CHAPTER 3

CLOUD PLATFORM CREATION AND
SERVICE DEPLOYMENT

3.1 INTRODUCTION

Web has become an open platform where applications are built and run from anywhere. Infrastructure-as-a-Service (IaaS) is the delivery of computer infrastructure such as servers, network, applications, storage, software, hardware as a service. The customer is charged only for the resources consumed like utility based computing. Virtualization is a technique to implement cloud computing resources such as platform, application, storage, and network. One of the advantages of virtualization solutions is that multiple Virtual Machines (VM) are created simultaneously on a single computer/server and well-known companies are investing on this technology. Few notable, important and popular cloud based products are g-mail and yahoo mail. Cloud based products will eliminate the need to install and manage client rich applications. A large number of small companies outsource their work to cloud providers, such that they concentrate only on their core business instead of concentrating on the hosting environment. The following section provides implementation details of creating a virtual platform using Windows and the process of deploying applications on Stax.
3.2 CREATING A VIRTUAL PLATFORM FOR CLOUD COMPUTING

3.2.1 Generalized Cloud Architecture

The Generalized Cloud Architecture is depicted in Figure 3.1. First layer consists of resources, second layer is the virtual management layer, third layer is cloud management layer, fourth layer consists of different type of web services and last one, top most layer facilitates end user connectivity using GUI/interfaces to the cloud.

<table>
<thead>
<tr>
<th>Cloud Security and Compliances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Client _ User Application and Services</td>
</tr>
<tr>
<td>Cloud programming, environments and tools, Compute, Database, Queuing, Storage, Identity Services, etc</td>
</tr>
<tr>
<td>SLA Management, Monitoring, etc</td>
</tr>
<tr>
<td>Virtual Machine (VM), VM Management and Deployment, hypervisor</td>
</tr>
<tr>
<td>CLOUD RESOUECES</td>
</tr>
<tr>
<td>Hardware, Software, legacy applications, storage, networks</td>
</tr>
</tbody>
</table>

Figure 3.1 Generalized cloud architecture

Cloud resources includes hardware, software, legacy application and network resources and this layer forms the basis of the cloud system. It forms the major operating components providing the elementary components for proper functioning of a cloud system. Virtual management layer consists of virtual machines, VM management and deployment and hypervisor. Creation of Virtual Machines for each user is performed on this layer. Further it deals
with functionalities required for deployment of the virtual machines and also provisions the management of these components.

Cloud Management layer consists of Management Console, QoS Negotiation, Admission Control, Pricing, SLA Management, Monitoring, Execution Management, Metering, Accounting, and Billing. It provisions the management of the resources provided by the cloud. It also performs management and authentication checks for the administrator, and usage accounting and billings for the user.

Fourth layer consists of Cloud Programming tools, Compute, Database, Queuing, Storage, Identity Services. It provides a programming environment for the developer’s code. The essential programming and services are provided to the users from this layer.

Topmost layer is the application layer, in which the users run their applications and services. The services are passed on directly to the programming layer, which runs the services in the corresponding virtual machines to provide appropriate results to the user. Security and Compliances are applied to all the layers.

To Deploy Web services, JDK™ 6 is used and JDK™ 6 Java EE 5 Development using GlassFish Application Server is used for the process. GlassFish, allow application developers to develop and deploy Java EE-compliant applications such as Deploying Enterprise Applications, Web Applications, EJB module, Connector module, Life cycle Module, Application client module and Custom MBeans.
3.2.2 Working Process

This session gives details about the installation of hypervisor and virtual machines. Any hypervisor can be installed depending upon the host operating system. Oracle Virtual Box is used as a hypervisor. After installing the hypervisor, any operating system images can be installed to run as a virtual machine.

3.2.3 Installing Hypervisor

The advantage of virtualization solutions is that multiple operating systems can be run simultaneously on a single host. Many hypervisors are available for different operating system. It is necessary to choose the type of file that is most suitable to the operating system.

3.2.4 Adding a Virtual Machine to Hypervisor

Hypervisor allows running guest operating systems using its own virtual computer system. Virtual Machine or the guest operating system runs as a real operating system. By changing the desired parameters such as OS details, system, display, storage, audio, network, serial ports, USB, Shared folders, the Virtual Machine operates according to the specification. These settings can be modified later depending on the necessities.

Next step is to add an open source operating system Fedora to the host machine using the image of fedora, by properly choosing the image of the operating system to be installed as a Virtual Machine to start the download process. Choice of architecture (i.e., 32-bit or 64-bit) should be selected as the next step. It is also mandatory to note down the password for root user and default user for the current virtual machine.
After the process of image extraction, the extracted folder contains two folders, Machines and VDI. The VirtualBox image for FreeDOS is available in the VDI folder.

To add the guest operating system to the VirtualBox host, The New Virtual Machine Wizard dialog box is used. The wizard is an easy-to-follow guided setup for installation of the guest operating system. The basic information and appropriate memory configurations parameters are provided. (Figure 3.2).

![Figure 3.2 Virtual hard disk dialog box](image)

Below in Figure 3.3, a sample interface is shown for installing Fedora. The Fedora.vdi file is selected for completing the installation. A summary screen appears and Fedora gets added to the list of operating systems installed (Figure 3.3).
Select the appropriate operating system (Fedora) and provide the authentication details to run Fedora.

Virtual Machines can access the real network via its virtual network card. It is even possible to give the host system, other guests, or computers on the Internet access to the guest system. The following GUI shows accessing Internet via Virtual Machine.
Figure 3.4 Accessing the internet through virtual machine

Cloud based computations also have their own drawbacks. This is due to the fact that a high speed dedicated internet connection is mandatory. Data is often the most valuable asset for a company and it must be protected with as much vigilance as any other asset. It is easy to argue that more vigilance is needed to protect data, an intruder can potentially reach a company’s data from anywhere on the Internet.

In the event of failure or outages, contingency plans must take effect smoothly, and for disastrous or catastrophic failure, recovery plans must begin with minimum disruption. Additional costs may be associated with the required levels of reliability. Hidden costs could include support, disaster recovery, application modification, and data loss insurance. It is not clear whether the cloud computing model provides adequate protection for privacy information.
Since cloud computing is necessarily a new model, there is a great deal of uncertainty about how security at all levels (e.g., network, host, application, and data levels) can be achieved. That uncertainty has consistently led information executives to state that security is their number one concern with cloud computing.

In the cloud computing world, there is variability in terms of where the physical data resides, where processing takes place, and from where the data is accessed. Given this variability, different privacy rules and regulations may apply. Cloud computing depends largely on global politics to survive. Politics are affecting the scalability of the Internet, the availability of Internet access, the free flow of information, and the cloud-based global economy on a daily basis.

3.3 CLOUD COMPUTING IN STAX PLATFORM

3.3.1 Stax

Stax is a cloud based application based on Java that provides Platform as a service. It uses Amazon’s EC2 servers to deploy Java Applications on cloud. Applications can be deployed in cloud that can be leveraged by a wide variety of users. When the number of requests increase, it becomes mandatory for the organization to scale up or upgrade their resources, if the company maintains the infrastructure within its premises. However, for an application deployed in cloud, this process becomes easy, since cloud is pay- as- you- go service.
Figure 3.5. **Cloud application deployment and access through cloud providers**

In Figure 3.5 a typical cloud application deployment process along with access mechanisms through cloud device providers is depicted. The end users are connected to the cloud via Internet using communication/computing devices.

A developer can register with their email to utilize the services in Stax. Stax is a Platform as a Service for Java backed by Amazon EC2 infrastructure. It is restricted to Java and makes Java development, deployment, scaling much faster and deploys the application in Amazon infrastructure, not in Stax infrastructure. Stax provides easy deployment of test and production environments, a local development model, and strong integration with existing development tools, frameworks, and processes. Stax is designed to provide a version of the Java web container (Tomcat) that is integrated with an elastic computing cloud. Stax goal is to help developers achieve a new level of application deployment flexibility which not only includes scaling apps up and down by deploying on elastic PaaS,
but also provide developers the flexibility to take their applications and deploy them into their own application containers.

Cloud needs to provide the features designed to help developers gather information about their running servers, by offering features such as reporting/analytics and monitoring/alert APIs to make it much easier for developers to debug and understand the health of their applications and respond accordingly. Of course at some level, developers will always have to understand the internals of their own applications, and logs can be an invaluable part of debugging those kinds of issues.

Stax Network offers a complete set of Web tools for creating and managing the projects. A wide variety of starting points is available from basic servlets to Apache Struts or Apache Wicket, all running on Tomcat. Stax also offers JRuby and Jython running on top of the same Java foundation. All can talk to MySQL databases running in the same cluster.

At this point in time, developers can download the code, build their application on their machine, and redeploy code with a command-line tool. Developer can move their application to a cluster with up to five servers with just a click of the button.

3.3.1.1 Stax advantages

The Stax service improves production application security and reliability by allowing us to run dedicated servers on our EC2 account. Stax servers running on the EC2 cloud account are used only by the deployed applications, so they are isolated from applications deployed by other Stax users and can take full advantage of the underlying computing resources.
Dedicated Cloud based Stax servers are still 100% managed by the Stax service and works with streamlined deployment and configuration tools. This means that deploying and managing Java applications on cloud is now as just simple as using the public Cloud, but with complete control over computing capacity and how the deployed applications are grouped together.

Increase application capacity, reliability and security by running your Java applications on your own cloud based Java servers. Applications are published as standard WAR files, and run in standard JVMs, so you can use any Java libraries and frameworks.

### 3.3.1.2 EC2 stax features

- **More memory and computing cycles** – Apps are free to consume all available compute cycles and are no longer limited to 256MB of RAM
- **No restrictions** – Apps running on dedicated servers are not subjected to the resource utilization and cluster size restrictions placed on applications running on regular public cloud.
- **Monitoring and alerting** – Apps could be monitored 24×7 and could be proactively fixed if application failures are detected.

### 3.3.2 Deploying Applications on the Stax

This subsection provides a guide to deploy an application in Stax Cloud. Stax provides an end-to-end application environment that lets
developers quickly build, deploy, and scale Java web applications using standard Java frameworks and APIs.

Developers are free to use any classes or libraries that can be packaged and run from a standard Java Web Archive (WAR file). Stax provides templates, which help developers get started with some of the most popular Java application frameworks. Developers can build applications using the Basic application template.

Stax applications use the standard Servlet and JDBC APIs, so applications are portable and are deployed into any J2EE environment. Any Java web application that can be packaged as standard J2EE WAR file can be deployed on Stax, however the Stax environment is a very strict implementation of the J2EE web container. Since applications are coded using standard Java technologies and APIs, developer can easily deploy their applications on their own J2EE application servers.

3.3.2.1 Openness or scalability

The developer can use all the standard APIs, write to disk, log into a shell account, or just enjoy the freedom to move their application to another provider all without rewriting the application. Stax is suitable for openness as well as scalability, since it deploys the application to multiple servers and load balancers. Stax offers more standard solutions that can easily be duplicated because they're just Tomcat and a database under the hood. The downside of this platform is that developers are expected to handle all the deployment issues by themselves, but it's feasible.

A developer registers for deploying their services in Stax Cloud. The Stax Development Kit (SDK) can be downloaded using which the
developer can write and deploy their application in cloud freely. After successful testing in the local machine, services can be deployed in cloud.

The following screenshots shows the implementation of simple calculator service deployed in cloud (Figures 3.6–3.12). Any commercial product can be deployed using the same logic.

![Figure 3.6 A simple calculator application](image)

After deployment, the link can be shared to the users of the service. E.g.: http://calci.loksram.staxapps.net/. Figure 3.6 shows the process of creating an application in Stax.

Figure 3.7 shows the input screen of the Calculator application. This application is created with two inputs and the button performs the processing.

Figure 3.8 shows the results displayed from the calculator application. The inputs given in Figure 3.7 are processed and the results are shown providing the sum, product, multiplication and division of the numbers provided.
Figure 3.7 A simple calculator application _input screen

Figure 3.8 A simple calculator application _output screen
The Figure 3.9 shows the initial screen of the database creation screen in Stax. It shows the creation of a sample database. The user is usually prompted for authentication during the creation phase.

Figure 3.9 Database Creation

Figure 3.10 Database connectivity
Figure 3.10 shows the connectivity screen of the MySQL query browser in Stax. The simple GUI can be observed, which enables ease of working for novice users, while advanced panel is available for power users.

Figure 3.11 Viewing deployed application

Figure 3.12 A sample application creation
The Figure 3.11 shows a sample screenshot for viewing deployed application. It shows the query result window for the MySQL database. The user has provisions to perform all the operations on the database right from the query window in the database. This helps in troubleshooting.

The Figure 3.12 shows steps to create an application and database. Similarly application developers can create any kind of services / web services that can be deployed on cloud. Once if the application crosses the free tier limitation, cloud user will be charged on the basis of pay-as-you-go service.

3.4 SUMMARY

This section contributes that, this thesis is on Policy Monitoring for Enhanced Secure Cloud Computing Services. Since various components of the cloud are new, it is felt the creation of a virtual platform for cloud service and deployment of a service in an external cloud service provider are mandatory for understanding the cloud service deployment procedure. Thus various important steps followed for cloud configuration with different suitable tools are explained. Next Chapter 4 discusses about the advantages of virtualization and highlighted how cloud security can be improved with the use of virtualization.