irrespective of the technology. The ticketing system should be enabled with an appropriate emailing system; if any issue requires the attention of the development team, the ticketing system should be able to send emails to the appropriate team member.

Prepare the service level agreement (SLA)

An SLA must be in place while implementing the SaaS model. The SLA should clearly define the turnaround time and the response time along with the application availability.

Documentation

Once all of the above steps are completed, the entire infrastructure and its components must be documented. This document will help others to handle any exceptional behaviour of the application. It will also help if there are any modifications or alterations required in the infrastructure.

Virtualization and Software Delivery

[1]Virtualization encompasses various computing technologies and can be achieved both at the hardware level and at the software level. In an enterprise, virtualization can enhance the ability of software services, especially SaaS applications. It's also the most effective way for enterprises to reduce their IT costs. The concept of virtualization has been rightly adopted and accepted in the software development community. It has the ability to provide faster development and test mechanisms by creating development and test environments rapidly. [2]VMware and Virtual-Box are the most widely used technology, and they enable multiple users to run on different operating systems, versions and instances. Most software development enterprises adopt the virtualization technique by first adopting the software virtualization mechanism and then gradually moving toward hardware virtualization.

Virtualization and SaaS

In spite of having so many advantages, SaaS has many more helpful factors. These include:

- ✓ **Huge Start-Up Cost**: The revenue invested in the setup is recovered over a period of years.
- ✓ **It May Violate the Principles of Free Software**: [5]Software freedom activist Richard Stallman refers to SaaS as "service as a software substitute (SaaSS)," and considers it a violation of the principles of free software.

[5]"With SaaS, the users do not have a copy of the executable file: it is on the server, where the users can't see or touch it. Thus it is impossible for them to ascertain what it really does, and impossible to change it. SaaS inherently gives the server operator the power to change the software in use, or the users' data being operated on," Stallman wrote on the GNU website.A good example of SaaS over virtualization is Amazon Web Services (AWS). AWS offers a host

of software and platforms. The software is installed on virtual hosts and can be scaled up or down as and when required.

If we focus beyond the infrastructure and start-up cost, once deployed, an SaaS application platform should only be concerned with reproducibility. Each and every instance of the SaaS-based application should be identical to each other. There should be minimal differences in order to maintain the consistent behaviour of every application instance for each customer and for the support team. This is done so that they have a uniform base in order to troubleshoot any issue, if required. The support engineer would not like to discover a problem caused by a missing library module for a single customer instance. Similarly, neither would a customer like to know that there could be a problem in each application ordered because the SaaS-based company cannot reproduce the issue using the same steps for every order. The entire process should be automated for consistency and cost benefits.

Increasing complexity

It becomes important to understand the complex nature of deployment for today's applications be it the SaaS model or traditional model. Even the simplest Web application is no longer responsible for managing the underlying data storage layer. The standard practice is to have a database, for example, MySQL, Oracle, DB2 or SQL Server. Combining these with typical Web stacks such as Java, Ninja, Grails, Rails, etc., leads to a multi-tiered architecture demanding scalable deployment.

For example, while setting up a Rails environment, we used MySQL. The agile nature of applications, which allows for easy upgrades of the software via plugins, patches, macros and mashups, can easily be integrated into the SaaS model. An extension or a patch is developed for a smaller issue, most of the time a bug fix, which needs to be delivered as a patch on the exiting software. Usually a customer wouldn't like to hear that a problem occurred due to a resource constraint or some other circumstances, or that it is created by another customer. As per Wikipedia, separation of concerns is the premise to breaking down an application into distinct features, which minimizes functionality overlap. With virtualization in place, this concept can be applied to the infrastructure. Separation can be applied down to the per-application, per-customer, and/or per-cluster basis. While still using the hardware to its maximum capacity, it provides the ability to scale horizontally and vertically. This is beneficial for single-tenant applications that wish to enter the SaaS market.

SaaS Security Concerns

SaaS, sometimes referred to as on-demand software, is a model where software is licensed on a subscription basis and is centrally hosted. [11]Hackers are increasingly interested in not only breaking into your network but the value of the data they may find there. If the SaaS provider is compromised, data encryption is a good idea to help protect organizational data; however, it will not protect against phishing and malware attacks launched to steal individual user access credentials. [5]Encryption should be considered a "must have" technology; but organizations should remember that it, by itself, is not a vulnerable.

[5]The following is a brief listing of the top 10 security issues (by OWASP) that your SaaS offering should address:

1. SQL, operating system or LDAP injection
2. Insecure authentication and session management
3. Cross-site scripting because of lack of data validation
4. Insecure exposure to references like files and directories
5. Incorrectly configured (from a security perspective) databases, middleware and operating systems
6. Exposing sensitive data like user IDs, passwords and personal identification information
7. Checking for access inside the business logic on the server side

8. Cross-site request forgery
9. Using components with known vulnerabilities
10. Un-validated redirects and forwards

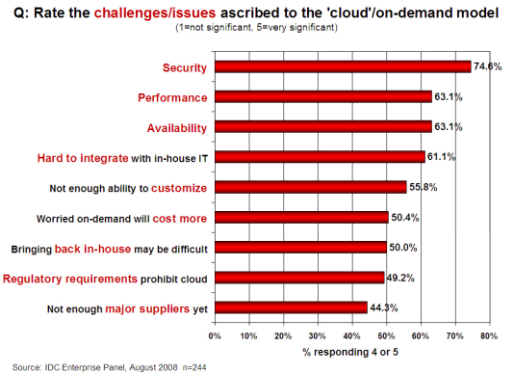

Figure 3. Factor which matters the most in cloud

Although SaaS providers must provide assurance that they are taking steps to mitigate breach risks, the responsibility for security cannot stop there. [12]Organizations that select SaaS solutions must also share security responsibility and implement internal procedures and processes. This includes education strategies to teach employees how to identify and respond to phishing campaigns, as well as setting company policies around what data should be placed in the cloud and what is better kept within the firewall. Just because an organization can store their data in the cloud doesn't mean that they should. [13]Organizations need to have a conversation with a trusted, knowledgeable partner to understand what (if any) data is best served on premise, in a hybrid setting, or totally "in the cloud" to understand the business and security consequences of doing so. Setting policies and best practices around what data may or may not need to be stored in the cloud can save numerous headaches, and potential data exposure and loss, later.

V. CONCLUSION

The Software as a Service (SaaS) model offers customers significant benefits, such as improved operational efficiency and reduced costs. However, to overcome customer concerns about application and data security, vendors must address these issues head-on. When it comes down to it, most enterprises' security concerns are centered on the lack of control and visibility into how their data is stored and secured with SaaS vendors. There is a strong apprehension about insider breaches, along with vulnerabilities in the applications and systems' availability that could lead to loss of sensitive data and money. Such challenges can dissuade enterprises from adopting SaaS applications.

The adoption of SaaS security practices (secure product engineering, secure deployment, GRC audits and regular SaaS security assessment) is vital to securing SaaS solutions. These can help identify any security issues upfront and ensure the safety of the data. SaaS vendors will benefit from the improved security of the solution and third-party validation of their security in the form of shortened sales cycles, and reduced operational risk. These measures will help them better answer any sales and marketing queries about security and address customer concerns. Customers will further be benefitted and assured about the security of their sensitive data and have higher confidence in the SaaS vendor.
Thus, adoption of the above SaaS security strategies and regular SaaS security assessment can enable SaaS vendors to boost customer confidence in the security of their solution and enable its faster and wider adoption.

VI. REFERENCES

[1]. wikipedia.org/wiki/Cloud_computing
[2]. www.incapsula.com/cloud
[3]. Selected aspects of security mechanisms for cloud computing – current solutions and development perspectives by Aneta Poniszewska-Maranda
[4]. http://ecsnamagazine.arrow.com/saas-paas-and-iaas-what-you-and-your-customers-need-to-know-about-the-risks/
[5]. https://www.techopedia.com/2/31037/trends/how-virtualization-can-empower-saas-applications

[6]. https://www.owasp.org/index.php/Cloud-10_Risks_with_SaaS
[7]. http://www.cio.com/article/2435262/enterprise-software/gartner--seven-cloud-computing-security-risks.html
[8]. Software as a Service (SaaS): Security issues and Solutions by Navneet Singh Patell, Prof. Rekha B.S.2
[9]. Cloud Computing Security Issues and Challenges by Kuyoro S. O. Department of Computer Science Babcock Uni versity Ilishan-Remo, 240001, Nigeria
[10]. Enhanced Security Mechanisms for Cloud Computing by Himanshu V. Taiwade Dept of Computer Technology, PIET, Nagpur, India
[11]. S Subashini, V Kavitha, "A survey on security issues in service delivery models of cloud computing", Journal of Network and Computer Applications, vol. 34, pp. 1-11, Jan 2011, Elsevier Ltd.
[12]. Qi Zhang, Lu Cheng, and Raouf Boutaba, "Cloud Computing: state-of-the-art and research challenges", Journal(Springer) of Internet Services and Applications, vol.1, issue 1, pp.7-18, May 2010.
[13]. Nabil Sultan, "Cloud Computing for education: A new dawn?" International Journal of Information Management, Elsevier Limited 2009.