CSS
RATIONALIZE · OPTIMIZE · INNOVATE

Automated Application Provisioning for Cloud

Application Provisioning in Cloud requires mechanism to automate and repeat as and when it requires. This is mainly because the building blocks of an IT infrastructure such as computing, storage, network, security is split into different building blocks and converted into a set of services. Now you can reach all of this services using a web service call rather than using a GUI/command line interface.

Introduction

This paper attempts to explain a new way of automated application provisioning over the Cloud using the combination of simple and elegant tools such as CloudSmart, HybridFox and CloudBuddy Personal.

Background

Automated Provisioning is not something new as far as any large scale or repeated deployment goes. In the traditional way of large scale deployment, the Operating System and commonly-used applications are packaged as an image and deployed at one go. On the flip side, this is not a flexible and scalable model. Hence application provisioning systems such as Puppet, Chef are being used extensively.

Application Provisioning in Cloud requires mechanism to automate and repeat as and when it requires. This is mainly because the building blocks of an IT infrastructure such as computing, storage, network, and security are split into various blocks and converted into a set of services. You can reach all of this services using a web service call rather than using a GUI/command line interface.

The flexibility of provisioning IT resources dynamically, **as and when it requires**, necessitates a change even while you design your deployment architecture. This is mainly because of the nature of Cloud building blocks, which is stateless when it comes to computing and that's the feature which allows one to reap the benefits of automated scaling. The traditional application deployment assumes that the deployed applications would reside persistent and the infrastructure is made available upfront rather than on demand.

Typical steps being executed to deploy any application over the Cloud are:

- **Pre Deployment**
- Obtain an account with a Cloud Provider (in this case it is Amazon)
- Design your application deployment architecture
- Develop a run book to capture the necessary steps for deployment

Deployment

- Perform the run book steps manually using various tools
- **Post Deployment**
- Verify the deployment

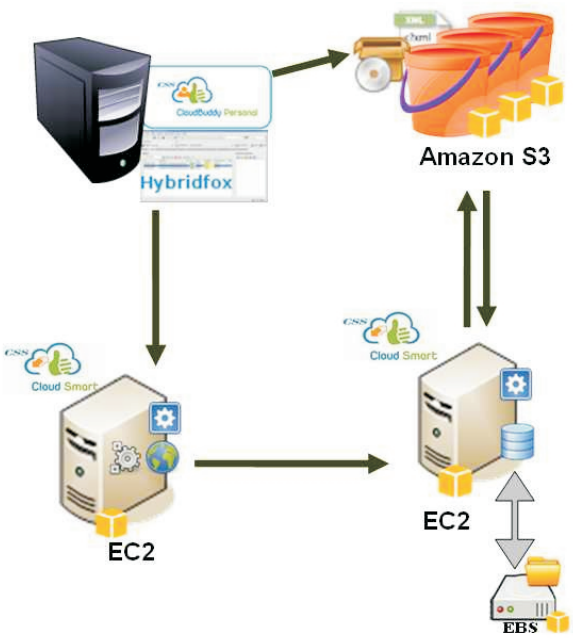
Automation here refers to automating the steps mentioned in any application deployment run book related to provisioning of instances, applications etc instead of performing them manually.

Solution - Automated Application Deployment

Amazon Web Services (AWS) is the pioneer in Cloud Computing and we have chosen AWS to explain our solution primarily because it is one of the most highly scalable and stable Cloud services available at this point. Also, we have chosen a web application to make it simpler.

Architecture

- Storing application related assets
- Launching an Instance
- Initialize CloudSmart during startup
- Creating EBS Volume
- Attaching EBS Volume
- Mounting & Formatting volume
- Creating DB directory in EBS volume
- Restarting SQL Server
- Deploying application in IIS
- Deploying Monitoring Agents
- Launching application



Introduction to the Tools

CloudSmart

CloudSmart tool provides flexible mechanism to automate your cloud application deployment. This tool is tested successfully and fully functioning in AWS (Amazon Web services) Cloud Environment. The tool is developed using ANT build tool

This document describes the mechanism for automating your application deployment in Windows and Linux environment over the AWS cloud infrastructure.

CloudSmart Tasks

Core Layer

The core layer of CloudSmart helps the user to perform the following activities:

S.No	Activities	Description
1	S3 Download	Helps trigger download application deployment files from S3 to the EC2 instance
2	BaseTask	Helps save the downloaded files in the required location of the EC2 instance
3	HTTPDownload	Facilitates the download from S3 using HTTP method
4	MetadataDownload	Helps acquiring the instance's metadata (AMI-ID, instance-ID, hostname etc) provided by AWS
5	ReplaceProperty	Helps replacing the properties of the files that are getting downloaded in the EC2 instance

Enhanced Layer(CloudSmart Tasks)

The enhancement layer of CloudSmart helps the user to perform few activities such as:

S.No	Activities	Description
1	CreateVolume	Helps creating the EBS volume by providing EBS size and device as inputs
2	AttachVolume	Helps attaching EBS volumes to EC2 instances
3	DetachVolume	Helps detaching EBS volumes from a EC2 instance
4	CreateVolAttach	Helps creating a EBS volume and attaching the same to a given EC2 instance
5	AttachIP	Helps associating an Elastic IP, which is already generated, to an Ec2 Instance

CloudBuddy Personal

CloudBuddy Personal is a handy tool to manage Amazon S3 storage. It has API's and Powershell CMDLETS to automate the tasks of uploading, downloading, etc.

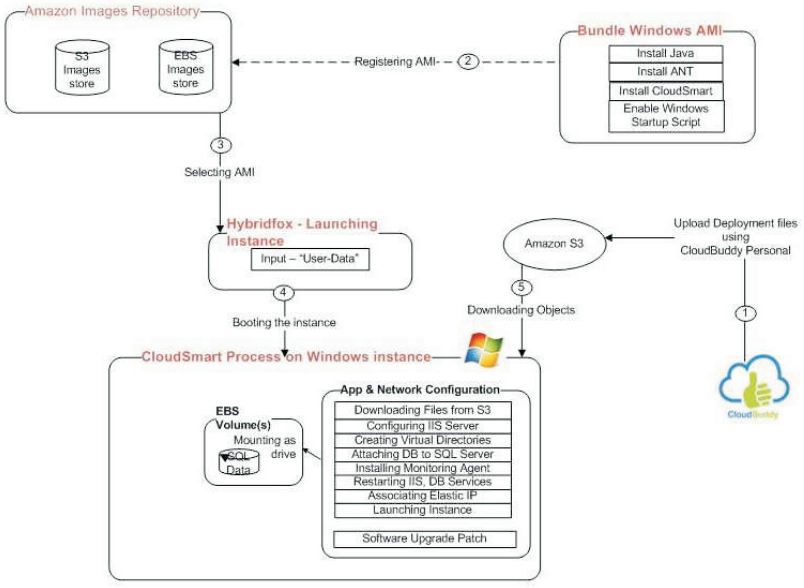
Please visit the following link for more information: http://www.mycloudbuddy.com/

HybridFox

Hybridfox is an attempt to get the best of both world of popular Cloud Computing environments, Amazon EC2 (public) and Eucalytpus(private). The idea is to use one Hybridfox tool, which itself is a modified or extended elasticfox, to switch seamless between your Amazon account and you Eucalyptus Account in order to manage your Cloud "Computing" environment.

Please visit the following link for more information: http://code.google.com/p/hybridfox/

Cloud Based Application Deployment Process



Uploading Deployment Files

All the deployment files will be stored in S3 using CloudBuddy Personal.

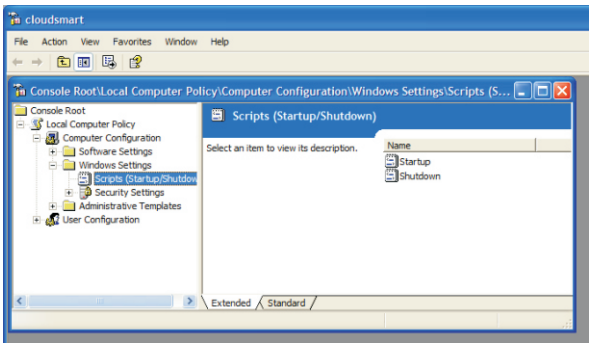
Bundling Windows AMI/EBS Image

- Windows AMI with following instances will be identified using Hybridfox /Elasticfox based on client requirements:
  - Application Servers
  - Database Servers
  - Web Servers etc
- Also it needs to be loaded with the following tools:
  - **JDK 1.5 or later from Sun**

    Install JDK on the instance and set the JAVA_HOME and add JAVA_HOME/bin in the "Path" as an environment variable on the Windows instance
  - **Ant 1.7 or later from Apache**

    Extract the ANT tool and set the ANT_HOME and add ANT_HOME/bin in the "Path"
  - **CloudSmart core tool**

    Configure the Start-up Script by utilizing Microsoft Management Console (MMC), where the user needs to mention the path of batch file which initiates CloudSmart to execute tasks associated with cloud deployment.



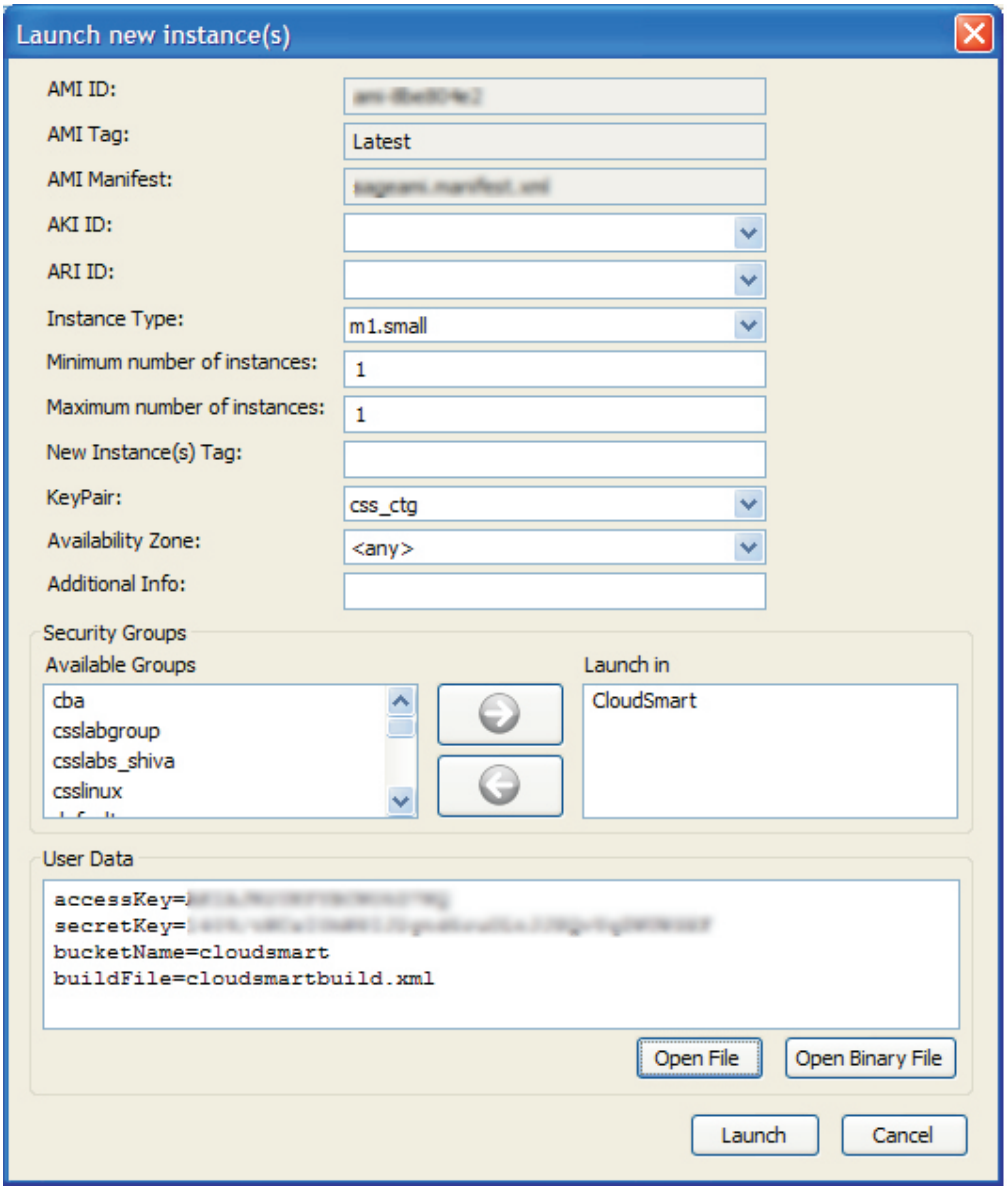
On successful completion of above tasks, bundle the instance which is having the CloudSmart deployment ready state. The bundled instance will be displayed as provided in below screenshot with a AMI ID associated with it

ID	Manifest	State	Owner	Visibility	Platform	Archi...	Tag
ami-08ab4c61	csspegfednew/css_pedfednew.manifest.xml	available	461933084714	private		i386	
ami-1b1efa72	csslabs_windows/ctg_windows_updated.manifest...	available	461933084714	private	windows	i386	
ami-2c8d6a45	css_ctgwinterminal/labswindows.manifest.xml	available	461933084714	private	windows	i386	
ami-44c1252d	csslinux/csslinux.manifest.xml	available	461933084714	private		i386	
ami-4ea34727	jboss-images/image.manifest.xml	available	461933084714	public		i386	
ami-5af81c33	cssdebian/cssdebian.manifest.xml	available	461933084714	public		i386	
ami-5e5ebb37	/jbosswithant/image.manifest.xml	available	461933084714	private		i386	
ami-612ace08	smartdig_1/smartdig_1.manifest.xml	available	461933084714	private		i386	
ami-629e7c0b	cbaservice-bundle/image.manifest.xml	available	461933084714	private		i386	
ami-66b95e0f	csssage_winbackup/css_winfinal.manifest.xml	available	461933084714	private	windows	i386	
ami-6842a701	csslabami/image.manifest.xml	available	461933084714	private		i386	
ami-7b15f612	cbanalytics-test/cbanalytics-test.manifest.xml	available	461933084714	private		i386	
ami-8642a7ef	jlibwithjungleami/image.manifest.xml	available	461933084714	private		i386	
ami-88e601e1	csssage_newwin/sage_newwindows.manifest.xml	available	461933084714	private	windows	i386	
ami-8be804e2	[redacted]ami.manifest.xml	available	461933084714	private	windows	i386	Latest
ami-8f39d5e6	cbaservice-image/cbaservice-with-sessions.12.02...	available	461933084714	private		i386	
ami-cb3bdfa2	sdig2/sdig2.manifest.xml	available	461933084714	private		i386	
ami-dfe007b6	cssage_windemo/sagewindowsserver.manifest.xml	available	461933084714	private	windows	i386	
ami-e3f01c8a	cbaservice-new/cbaservice-scale.06.02.2010.ma...	available	461933084714	private		i386	
ami-e552b78c	/jboss-working-ami/image.manifest.xml	available	461933084714	private		i386	
ami-eaab4c83	csspegdebnew/csspeg_debnew.manifest.xml	available	461933084714	private		i386	
ami-f9a94e90	css_pegnewwin/csspeg_win.manifest.xml	available	461933084714	private	windows	i386	

Launching the Windows AMP/EBS Image

- The bundled AMI with pre-requisites in place will be launched using Hybridfox by providing the following user-data:

S.No	Activities	Description
1	Access Key	The access key for the S3 Account
2	Secret Key	The secret key for the S3 Account
3	Bucket Name	The bucket where the file with cloud deployment scripts will exist
4	Build File	The file which contains the cloud deployment scripts



Samples of Deployment Scripts

To set property

```
<property name="IISRoot" value="C:\Inetpub\wwwroot\"/>
```

To download application deployment files from S3

```
<s3download accessKey="${accessKey}" secretKey="${secretKey}" bucketName="cloudsmart"
S3ObjectPath="cloudsmartwin/SRTS/SRTS.zip" localFolder="${s3buildPath}" />
        <exec executable="cmd">
                <arg line="/c jar xf SRTS.zip"/>
              </exec>
            <move todir="${IISRoot}\SRTS">
          <fileset dir="${s3buildPath}\SRTS"/>
        </move>
```

To create & attach EBS

```
<CreateVolAttach accessKey="${accessKey}" secretKey="${secretKey}" size="1"
instanceIDURL="http://169.254.169.254/latest/meta-data/instance-id" device="xvdp" />
```

To attach Elastic IP

```
<AttachIP accessKey="${accessKey}" secretKey="${secretKey}" elasticIP="75.101.137.183"
instanceIDURL="http://169.254.169.254/latest/meta-data/instance-id" />
```

Start-up Tasks

- Downloads the application requirement files to the EC2 instance from S3
- Save and extract all the files to web / application server
- Configure the server and dynamically creates the virtual directories, if required
- Change the host-name and the machine name in the “web.config” file, if needed
- Creates and attach the EBS volume to the instance
- Mount and format the drive as a windows drive for SQL DB, FileStore and application Backup drives, if required
- Copy the MS-SQL Server data directory in to EBS drive, if required
- Attach the database to the MS-SQL Server and refresh the DB, if required
- Associating Elastic IP to EC2 instance, if needed

Intermediate Tasks

- Run the update patches for software / OS / applications etc
- Upgrade EC2 instance configuration from small to medium/large, if needed

Future Enhancements

- Enable the logs using build file, which shall be loaded to S3 from EC2 instance
- Scheduler periodic snapshot of EBS volumes
- Enable automation script for cluster setup
- User-friendly management console for easy deployment.

CSS
RATIONALIZE · OPTIMIZE · INNOVATE

About CSS

CSS, a global technology solutions provider, focuses on IT operations management services aimed at optimizing its customer's IT operations. CSS offers solutions in the area of Enterprise and Consumer Technology Support, Application Lifecycle Management and Remote Infrastructure Management Services. The company has proven expertise in developing, supporting and managing its customer's entire technology stack (servers, networks, systems and technologies as well as applications running on them) on a 24x7 basis globally. In addition, CSS has an impeccable record of delivering the whole gamut of technical and customer support services for both enterprise and consumer technology products and applications on a global basis. With a unique "100% referenceable customers" mission statement, the company lives and breathes operations support and customer satisfaction.

Head-quartered in San Jose, CA, CSS currently has global operations centers in Chennai (India), Utah (USA), Poland (Europe) and in the Philippines. CSS has over 80 customers including Industry leaders such as Alcatel-Lucent, Sun Microsystems, & Blackboard Inc as its long term customers. CSS is a professionally managed, privately held company with investments from SAIF, Goldman Sachs and Sierra Ventures.

© 2010 CSS Corporation. All rights reserved.